Sicherung sensibler Daten mithilfe AWS Secrets Manager von HashiCorp Terraform

AWS Prescriptive Guidance

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Prescriptive Guidance: Sicherung sensibler Daten mithilfe AWS Secrets Manager von HashiCorp Terraform

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Die Handelsmarken und die Handelsaufmachung von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, durch die Kunden irregeführt werden könnten oder Amazon in schlechtem Licht dargestellt oder diskreditiert werden könnte. Alle anderen Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Einführung 1
Zielgruppe 1
Ziele 1
Secrets Manager verwenden und Terraform 3
AWS Secrets Manager 3
Terraform 3
Bewährte Methoden 5
Generieren zufälliger Passwörter 5
Geheimnisse rotierend 7
Beschränkung des Zugriffs 10
Verwendung einer Namenskonvention 11
Verwaltung sensibler Daten 13
Schutz sensibler Daten in Terraform Statusdatei 13
Verwaltung von Geheimnissen für Amazon EKS 14
Aufbewahrung sensibler Daten in bekannten Netzwerken 16
Ressourcen 18
AWS Dokumentation 18
AWS Blog-Beiträge 18
Sonstige Ressourcen 18
Mitwirkende 19
Verfassen 19
Überprüfend 19
Technisches Schreiben 19
Dokumentverlauf 20
..... xxi

Sicherung sensibler Daten durch die Verwendung von AWS Secrets Manager und HashiCorp Terraform

Amazon Web Services (Mitwirkende)

Mai 2025 (Verlauf der Dokumente)

Die Verwaltung sensibler Daten, darunter Anmeldeinformationen, geheime Zeichenketten und Passwörter, ist ein anerkannter Pfeiler des Infrastrukturmanagements sowie der Anwendungsentwicklung und -bereitstellung. Um Ihr Unternehmen zu schützen, sollten Sie bewährte Methoden für die Verwaltung sensibler Daten in der Cloud anwenden. Der Schutz sensibler Daten ist eine Grundvoraussetzung für Sicherheit und Compliance. AWS Secrets Managerkann dazu beitragen, sensible Daten in Ihrer Umgebung als geheime Daten zu schützen.

In diesem Handbuch werden bewährte Methoden für geheime Daten beschrieben, z. B. wie Sie Geheimnisse aus Secrets Manager abrufen und wie AWS Lambda Sie geheime Daten automatisch rotieren können. Es enthält auch Empfehlungen zur Verwaltung und Verwaltung von Geheimnissen mithilfe hierarchischer Namen. Schließlich hilft es Ihnen bei der Verwaltung der Verwendung von und des Zugriffs auf geheime Daten, z. B. in Bezug auf Zentralisierung, Terraform Integration und Netzwerkunterstützung.

HashiCorp Terraformwurde in der Branche weithin als Infrastructure-as-Code-Lösung (IaC) eingesetzt. TerraformZeigt sensible Daten jedoch als Klartext in der Statusdatei an. Dieses Handbuch enthält bewährte Methoden für die Verwaltung sensibler Daten sowie für die Erstellung und Verwendung von Secrets Manager Manager-Geheimnissen. Terraform

Zielgruppe

Dieses Handbuch richtet sich an Unternehmen, die es Terraform als IaC-Lösung verwenden möchten. Die bewährten Methoden in diesem Leitfaden sollen Datenbankarchitekten, Infrastrukturteams und Anwendungsentwicklern helfen. Vertrautheit mit diesem Leitfaden Terraform ist eine Grundvoraussetzung.

Ziele

Im Folgenden sind die Geschäftsergebnisse aufgeführt, die Sie nach der Umsetzung der Empfehlungen in diesem Leitfaden erwarten können:

- Sorgen Sie für schnellere Innovationen, indem Sie den Prozess der Verwaltung von Geheimnissen automatisieren.
- Verbessern Sie die Sicherheitslage Ihres Unternehmens in der. AWS Cloud

Im Folgenden sind die technischen Ergebnisse aufgeführt, die Sie nach der Umsetzung der Empfehlungen in diesem Leitfaden erwarten können:

- Verwenden Sie Secrets Manager, um die Offenlegung vertraulicher Daten in der Terraform Statusdatei zu verhindern.
- Zentralisieren Sie die Verwaltung von Geheimnissen und sensiblen Daten, um die Verwaltung zu verbessern und die Einhaltung von Vorschriften zu gewährleisten.
- Setzen Sie bewährte Sicherheitsverfahren in den Prozessen Ihres Unternehmens für die Bereitstellung der Cloud-Infrastruktur durch.

Secrets Manager verwenden und Terraform

AWS Secrets Manager

AWS Secrets Manager ist ein Service zum sicheren Verschlüsseln, Speichern und Rotieren von Anmeldeinformationen für Datenbanken und andere Services. Es hilft Ihnen dabei, hartcodierte Anmeldeinformationen in Ihrem Code, einschließlich Kennwörtern, durch einen API-Aufruf zu ersetzen, um das Geheimnis programmgesteuert abzurufen. In Secrets Manager besteht ein Geheimnis aus Anmeldeinformationen (was der geheime Wert ist) und seinen Metadaten. Der geheime Wert kann binär, eine einzelne Zeichenfolge oder mehrere Zeichenketten sein. Weitere Informationen finden Sie unter Secret.

Secrets Manager verwendet symmetrische 256-Bit-Datenschlüssel des Advanced Encryption Standard (AES), um geheime Werte zu verschlüsseln. Weitere Informationen finden Sie unter Verschlüsseln und Entschlüsseln von Geheimnissen in AWS Secrets Manager.

Sie können auf Secrets Manager zugreifen und mit ihm arbeiten, indem Sie eine der folgenden Methoden verwenden:

- Secrets-Manager-Konsole
- Befehlszeilen-Tools
- AWS SDKs
- HTTPS-Abfrage-API, auch Secrets Manager Manager-API genannt
- AWS Secrets Manager Endpunkte

Terraform

Terraformist ein IaC-Tool von HashiCorp , mit dem Sie Cloud- und lokale Ressourcen erstellen und verwalten können. Sie können es verwendenTerraform, um Ressourcen und Infrastruktur in der bereitzustellen. AWS Cloud

Terraformspeichert Informationen über Ihre verwaltete AWS Infrastruktur und deren Konfigurationen. Diese Informationen werden als Status bezeichnet. Standardmäßig wird der Status in einer lokalen Datei mit dem Namen gespeichert`Terraform.tfstate`. Diese Datei ist im JSON-Format und speichert Terraform möglicherweise sensible Daten in dieser Statusdatei im Klartext. Dies stellt

ein Risiko für die vertraulichen Daten dar, da jeder Benutzer mit Zugriff auf die Statusdatei auf die vertraulichen Daten zugreifen kann.

Dieses Handbuch enthält bewährte Verfahren und Empfehlungen zum Schutz vertraulicher Daten bei der Verwaltung Ihrer AWS Ressourcen. Terraform

Bewährte Methoden für das Erstellen, Rotieren und Verwenden von Geheimnissen

In diesem Abschnitt werden bewährte Verfahren und Standards beschrieben, die Sie bei der Programmierung von Anwendungen und Infrastruktur für die Bereitstellung beachten könnenTerraform. Sie enthält Folgendes:

- Generieren zufälliger Passwörter mithilfe von AWS Secrets Manager — Generieren Sie ein zufälliges Geheimnis mit einer bestimmten Komplexität gemäß den Compliance-Richtlinien Ihres Unternehmens.
- Verwendung AWS Lambda zur Rotation von Geheimnissen — AWS Lambda kann so geplant werden, dass sie bei der Rotation von Geheimnissen automatisch ausgeführt wird.
- Beschränkung des Zugriffs auf Geheimnisse — Schränken Sie ein, wer auf Geheimnisse zugreifen kann und wie sie darauf zugreifen können.
- Verwendung einer hierarchischen Namenskonvention für Geheimnisse — Definieren und verwenden Sie eine Namenskonvention, die Ihnen hilft, Geheimnisse in großem Umfang zu verwalten.

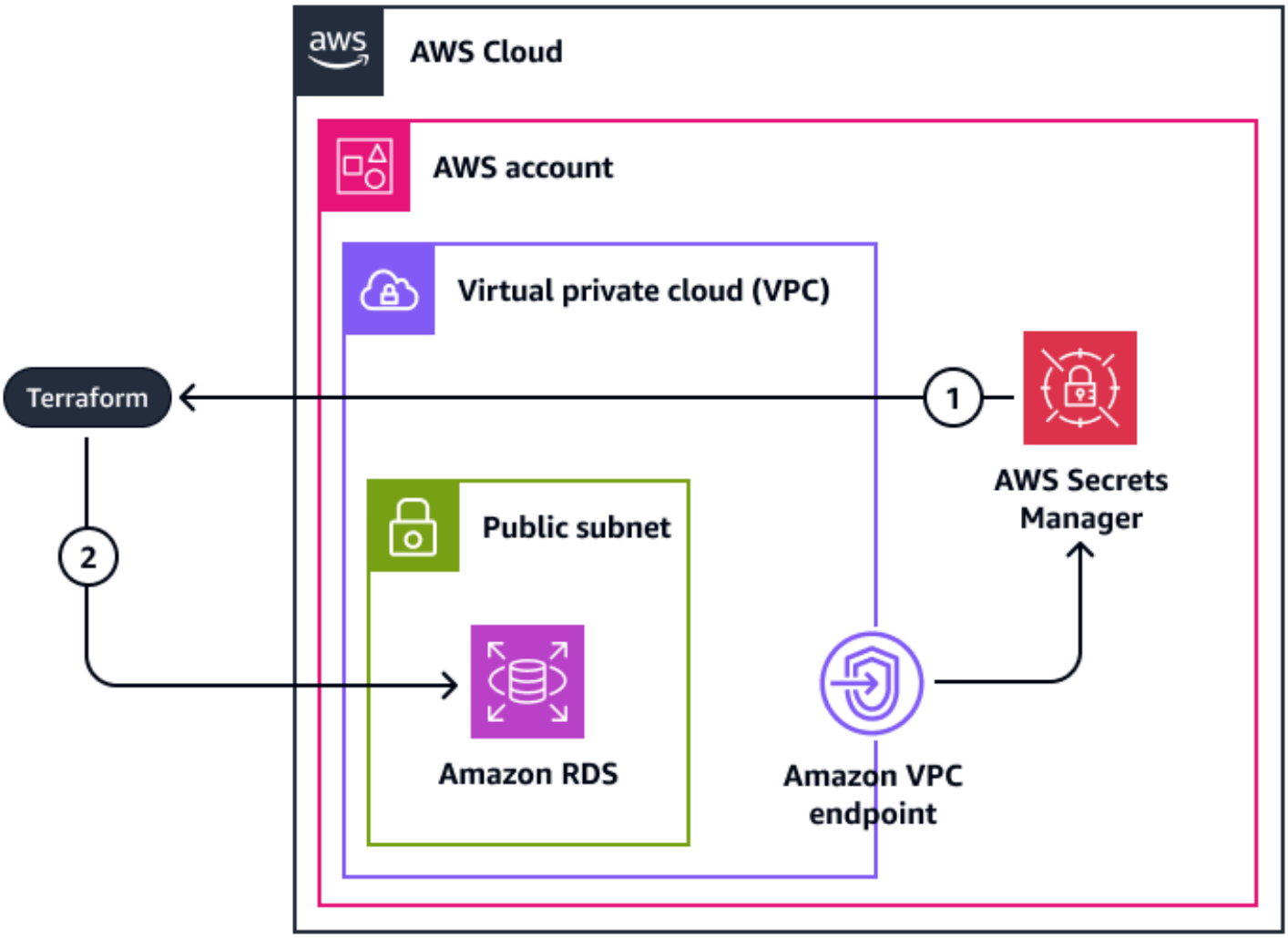
Generieren zufälliger Passwörter mithilfe von AWS Secrets Manager

Das AWS Well-Architected Framework empfiehlt, dass Sie Geheimnisse sicher speichern und verwenden. Sie können die AWS Secrets Manager API verwenden, um zufällige Passwörter zu generieren, und Sie können die Anforderungen an die Passwortkomplexität anpassen. Die `GetRandomPassword` Aktion unterstützt Kennwortzeichenfolgen mit Längen zwischen 1 und 4.096 Zeichen. Weitere Informationen finden Sie unter GetRandomPassword in der AWS Secrets Manager -API-Referenz. Wir empfehlen, diesen Ansatz zur Generierung von Geheimnissen zu verwenden, anstatt es Benutzern zu ermöglichen, Geheimnisse manuell zu definieren.

Das folgende Codebeispiel zeigt, wie Sie zufällige Passwörter mit einer Länge von 20 Zeichen generieren können, die Zahlen enthalten, Satzzeichen ausschließen und Leerzeichen ausschließen. Sie können dieses Codebeispiel ändern, um die Kennwortsicherheitsanforderungen Ihrer Organisation zu erfüllen.

```
data "aws_secretsmanager_random_password" "test" {
password_length = 20
exclude_numbers = false
exclude_punctuation = true
include_space = false
}
```

Durch die Generierung zufälliger Geheimnisse bei der Bereitstellung von IaC können Sie sensible Daten von Anfang an schützen, was auch als Nullzeit bezeichnet wird. Die sensiblen Daten sind von der Bereitstellungsphase an niemandem bekannt.



1. DurchTerraform, wird verwendet, AWS Secrets Manager um ein zufälliges geheimes Passwort zu generieren.

2. Terraformverwendet dieses zufällige geheime Passwort, das in gespeichert ist AWS Secrets Manager, für den Zugriff auf die Datenbank.

> **Important**
>
> Wenn Sie es Terraform als Datenquelle verwenden, werden Geheimnisse nicht in der Statusdatei gespeichert. Nachdem Sie dieses Geheimnis jedoch in einer Datenbank oder einem Dienst verwendet haben, wird es in der Statusdatei gespeichert. Wir empfehlen, dass Sie die Secrets sofort rotieren oder sehr restriktive Zugriffsberechtigungen für die Statusdatei einrichten. Weitere Informationen finden Sie unter Schutz sensibler Daten in Terraform Statusdatei in diesem Handbuch.

Verwenden AWS Lambda um Geheimnisse zu rotieren

Das AWS Well-Architected Framework empfiehlt, dass Sie Geheimnisse sicher speichern und verwenden. Diese bewährte Methode empfiehlt, die Rotation von Anmeldeinformationen in regelmäßigen Abständen zu automatisieren. Die Rotation ist der Prozess, bei dem Sie das Secret in regelmäßigen Abständen aktualisieren. So erschweren Sie einem Angreifer den Zugriff auf die Anmeldeinformationen. Viele Compliance-Frameworks und -Vorschriften verlangen außerdem, dass Sie geheime Daten austauschen müssen.

Für Terraform IaC können Sie die automatische Rotation verwenden AWS Secrets Manager und AWS Lambdaeinrichten. Im Secrets Manager können Sie die automatische Rotation für Ihre Secrets einrichten. Wenn Secrets Manager ein Secret rotiert, werden die Anmeldeinformationen sowohl im Secret als auch in der Datenbank oder im Dienst aktualisiert.

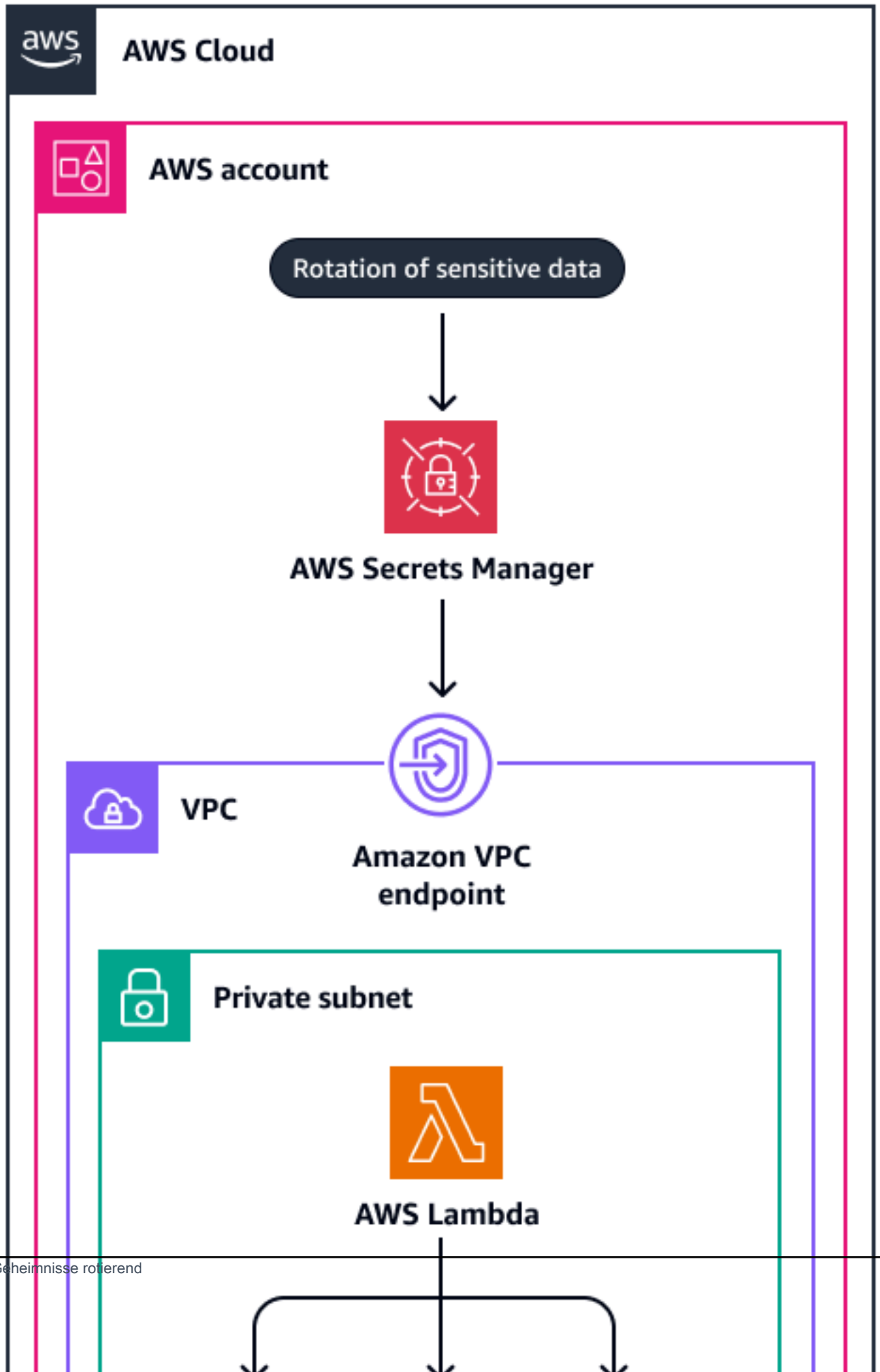
Für Datenbanken empfehlen wir, dass Sie die primären Anmeldeinformationen in Secrets Manager verwalten und die Secrets in regelmäßigen Abständen rotieren. Secrets Manager bietet Rotationsfunktionsvorlagen für Lambda für verschiedene Arten von Datenbankanmeldedaten. Weitere Informationen finden Sie unter Vorlagen für AWS Secrets Manager Rotationsfunktionen in der Secrets Manager Manager-Dokumentation und in den Codebeispielen unter GitHub. Im Folgenden finden Sie ein Beispiel für eine Terraform IaC, mit der Sie geheime oder sensible Daten rotieren können.

```
resource "aws_secretsmanager_secret_rotation" "createrotation" {
  count               = var.needrotation == true ? 1 : 0
```

```
  secret_id           = aws_secretsmanager_secret.initiatesecret.id
  rotation_lambda_arn = aws_lambda_function.rotationlambda.arn

  rotation_rules {
    automatically_after_days = 1
  }
}
```

Das folgende Architekturdiagramm zeigt, wie Sie Secrets Manager, einen Amazon VPC-Endpunkt und eine Lambda-Funktion verwenden können, um sensible Daten in einem zu rotieren. AWS-Konto


AWS Cloud
AWS account
Rotation of sensitive data
AWS Secrets Manager
VPC
Amazon VPC endpoint
Private subnet
AWS Lambda

Beschränken des Zugriffs auf Geheimnisse

Sicherheit hat bei oberste Priorität AWS. Im AWS Well-Architected Framework empfiehlt die bewährte Methode zum sicheren Speichern und Verwenden von Geheimnissen, dass Sie den Zugriff auf Geheimnisse einschränken, den Zugriff auf Geheimnisse protokollieren und prüfen und dass Sie den Zugriff über eine API erhalten, anstatt einer Person Zugriff zu gewähren. Dies bezieht sich auf das Generieren zufälliger Passwörter mithilfe von AWS Secrets Manager, das bereits in diesem Handbuch behandelt wurde.

In der Vergangenheit waren Anmeldeinformationen, die zur Authentifizierung bei Datenbanken, APIs, Token und anderen Geheimnissen von Drittanbietern verwendet wurden, möglicherweise in Quellcode oder in Umgebungsdateien eingebettet. In vielen Organisationen können Datenbankadministratoren immer noch auf alle Anmeldeinformationen von Datenbankbenutzern zugreifen. Es ist jedoch sicherer, wenn sie dies nicht tun. Wir empfehlen Ihnen:

- Beschränken Sie den Zugriff auf Secrets Manager über eine AWS Identity and Access Management (IAM-) Richtlinie, die an eine AWS Service- oder IAM-Rolle für Benutzer angehängt ist.
- Erweitern Sie den programmatischen Zugriff auf Secrets Manager nur durch API-Aufrufe an Secrets Manager.
- Vermeiden Sie die Verwendung von fest codierten Geheimnissen.
- Übergeben Sie keine Werte für Anmeldeinformationen in Variablen.
- Stellen Sie sicher, dass Geheimnisse direkt von der Anwendung oder Infrastruktur und nicht von einem menschlichen Benutzer verwendet werden.

Das folgende Codebeispiel zeigt einen Ansatz, mit dem Sie vertrauliche Daten beim Erstellen einer Datenbank schützen können. Wenn Sie die Datenbank erstellen, können Sie Benutzereingaben als Passwort verlangen und dann das Passwort wechseln, oder Sie können von Anfang an ein zufälliges Passwort verwenden. Wenn Sie von Anfang an ein zufälliges Passwort generieren, können Sie Anmeldeinformationen sicherer handhaben und verwalten.

Der folgende Codeausschnitt erstellt eine Datenbankressource und `password = local.passwddata` ruft das Passwort aus einer lokalen Variablen ab.

```
resource "aws_db_instance" "default" {
  count                  = var.createdb == true ? 1 : 0
```

```
  allocated_storage      = 20
  identifier             = var.identifier
  db_name                = var.databasename
  engine                 = "mysql"
  engine_version         = "8.0.32"
  instance_class         = "db.t3.micro"
  username               = "admin"
  password               = local.passwddata
  skip_final_snapshot    = true
  publicly_accessible    = false
  vpc_security_group_ids = [aws_security_group.db.id]
  db_subnet_group_name   = aws_db_subnet_group.db.0.name
}
```

Der folgende Codeausschnitt erstellt ein zufälliges Passwort oder akzeptiert das vom Benutzer angegebene Passwort. Terraformspeichert dieses Passwort als lokale Variable, sodass Sie diese Kennwortvariable später im Code verwenden können.

```
locals {
  passwddata = var.passwddata != null ? var.passwddata :
 data.aws_secretsmanager_random_password.test.random_secrets
}
```

Es wird eine hierarchische Namenskonvention für Geheimnisse verwendet

Verwenden Sie beim Aufnehmen und Speichern von AWS Secrets Manager Geheimnissen eine definierte Namenskonvention. Ihre Benennungskonvention könnte beispielsweise lauten. `<client name>/<dev or prod>/<project>/<version>` Wählen Sie eine Benennungskonvention, die für Ihre Organisation sinnvoll ist. Verwenden Sie eine hierarchische Struktur, die von den allgemeinsten Informationen zu den spezifischsten Informationen übergeht.

Dies hilft Ihnen bei der Verwaltung von Geheimnissen, insbesondere wenn ein zentrales Konto für viele Geheimnisse in Ihrer AWS Umgebung verwendet wird. Eine hierarchische Namenskonvention kann Ihnen auf folgende Weise helfen:

- Sie hilft Ihnen dabei, den Zugriff auf verschiedene Geheimnisse voneinander zu trennen. Sie können beispielsweise definieren, welches Projekt oder welche Umgebung Zugriff auf welche Geheimnisse haben soll.

- Auf diese Weise können Sie anhand ihrer Namen detaillierte Zugriffskontrollen für geheime Daten einrichten.

Das folgende Codebeispiel zeigt ein Beispiel für eine Namenskonvention, mit der Sie Geheimnisse in großem Umfang verwalten können.

```
resource "aws_secretsmanager_secret" "initiatesecret" {
  name                    = "org-name/dev-env/project-name"
  kms_key_id              = var.kmskeyarn
  recovery_window_in_days = var.recoverywindow
}
```

Verwaltung sensibler Daten bei der Verwendung Terraform

Sensible Daten müssen gut strukturiert sein. Mit zunehmender Skalierung von Anwendungen und Infrastruktur wird es immer wichtiger, sensible Daten sorgfältig zu verfolgen und zu behandeln. AWS-Konten Bei der Implementierung von Terraform IaC können Sie die folgenden Ansätze verwenden, um vertrauliche Daten in Ihrem System zu schützen:

- Schutz sensibler Daten in Terraform Statusdatei— Sie können dazu beitragen, sensible Daten von dem Moment an zu schützen, in dem sie zum ersten Mal aufgenommen werden. AWS Secrets Manager Sie könnten das Geheimnis beispielsweise sofort rotieren lassen, um seine Geheimhaltung zu wahren.
- Zugriff auf und Verwaltung von Geheimnissen für Amazon EKS— Verwaltet alle Secrets für Amazon Elastic Kubernetes Service (Amazon EKS) im Secrets Manager.
- Verwendung von VPC-Endpunkten zur Aufbewahrung sensibler Daten in bekannten Netzwerken— Der Datenverkehr für sensible Daten sollte private Netzwerke nicht verlassen. Dies trägt dazu bei, Angriffe und Datenexfiltration zu verhindern.

Schutz sensibler Daten in Terraform Statusdatei

In diesem Abschnitt wird die Verschleierung von Geheimnissen und Zeigern für den Umgang mit den sensiblen Daten in der Terraform Statusdatei beschrieben, genannt. `tfstate` In der Regel handelt es sich dabei um eine Klartextdatei, die Daten über Terraform Bereitstellungen sowie alle sensiblen und nicht sensiblen Daten über die bereitgestellte Infrastruktur enthält. Vertrauliche Daten sind in der Terraform Statusdatei im Klartext sichtbar. Gehen Sie wie folgt vor, um vertrauliche Daten zu schützen:

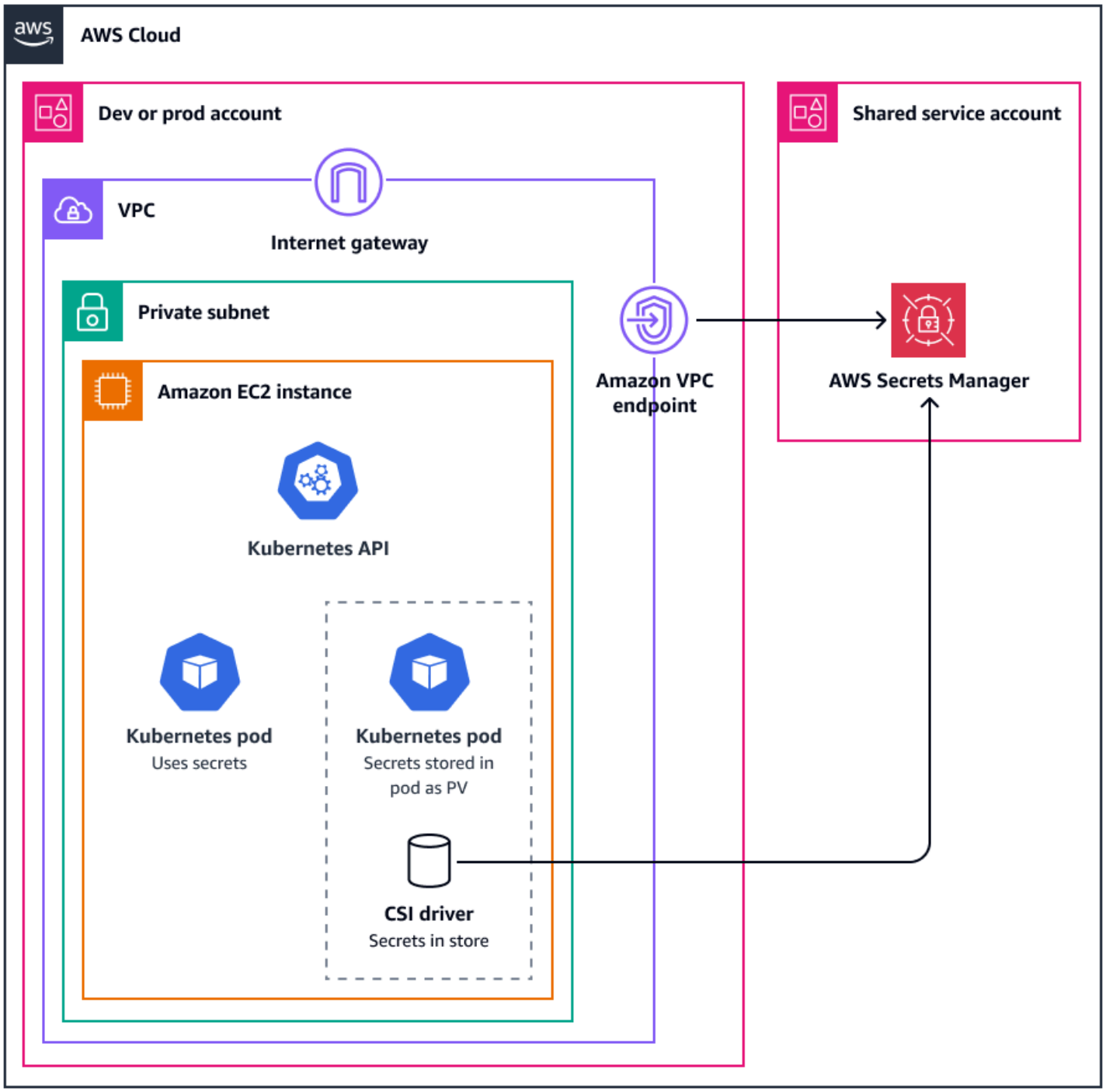
- Wenn Sie ein Geheimnis aufnehmen, wählen Sie, ob das Geheimnis sofort rotiert werden soll. Weitere Informationen finden Sie unter Sofortiges Rotieren eines AWS Secrets Manager Secrets in der Secrets Manager Manager-Dokumentation.
- Speichern Sie die Terraform Statusdatei in der Zentrale AWS-Konto , in der Sie Secrets Manager betreiben. Speichern Sie die Datei in einem Amazon Simple Storage Service (Amazon S3) -Bucket und konfigurieren Sie Richtlinien, die den Zugriff darauf einschränken. Weitere Informationen finden Sie unter Bucket-Richtlinien und Benutzerrichtlinien in der Amazon S3 S3-Dokumentation.

- Sie können den Terraform Status sperren, um Korruption zu verhindern. Weitere Informationen zum Sperren des Status und zum Schutz der Statusdatei finden Sie im Amazon S3 S3-Backend in der Terraform Dokumentation.

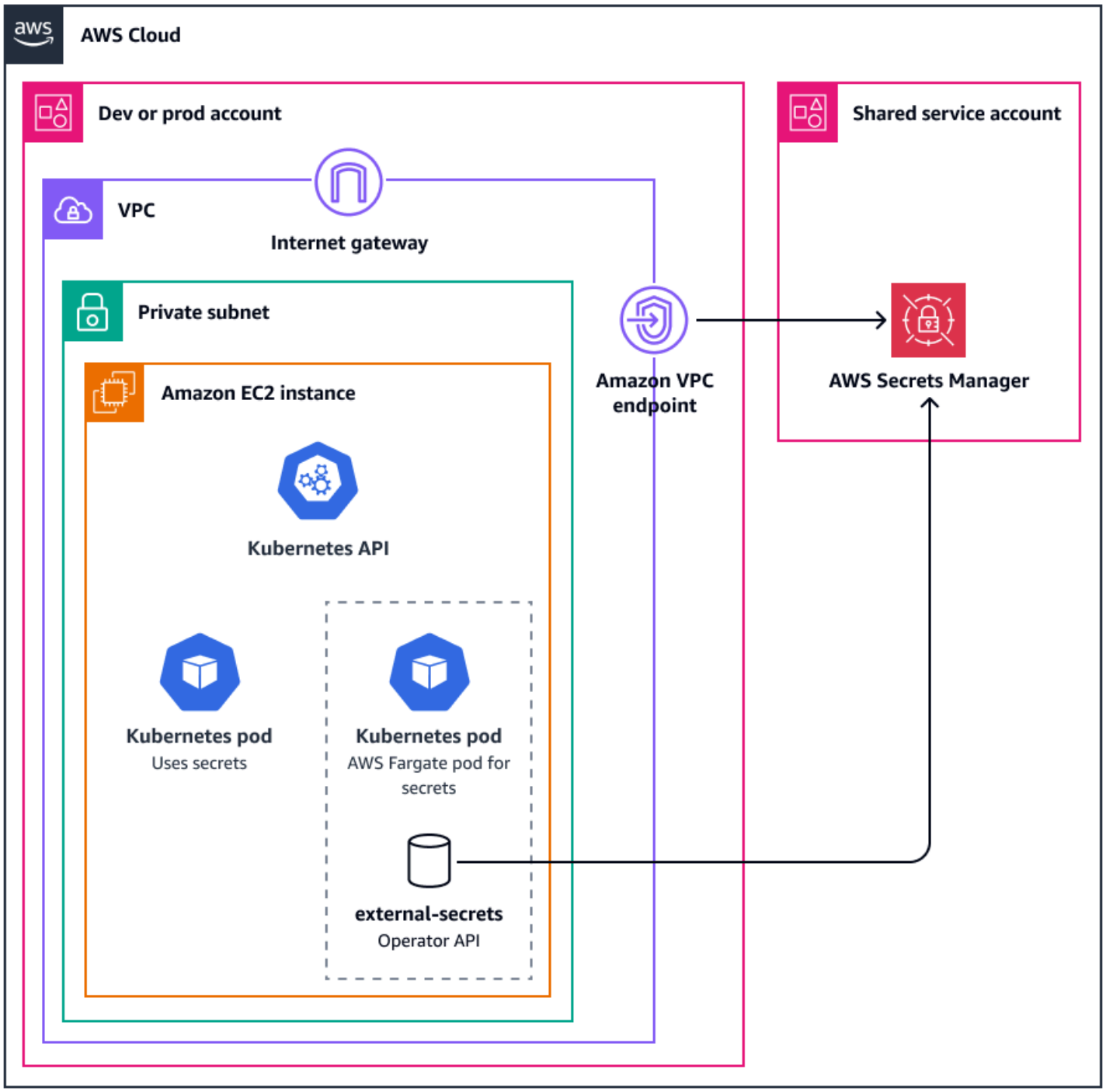
Zugriff auf und Verwaltung von Geheimnissen für Amazon EKS

Mit Amazon Elastic Kubernetes Service (Amazon EKS) können Sie Kubernetes ausführen, AWS ohne dass Sie Ihre eigene Kubernetes-Steuerebene oder Knoten installieren oder verwalten müssen. Amazon EKS verwendet die Base64-Kodierung (Wikipedia), um sensible Daten zu schützen. Die Verschlüsselung wurde entwickelt, um Datenänderungen während der Übertragung zwischen Systemen zu verhindern, und die Verschlüsselung soll unbefugten Zugriff auf die Daten verhindern. Die Base64-Kodierung reicht nicht aus, um Daten vor unbefugtem Zugriff zu schützen. Wird verwendet AWS Secrets Manager , um sensible Daten in Amazon EKS zu schützen.

Die folgende Abbildung zeigt Amazon EKS, das auf einer Amazon Elastic Compute Cloud (Amazon EC2) -Instance bereitgestellt wird, die als Kubernetes-Worker-Knoten fungiert. Sie können einen Container Storage Interface (CSI) -Treiber verwenden, um Geheimnisse aus Secrets Manager abzurufen. Weitere Informationen finden Sie im AWS Sicherheitsblog unter So verwenden Sie AWS Secrets & Configuration Provider mit Ihrem Kubernetes Secrets Store CSI-Treiber.



Die folgende Abbildung zeigt Amazon EKS, das auf bereitgestellt wird AWS Fargate. Sie können die Open-Source-API für externe Secrets Operator (GitHub) verwenden, um Geheimnisse aus Secrets Manager abzurufen. Weitere Informationen finden Sie unter Leverage AWS Secrets Stores von EKS Fargate with External Secrets Operator im AWS Container-Blog.



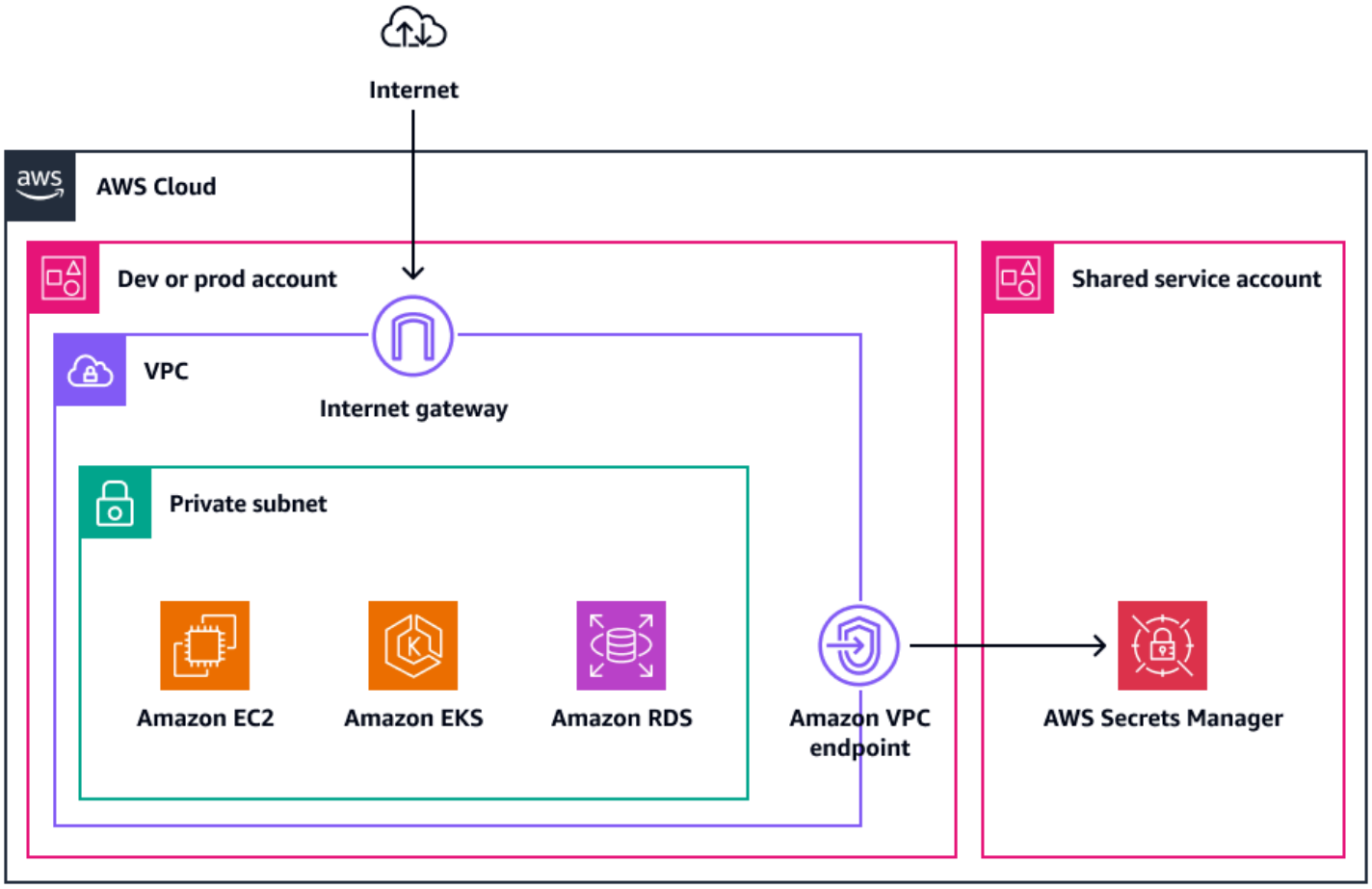
Verwendung von VPC-Endpunkten zur Aufbewahrung sensibler Daten in bekannten Netzwerken

Geheimnisse sollten niemals über das Internet zugänglich sein. AWS bietet Optionen zur Wahrung der Privatsphäre bei der Weiterleitung von Datenverkehr über bekannte und private Netzwerkrouten.

Bei der Konfiguration des Datenverkehrs zwischen AWS Secrets Manager und lokalen Clients und Anwendungen können Sie einen der folgenden Ansätze verwenden:

- Eine AWS Site-to-Site VPN VPN-Verbindung
- AWS Direct Connect-Verbindung

Wenn Sie den Datenverkehr zwischen Secrets Manager und API-Clients damit sichern möchten AWS-Region, verwenden Sie diese Option, AWS PrivateLinkum VPC-Endpunkte für Schnittstellen zu erstellen. Wenn Sie diese Option verwenden, behalten Sie den gesamten Datenverkehr in Ihrem privaten Netzwerk geheim. Weitere Informationen finden Sie unter Verwenden von AWS Secrets Manager -VPC-Endpunkten.



Ressourcen

AWS Dokumentation

- AWS Secrets Manager Geheimnisse rotieren
- Drehe ein AWS Secrets Manager Geheimnis sofort um
- AWS Secrets Manager Vorlagen für Rotationsfunktionen
- Datenschutz in AWS Secrets Manager
- Verwenden eines AWS Secrets Manager VPC-Endpunkts
- S3-Bucket-Richtlinien und Benutzerrichtlinien
- Geheimnisse sicher speichern und verwenden (AWS Well-Architected Framework)
- Regelmäßige Prüfung und Rotation der Anmeldeinformationen (AWS Well-Architected Framework)

AWS Blog-Beiträge

- Wie kann man auf geheime Daten zugreifen, AWS-Konten indem man ressourcenbasierte Richtlinien anhängt
- So verwenden Sie AWS Secrets & Configuration Provider mit Ihrem Kubernetes Secrets Store CSI-Treiber
- Nutzen Sie AWS Secrets Stores von EKS Fargate mit External Secrets Operator

Sonstige Ressourcen

- Amazon S3 S3-Backend (TerraformDokumentation)

Mitwirkende

Die folgenden Personen haben zu diesem Leitfaden beigetragen.

Verfassen

- Chintamani Aphale, leitender Berater, AWS
- Aarti Rajput, Leitender Berater, AWS
- T.V.R.L. Phani Kumar Dadi, Berater, AWS
- Pradip Kumar Pandey, leitender Berater, AWS
- Mayuri Shinde, Berater, DevOps AWS
- Pratap Kumar Nanda, leitender Berater, AWS

Überprüfend

- Janardhan Malyala, leitender Sicherheitsberater, AWS
- Anand Krishna Varanasi, Leitender Berater, AWS
- Sandeep Gawande, Leitender Berater, AWS
- Mike Zazon, Hauptarchitekt für Cloud-Infrastruktur, AWS

Technisches Schreiben

- Lilly AbouHarb, leitende technische Redakteurin, AWS

Dokumentverlauf

In der folgenden Tabelle werden wichtige Änderungen in diesem Leitfaden beschrieben. Um Benachrichtigungen über zukünftige Aktualisierungen zu erhalten, können Sie einen RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
Sperre der Statusdatei	Wir haben den Abschnitt Schutz sensibler Daten in der Terraform Statusdatei aktualisiert. Die Verwendung einer Amazon DynamoDB-Tabelle, um die Statusdatei gesperrt zu halten, wird nicht mehr unterstützt.	20. Mai 2025
Erste Veröffentlichung	—	5. Januar 2024

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.