Umplattformierung von COTS und internen Anwendungen während einer Migration in die Cloud AWS

AWS Prescriptive Guidance

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

AWS Prescriptive Guidance: Umplattformierung von COTS und internen Anwendungen während einer Migration in die Cloud AWS

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
Gezielte Geschäftsergebnisse 3
Wahl der Replatforming-Umgebung 4
Umplattformierung von Anwendungskomponenten, die auf nicht unterstützten Betriebssystemen ausgeführt werden OSs 5
Ersetzen von nicht unterstützten Servern oder Anwendungsservern OSs 6
Aktualisierung des Betriebssystems für COTS-Anwendungen 6
Aktualisierung des Betriebssystems für interne Anwendungen 6
Umgestaltung von Anwendungsbibliotheken und abhängiger Software 7
Neuplattformierung von Backend-Datenbanken 8
Umplatformung von Backend-Datenbanken für COTS-Anwendungen 8
Umplattformierung von Backend-Datenbanken für interne Anwendungen 9
Umplattformierung von Dateifreigaben 10
Aktualisierung der Protokollierungs- und Überwachungskomponenten 11
Testen und Validieren Ihrer Anwendungen 12
Automatisieren des laufenden Betriebssystem-Patchings 13
Verwendung von Automatisierungstools und Infrastruktur als Code (IaC) 13
Häufig gestellte Fragen 15
Wann sollte ich mich für eine Replattform statt für ein Rehosting entscheiden? 15
Kann ich meine COTS-Anwendung auf eine Open-Source-Datenbank umstellen? 15
Was AWS Tools, mit denen ich meine Server schnell auf folgende Server umhosten kann AWS Wolke? 15
Was AWS Tools, mit denen ich meine Anwendungen auf eine neue Plattform umstellen kann? 16
Ressourcen 17
Referenzen 17
Tools 17
Dokumentverlauf 18
..... xix

Umplattformierung von COTS und internen Anwendungen während einer Migration auf die AWS Cloud

Anbu Selvan, Amazon Web Services ()AWS

März 2021 (Dokumentenverlauf)

In diesem Leitfaden werden sieben Bereiche beschrieben, auf die Sie sich konzentrieren sollten, wenn Sie kommerzielle Standardanwendungen (COTS) und interne Anwendungen in der Amazon Web Services (AWS) Cloud neu plattformieren. Der Leitfaden bietet auch Strategien, Tools und AWS Services, die Sie bei der Neuplattformierung von Anwendungskomponenten unterstützen. COTS-Anwendungen sind vorgefertigte Anwendungen von Drittanbietern, die auf einem kommerziellen Markt erworben werden können (z. B. AWS Marketplace). In-house Anwendungen werden intern von Ihrem Unternehmen entwickelt und verwendet.

Nachdem Sie sich für die Migration Ihrer COTS- oder internen Anwendungen auf die entschieden haben AWS Cloud, müssen Sie abwägen, welche der sieben gängigen Migrationsstrategien (7 Rs) Sie verwenden sollten. Bei diesen Strategien handelt es sich um Refactoring, Replatform, Repurchase, Rehosting, Relocation, Retention und Außerbetriebnahme. Wir empfehlen Ihnen, Anwendungen, die Komponenten oder Datenbanken verwenden, deren Support-Ende (EOS) erreicht oder kurz davor steht, diese Plattform neu zu platzieren. EOS liegt vor, wenn ein Anbieter den technischen Support für ein Produkt zurückzieht. Wenn Sie sich dafür entscheiden, eine Anwendung in der AWS Cloud neu zu platzieren, können Sie von den folgenden Funktionen profitieren:

- Automatisieren Sie direkte Betriebssystem-Upgrades (OS) mit. AWS Systems Manager
- Verwenden Sie Snapshot-Speichervolumes, um schnell Amazon Machine Images (AMIs) aus Amazon Elastic Compute Cloud (Amazon EC2) -Instances zu erstellen.
- Erstellen Sie ein privates Subnetz, um Workloads zu isolieren, die auf veralteten Betriebssystemen (OS) ausgeführt werden.
- Verwenden Sie Hochgeschwindigkeitsnetzwerke, um Produktionsumgebungen schnell zu replizieren und die Platformierung zu testen.
- Richten Sie schnell einen separaten Anwendungsstapel mit On-Demand-EC2-Instances ein, ohne zusätzliche lokale Hardware zu verwenden.

Um von diesen und anderen in der AWS Cloud verfügbaren Funktionen zu profitieren, empfehlen wir Ihnen, Ihre Anwendung zunächst mithilfe von neu zu hosten. AWS Transform MGN Anschließend können Sie die Anwendung in der AWS Cloud aktualisieren. Die folgende Liste enthält Beispiele dafür, wann eine Anwendung auf eine neue Plattform umgestellt werden sollte:

- Für das Betriebssystem, die Laufzeiten (z. B. Apache Tomcat, JBoss oder Oracle WebLogic Server), Datenbanken oder Laufzeitkomponenten (z. B. Java, Python oder Perl) der Anwendung ist keine Support mehr verfügbar.
- Die Anwendung muss widerstandsfähiger werden und nach Ausfällen (z. B. Softwarefehlern oder Infrastrukturproblemen) automatisch wiederhergestellt werden.
- Neue Anwendungsfunktionen sind für neue Kundensegmente oder zur Unterstützung erhöhter Lasten erforderlich.
- Die Anwendung ist instabil und muss verbessert werden, um die Betriebsstabilität zu erhöhen.

Bevor Sie mit der Umstellung der Plattform beginnen, sollten Sie nach Alternativen zu den Funktionen Ihrer Anwendung suchen. Prüfen Sie beispielsweise, ob Sie diese durch eine SaaS-Lösung (Software as a Service) eines unabhängigen Softwareanbieters (ISV) ersetzen können. Möglicherweise können Sie Anwendungsfunktionen auch mithilfe von AWS Diensten wie Amazon Cognito AWS Lambda, Amazon MQ AWS Glue, Amazon Quick oder Amazon Aurora neu aufbauen.

Dieser Leitfaden richtet sich an IT-Administratoren, Anwendungsinhaber, Architekten, technische Leiter und Projektmanager. Der Leitfaden enthält die folgenden sieben Bereiche, auf die Sie sich bei der Neuplattformierung von COTS und internen Anwendungen in der AWS Cloud konzentrieren sollten:

- Wahl der Replatforming-Umgebung
- Umplattformierung von Anwendungskomponenten, die auf nicht unterstützten Betriebssystemen ausgeführt werden OSs
- Neuplattformierung von Backend-Datenbanken
- Umplattformierung von Dateifreigaben
- Aktualisierung der Protokollierungs- und Überwachungskomponenten
- Testen und Validieren Ihrer Anwendungen
- Automatisieren des laufenden Betriebssystem-Patchings

Gezielte Geschäftsergebnisse

Nach der Umstellung von COTS und internen Anwendungen in der Cloud sollten Sie mit den folgenden vier Ergebnissen rechnen: AWS

- Reduzieren Sie die Sicherheitsrisiken, die durch ältere Anwendungen entstehen, auf denen Software oder Betriebssysteme ausgeführt werden, die nicht unterstützt werden.
- Senken Sie Ihre Gesamtbetriebskosten für Anwendungen, indem Sie teure, überflüssige Datenbankeditionen entfernen oder Open-Source-Datenbanken einsetzen.
- Reduzieren Sie den betrieblichen Aufwand, indem Sie AWS verwaltete Datenbanken (z. B. Amazon Relational Database Service (Amazon RDS) oder Aurora) verwenden, um eine höhere Verfügbarkeit und Zuverlässigkeit Ihrer Anwendungen zu erreichen.
- Machen Sie ältere Anwendungen widerstandsfähiger, indem Sie Cloud-native Automatisierungs- und Überwachungsfunktionen wie CloudWatchAmazon-Überwachung oder Manager-based Systembetriebssystem-Patching einsetzen.

Wahl der Replatforming-Umgebung

Wir empfehlen, dass Sie eine Anwendung in der AWS Cloud neu plattformieren, um Funktionen wie Amazon Elastic Block Store (Amazon EBS) -Snapshots oder das Klonen einer EC2-Instance zu verwenden, um ein AMI zu erstellen. Diese Funktionen unterstützen Ihren Upgrade- und Testprozess.

In der Regel beginnen Sie mit dem Rehosten der Anwendung in der AWS Cloud und beginnen dann mit dem Replatforming-Prozess. Sie müssen jedoch vorsichtig sein, wenn Sie Anwendungen migrieren, die Sicherheitslücken aufweisen oder nicht unterstützte Software und Betriebssysteme ausführen. Diese Anwendungen können Sicherheitslücken aufdecken, die für Ihre Migration und future Abläufe gefährlich sind. Stattdessen empfehlen wir, MGN zu verwenden, um die Anwendungskomponenten in ein privates Subnetz mit begrenztem Ausgangsvolumen zu replizieren. Bei diesem Ansatz werden Ihre Workloads isoliert, sodass Sie sie dann sicher auf eine neue Plattform verschieben und testen können, bevor sie in einer Produktionsumgebung bereitgestellt werden.

Ihr Unternehmen hat möglicherweise erweiterten Betriebssystemsupport von einem ISV erworben, der das Patchen von Betriebssystemen über das offizielle EOS-Datum hinaus um einen Zeitraum von drei bis fünf Jahren verlängert. Dies ist eine vorübergehende Maßnahme und bietet zusätzliche Zeit, um Ihre Produkt-Upgrade- oder Migrationszeitpläne umzugestalten oder an den Zeitplan für die Produktveröffentlichung eines COTS-Anwendungsherstellers anzupassen. Wir empfehlen jedoch, AWS Tools und Fachwissen von AWS Professional Services oder AWS-Migrationskompetenzpartnern zu verwenden, um diese Workloads auf neuere Betriebssysteme umzustellen. Dies hilft, teure erweiterte Lizenzverträge zu vermeiden und bedeutet, dass Sie Ihre Migration schneller abschließen können.

Umplattformierung von Anwendungskomponenten, die auf nicht unterstützten Betriebssystemen ausgeführt werden OSs

Der Ansatz zur Neuplatformierung von Anwendungskomponenten, die auf nicht unterstützten Betriebssystemen ausgeführt werden, OSs ist für jede Anwendungskomponente unterschiedlich. In der folgenden Tabelle sind die Replatforming-Optionen zusammengefasst, die für Anwendungskomponenten verfügbar sind, die EOS erreicht haben.

Anwendungskomponente	Lösung für COTS-Anwendungen	Lösung für interne Anwendungen
Anwendungsserver	Führen Sie ein Upgrade auf die vom Hersteller der Anwendung empfohlene Version durch.	Identifizieren Sie die neueste Version des Anwendungsservers. Erstellen und validieren Sie es in einer Entwicklungsumgebung, bevor Sie das Upgrade durchführen.
OS	Führen Sie ein Upgrade auf die vom Hersteller der Anwendung empfohlene Version durch.	Identifizieren Sie die neueste Betriebssystemversion. Erstellen und validieren Sie es in einer Entwicklungsumgebung, bevor Sie das Upgrade durchführen.
Laufzeit-Bibliotheken	Führen Sie ein Upgrade auf die vom Hersteller der Anwendung empfohlene Version durch.	Führen Sie ein Upgrade durch und validieren Sie es mit der neuesten Version.
Andere Anwendungskomponenten	Fordern Sie neue Anwendungsbinärdateien vom Anbieter der Anwendung an.	Verwenden Sie die neuesten Versionen des Betriebssystems, der Laufzeit und des Anwendungsservers.

In den folgenden Abschnitten finden Sie weitere Informationen zu Umplattform-Ansätzen für Ihre Anwendungskomponenten.

Ersetzen von nicht unterstützten Servern oder Anwendungsservern OSs

Wenn Sie Anwendungsserver ersetzen, die nicht unterstützt werden (z. B. Apache Tomcat 6.0, Apache 2.2 oder IIS 7.x), ist für Ihre neuen Anwendungsserverversionen möglicherweise ein Betriebssystem-Upgrade erforderlich. Am meisten nicht unterstützt OSs werden Red Hat Enterprise Linux (RHEL) Versionen 5 und 6, CentOS Versionen 5 und 6 oder Windows 2008 R2. Sie sollten die folgenden Schritte für Anwendungen implementieren, auf denen diese ausgeführt werden: OSs

1. Starten Sie eine EC2-Instance mit der erforderlichen Betriebssystemversion.
2. Installieren Sie die erforderliche Version des Anwendungsservers.
3. Es gibt zwei unterschiedliche Ansätze für interne Anwendungen und COTS-Anwendungen:
   - Interne Anwendungen — Stellen Sie die Anwendung erneut auf der EC2-Instance bereit.
   - COTS-Anwendungen — Wenden Sie sich an den Anbieter der Anwendung und fordern Sie Anwendungsbinärdateien an, die für die erforderlichen Betriebssystem- oder Anwendungsserverversionen zertifiziert sind.

Aktualisierung des Betriebssystems für COTS-Anwendungen

Die meisten Anbieter von COTS-Anwendungen unterstützen Windows 2016 oder RHEL 7. Wenn Ihre ältere COTS-Anwendung Windows 2016 nicht unterstützt, empfehlen wir ein direktes Upgrade von Windows 2008 R2 auf Windows 2012 R2 mithilfe der von Microsoft bereitgestellten Option für das direkte Upgrade. Sie können auch AWS Systems Manager Automation-Runbooks verwenden, um Windows Server, das auf EC2-Instances ausgeführt wird, automatisch zu aktualisieren. Wir empfehlen, dass Sie sich an den Hersteller der Anwendung wenden und ihn bitten, seine Software für die neueste Betriebssystemversion zu zertifizieren.

Aktualisierung des Betriebssystems für interne Anwendungen

Wir empfehlen, dass Sie die Software Ihrer internen Anwendung kompilieren und neu erstellen, indem Sie die neuesten Betriebssystem- und Softwarelaufzeitversionen verwenden (z. B. Java, C ++, .NET oder Python). Anschließend können Sie die bestehende Anwendungsumgebung klonen,

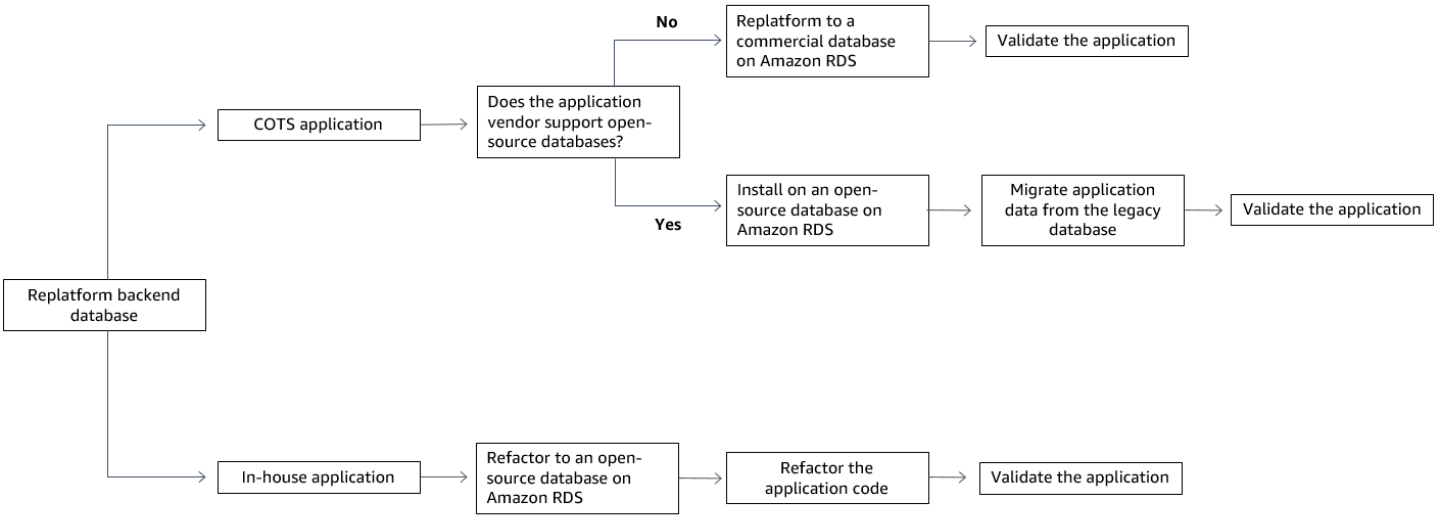
die Funktionalität manuell bereitstellen und validieren und Ihre Build-Umgebung auf das neueste Betriebssystem, die neuesten Runtime-Softwarekomponenten und Bibliotheken aktualisieren, bevor Sie ein Upgrade auf Ihre Produktionsumgebung durchführen.

Umgestaltung von Anwendungsbibliotheken und abhängiger Software

Der Ansatz für die Neuplattformierung von Anwendungsbibliotheken und abhängiger Software ähnelt dem Ansatz für, Sie aktualisieren OSs jedoch nur die Bibliotheken. Anschließend testen Sie die Funktionalität der Anwendung und replizieren die erforderlichen Bibliotheken auf Ihren Vorproduktions- und Produktionsservern. In der Regel kümmert sich der Anbieter der COTS-Anwendung um die erforderlichen Updates für die Anwendungskomponenten im Rahmen seiner laufenden Softwareversionen.

Neuplattformierung von Backend-Datenbanken

Der Ansatz für die Neuplattformierung von Backend-Datenbanken unterscheidet sich für COTS und interne Anwendungen. Dies liegt daran, dass der Quellcode normalerweise nur für interne Anwendungen verfügbar ist. Die folgende Abbildung zeigt die Replatforming-Optionen, die für die Backend-Datenbanken Ihrer Anwendung verfügbar sind.



In den folgenden Abschnitten werden die Umplattforming-Ansätze für Backend-Datenbanken, die zu COTS oder internen Anwendungen gehören, erläutert.

Umplatformung von Backend-Datenbanken für COTS-Anwendungen

Wir empfehlen Ihnen, eine Aurora-Datenbank zu verwenden, wenn Ihre COTS-Anwendung Open-Source-Datenbanken unterstützt. Die Verwendung einer Open-Source-Datenbank trägt zur Senkung der Lizenzkosten bei, und Sie können auch Tools wie AWS Schema Conversion Tool (AWS SCT) und AWS Database Migration Service (AWS DMS) verwenden, um eine Umstellung mit minimalen Ausfallzeiten während Ihrer Migration zu erreichen.

Wenn Ihre COTS-Anwendung keine Open-Source-Datenbanken unterstützt, empfehlen wir einen Umstieg auf eine kommerzielle Datenbank auf Amazon Relational Database Service (Amazon RDS) wie Amazon RDS for Oracle oder Amazon RDS for Microsoft SQL Server. Sie sollten die von Ihrer Anwendung verwendeten Datenbankfunktionen bewerten und sicherstellen, dass sie in Amazon RDS

unterstützt werden, bevor Sie mit der Migration beginnen. Weitere Informationen finden Sie unter Grenzwerte für Microsoft SQL Server-Datenbank-Instances in der Amazon RDS-Dokumentation.

Sie können auch Ihre verbleibenden Datenbanklizenzen verwenden und selbstverwaltete kommerzielle Datenbanken auf EC2-Instances ausführen. Wenn Sie sich für diesen Ansatz entscheiden, empfehlen wir Ihnen, den Lizenzverifizierungsprozess mit dem Anbieter Ihrer Datenbank zu beginnen. Nachdem der Lizenzverifizierungsprozess abgeschlossen ist, sollten Sie eine selbstverwaltete Datenbanklösung auf Amazon EC2 entwerfen, die den für Ihre Anwendung erforderlichen Recovery Time Objective (RTO) oder Recovery Point Objective (RPO) entspricht.

Schließlich empfehlen wir, sicherheitssensible, leistungsstarke COTS-Anwendungen, die SQL Server-Datenbanken verwenden, auf SQL Server umzustellen, die auf Amazon EC2 EC2-Linux-Instances ausgeführt werden. Weitere Informationen dazu finden Sie unter Migration Ihrer lokalen SQL Server-Windows-Workloads zu Amazon EC2 Linux.

Umplattformierung von Backend-Datenbanken für interne Anwendungen

Sie können Ihre Datenbanklizenzkosten senken und die Skalierbarkeit erhöhen, indem Sie die Backend-Datenbanken Ihrer internen Anwendung auf AWS verwaltete Datenbanken umstellen (z. B. Amazon RDS for PostgreSQL, Amazon RDS for MySQL, Aurora oder Amazon DynamoDB).

Mit AWS-verwalteten Datenbanken können Sie wiederkehrende Verwaltungsaufgaben für Ihre Datenbanken reduzieren (z. B. das Durchführen von Backups oder das Patchen von Datenbanken und Betriebssystemen). Wenn Sie Amazon Multi-AZ RDS-Bereitstellungen verwenden, können Sie auch die Verfügbarkeit Ihrer Anwendung erhöhen, indem Sie Ausfälle aufgrund von Datenbankhardwarefehlern verhindern. Multi-AZ Datenbanken werden kontinuierlich in eine andere Availability Zone repliziert, und die Anwendung führt bei Ausfällen einen transparenten Failover auf die replizierte Datenbank durch.

Sie können kommerzielle Datenbanken verwenden AWS DMS und AWS SCT in Aurora und Amazon RDS konvertieren. AWS SCT automatisiert den Konvertierungsprozess des Datenbankschemas und AWS DMS ermöglicht die Datenreplikation von lokalen Datenbanken nach Amazon RDS. AWS DMS trägt außerdem dazu bei, Ausfallzeiten auf ein Minimum zu reduzieren, wenn Sie lokale Anwendungen in die Cloud migrieren. AWS

Umplattformierung von Dateifreigaben

Wenn Ihre internen Anwendungen lokale externe Dateifreigaben wie Network File Systems (NFS) oder Server Message Block (SMB) verwenden, empfehlen wir, den Code der Anwendung so zu ändern, dass er Amazon Simple Storage Service (Amazon S3) und Amazon verwendet. CloudFront Wenn jedoch blockbasierter Zugriff erforderlich ist, können Sie Amazon Elastic File System (Amazon EFS) oder Amazon FSx for Lustre verwenden. Amazon EFS und Amazon FSx sind AWS Optionen für Anwendungen, die gemeinsam genutzten Dateispeicher benötigen.

Für COTS-Anwendungen empfehlen wir, Daten von lokalen Dateifreigaben zu Amazon EFS unter Linux oder FSx für Windows File Server zu migrieren. Sie sollten die blockbasierten Dateifreigaben für die Anwendung genauso verfügbar machen wie lokale Dateifreigaben. Amazon EFS und Amazon FSx bieten NFS- und SMB-basierte Dateifreigaben mit denselben Funktionen wie lokale Dateifreigaben. Um Amazon EFS oder Amazon FSx für Dateifreigaben zu verwenden, sind nur Konfigurationsänderungen für COTS-Anwendungen erforderlich.

Aktualisierung der Protokollierungs- und Überwachungskomponenten

Einige ältere Umgebungen verwenden zentralisierte Protokollierungs- und Überwachungstools (z. B. Splunk oder Zabbix) für die Infrastruktur- und Anwendungsüberwachung. SolarWinds Anwendungsunterstützungsteams können auch das Secure Shell (SSH) -Protokoll oder das Remote Desktop Protocol (RDP) zur Überwachung und zum Debuggen verwenden. Um diesen manuellen und sich wiederholenden Prozess zu vermeiden, können Sie die Überwachung in der CloudWatchCloud automatisieren. AWS

Wir empfehlen, CloudWatch Metriken zur Überwachung Ihrer Infrastruktur und CloudWatch Agenten zu verwenden, an die Anwendungsprotokolle gesendet werden sollen CloudWatch. Nachdem die Anwendungsprotokolle bei eingegangen sind CloudWatch, können Sie CloudWatchMetrikfilter erstellen und CloudWatch Alarme verwenden, um Anwendungsfehler zu überwachen und die Support-Teams automatisch zu benachrichtigen.

CloudWatch bietet außerdem Tools zur Erstellung von operativen Dashboards für die laufende Überprüfung der Produktionsabläufe Ihrer Anwendungen. Zentralisierte Überwachungstools von Drittanbietern können in CloudWatch und andere AWS Dienste integriert werden. Auf diese Weise können Sie Ihre bestehenden Betriebspraktiken auf Infrastrukturen und Anwendungen in der AWS Cloud ausdehnen. Möglicherweise müssen Sie jedoch Ihre Support- oder Betriebsteams umschulen, wenn Sie Anwendungen in der AWS Cloud betreiben möchten. Weitere Informationen dazu finden Sie im Abschnitt Betriebsperspektive: Verwaltung und Skalierung des Whitepapers AWS Cloud Adoption Framework (AWS CAF).

Testen und Validieren Ihrer Anwendungen

Funktions- und Leistungstests sind ein wichtiger Teil des Umstellungsprozesses einer Anwendung. In der Regel verlassen sich ältere Anwendungen beim Testen auf das Wissen eines Anwendungsbesitzers, da funktionale Details nicht korrekt oder vollständig dokumentiert sind. Wir empfehlen jedoch, Anwendungsfälle von Anwendungen mithilfe von verhaltensbezogenen und automatisierten Tests aufzuzeichnen. Mit diesem Ansatz wird die Funktionalität einer Anwendung vor und nach der Platformierung schnell und zuverlässig validiert. Sie können automatisierte Testtools (z. B. Selenium, Tricentis oder Gatling) verwenden, um Funktions- und Leistungstests zu erstellen. Ein Basisergebnis muss durch die Ausführung von Funktions- und Leistungstests in Ihrer aktuellen Anwendungsumgebung generiert werden. Die Testergebnisse zwischen der aktuellen Anwendungsumgebung und der Zielanwendungsumgebung können verglichen und als Akzeptanzkriterien verwendet werden.

Wir empfehlen die Verwendung von Canary-Tests für kundenorientierte Anwendungen. Canary Testing testet regelmäßig kritische Anwendungsworkflows in der Produktionsumgebung und benachrichtigt die Support-Teams über Fehler. Weitere Informationen finden Sie im Abschnitt zur Bereitstellung auf Canary des AWS Well-Architected Framework.

Automatisieren des laufenden Betriebssystem-Patchings

Ältere Anwendungen in lokalen Rechenzentren verlassen sich häufig auf manuelle Betriebsprozesse für fortlaufende Betriebssystem-Patches und Softwareupdates. Während Ihrer Umstellung auf die Plattform empfehlen wir Ihnen, das Betriebssystem-Patching mithilfe von Systems Manager Patch Manager oder anderen automatisierten Patching-Prozessen zu automatisieren. Patch Manager bietet einen zentralisierten und konsistenten Prozess zur Erfassung betrieblicher Erkenntnisse und zur Implementierung routinemäßiger betrieblicher Aufgaben sowohl in der AWS Cloud als auch in lokalen Ressourcen.

Wir empfehlen, Entwicklungsumgebungen vor dem für Produktionsumgebungen üblichen Patch-Zeitfenster zu patchen. Weitere Informationen dazu finden Sie im Patch Manager-Runbook zur Automatisierung des Betriebssystem-Patchings. Außerdem sollten Sie Canary-Tests einsetzen, um wichtige Anwendungsfunktionalitäten in Vorproduktions- oder Produktionsumgebungen regelmäßig zu testen und die Support-Teams zu benachrichtigen, falls die Tests fehlschlagen. Dies hilft, ungeplante Ausfälle Ihrer Anwendung zu vermeiden.

Verwendung von Automatisierungstools und Infrastruktur als Code (IaC)

Im Rahmen der Umstellung Ihrer Anwendung sollten Sie Plattform-Builds mithilfe von Konfigurationsmanagement-Tools wie Chef, Puppet oder Ansible automatisieren. Diese Tools ermöglichen eine wiederholbare Erstellung des Anwendungsstapels und formalisieren die Schritte zur Generierung einer Anwendungsinstanz, einschließlich der Konfiguration des Stacks.

Wir empfehlen, dass Sie Ihre Infrastruktur mithilfe von IaC-Best Practices bereitstellen. Dafür stehen mehrere Optionen zur Verfügung AWS Cloud Development Kit (AWS CDK), darunter CloudFormationTerraform. Chef, Ansible und Puppet verfügen ebenfalls über begrenzte Funktionen, die möglicherweise ausreichend Automatisierung für Ihren Anwendungsfall bieten.

Wiederholbare Builds, die IaC und Konfigurationsmanagement-Code verwenden, helfen Ihnen beim Testen der Infrastruktur, ohne den Aufwand und das Risiko einer Neuerstellung dieser Ressourcen. Das Patchen und Aktualisieren einer vorhandenen Instanz kann zu einem Zustand führen, der es schwierig macht, Probleme zu reproduzieren und zu identifizieren.

Wenn eine COTS-Anwendung keine automatische Installation unterstützt, empfehlen wir, sich an das AWS-Partnernetzwerk (APN) zu wenden. Weitere Informationen dazu finden Sie im Abschnitt Plattformperspektive: Anwendungen und Infrastruktur des AWS CAF-Whitepapers.

Häufig gestellte Fragen zur Umstellung von COTS und internen Anwendungen

Dieser Abschnitt enthält Antworten auf häufig gestellte Fragen zur Umstellung von COTS und internen Anwendungen.

Wann sollte ich mich für eine Replattform statt für ein Rehosting entscheiden?

Sie sollten Komponenten, die keine Datenbanken sind, auf eine neue Plattform verlagern, wenn der Betriebssystemanbieter Sicherheitsupdates eingestellt hat (z. B. RHEL 5 und Windows 2008). Bei nicht unterstützten Betriebssystemen können Sie Komponenten in einem isolierten privaten Subnetz rehosten und sie aktualisieren, bevor Sie sie in einer Produktionsumgebung bereitstellen.

Bei Datenbankkomponenten sollten Sie prüfen, ob ein Umstieg auf eine kommerzielle Datenbank auf Amazon RDS eine angemessene Wahl ist. Sie können auch eine selbstverwaltete Datenbank verwenden, die auf Amazon EC2 läuft, wenn der Anbieter der Datenbank Ihnen erlaubt, Ihre aktuelle Lizenz in der AWS Cloud zu verwenden.

Kann ich meine COTS-Anwendung auf eine Open-Source-Datenbank umstellen?

Es ist möglich, eine COTS-Anwendung in eine Open-Source-Datenbank umzuwandeln, wenn der Anbieter Ihrer Anwendung Open-Source-Datenbanken unterstützt. Wir empfehlen nicht, Open-Source-Datenbanken für COTS-Anwendungen zu verwenden, ohne den Support des Anbieters einzuholen.

Was AWS Tools, mit denen ich meine Server schnell auf folgende Server umhosten kann AWS Wolke?

Sie können MGN als Ihren primären Migrationsservice verwenden, um Ihre Anwendungen hochzuschalten und zu AWS verlagern.

Was AWS Tools, mit denen ich meine Anwendungen auf eine neue Plattform umstellen kann?

In der AWS Cloud sind mehrere Tools verfügbar, darunter:

- Automatisierte Upgrades von Systems Manager — Aktualisieren Sie Windows 2008 und SQL Server 2008 in der AWS Cloud.
- AWS SCT— Konvertiert das Datenbankschema von kommerziellen in Open-Source-Datenbanken.
- AWS DMS— Replizieren Sie Daten aus lokalen Datenbanken in die AWS Cloud, entweder in Amazon RDS-Datenbanken oder in Datenbanken, die auf Amazon EC2 laufen.

Ressourcen

Referenzen

- AWS Cloud Adoption Framework aus dem Whitepaper zur Migration AWS
- Erstellen Sie ein Dashboard CloudWatch
- Grenzwerte für Microsoft SQL Server-DB-Instances
- Migrieren Sie eine selbstverwaltete Microsoft SQL Server-Datenbank zu einer vollständig verwalteten Datenbank auf Amazon RDS
- AWS Workshop: Microsoft SQL Server-Failover-Clusterinstanz auf Amazon FSx
- Aktualisieren Sie das Betriebssystem der Amazon EC2 EC2-Windows-Serverinstanz

Tools

- AWS CodeBuildund AWS CodePipelinehelfen Ihnen beim Aufbau von Pipelines für kontinuierliche Integration und kontinuierliche Entwicklung (CI/CD) in der AWS-Cloud.
- AWS DataSync— DataSync hilft bei der Replikation von lokalen Dateispeichern in die Cloud. AWS
- AWS DMS— AWS Database Migration Service (AWS DMS) hilft bei der Replikation von Daten aus lokalen Datenbanken in Amazon RDS-Datenbanken oder Datenbanken auf Amazon EC2.
- AWS Systems Manager basierte automatisierte Upgrades helfen bei der Aktualisierung von Windows 2008 und SQL Server 2008 in der Cloud. AWS
- AWS SCT— AWS Schema Conversion Tool hilft bei der Konvertierung von Datenbankschemas von kommerziellen Datenbanken in Open-Source-Datenbanken.
- MGN — AWS Transform MGN ist eine hochautomatisierte Lift-and-Shift-Lösung (Rehost), die die Migration von Anwendungen zu vereinfacht, beschleunigt und die Kosten senkt. AWS
- Der Windows to Linux Replatforming Assistant für Microsoft SQL Server-Datenbanken ist ein Tool zur Plattformumstellung.

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
Aktualisierter AWS-Servicename	Der Service für die CloudEndure Migration zum Anwendungsmigrationsservice wurde aktualisiert	12. Mai 2022
—	Erste Veröffentlichung	12. März 2021

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.