

KNIME Server auf AWS Marketplace

KNIME AG, Zürich, Schweiz

Version 4.18 (letzte Aktualisierung auf)



Inhaltsverzeichnis

Einleitung	Weiter lesen.
Bereitstellung auf AWS	Weiter lesen.
Voraussetzungen	Weiter lesen.
.	Weiter lesen.
Vorinstallierte Software (AWS)	Weiter lesen.
Gestützte Regionen	Weiter lesen.
Vorinstallierte Software	Weiter lesen.
Erweiterungen, die mit der AWS	Weiter lesen.
Optionale externe Abhängigkeiten	Weiter lesen.
Veröffentlichungshinweise	Weiter lesen.
.	Weiter lesen.
.	Weiter lesen.
.	Weiter lesen.
.	Weiter lesen.
Architektur Überblick	Weiter lesen.
KNIME Server Small/Medium	Weiter lesen.
KNIME Server Large (AWS)	Weiter lesen.
Sicherheit	Weiter lesen.
Audit Trail	Weiter lesen.
Zugang zu KNIME Server Instanz	Weiter lesen.
.	Weiter lesen.
Authentifizierung mit AWS	Weiter lesen.
.	Weiter lesen.
Gespeicherte Geheimnisse	Weiter lesen.
EC2 Sicherheitsgruppen	Weiter lesen.
Datenverschlüsselung	Weiter lesen.
Tag der Ressourcen	Weiter lesen.
Datenstellen.	Weiter lesen.
Permanente Daten	Weiter lesen.
Ephemere Daten	Weiter lesen.
Dimensionierung	Weiter lesen.
KNIME Server Small/Medium	Weiter lesen.
KNIME Server Large	Weiter lesen.

[Dimensionierung \(AWS\)](#page20)

[EG2](#page20)

[EBS-Volumenauswahl](#page21)

[Kosten](#page22)

[Softwarepreise](#page22)

[Hardware-Preis](#page22)

[KNIME Server Small](#page22)

[Erforderliche Dienste \(Server\)](#page22)

[Bereitstellung](#page23)

[KNIME Serverinstallation](#page23)

[Prüfung der Bereitstellung](#page23)

[Verbindung über den Browser](#page23)

[Verbindung über die Anaconda](#page23)

[Prüfung der Workflow-Ausführung](#page23)

[Empfohlene AWS-Bereitstellung](#page24)

[Empfohlene AWS-Bereitstellung](#page27)

[Standard KNIME Server-Pass](#page27)

[Lizenzdatei auf KNIME Server](#page27)

[Template-Bereitstellung](#page29)

[Templat-Engine](#page29)

[Vorlage VM build](#page29)

[Packerschritte für KNIME Server](#page30)

[Packerschritte für KNIME Server](#page30)

[AWS CloudFormation Template](#page30)

[Erstellung einer CloudFormation](#page31)

[KNIME Ausführen auf AWS](#page32)

[Einrichtung des VPC](#page32)

[CloudFormation Template](#page33)

[KNIME beenden Ausführung](#page34)

[Operationen](#page35)

[Anwendungsfehler](#page35)

[AZ-Fehler](#page36)

[Gerichtsfehler](#page36)

[Speicherkapazität](#page36)

[Sicherheitsbeschleunigung](#page36)

[Sicherung und Erholung](#page37)

[Backup](#page37) [Erholung.](#page37)
[Backup \(AWS\)](#page37) [Erholung \(AWS\).](#page37)
[Routine Wartung.](#page38) [Beginn der KNIME Server-](#page38)
[KNIME stoppen Server](#page38) [Neustart von KNIME Server](#page38)
[Bootnote für ältere Versionen](#page38) [Neustart des Ausführenden](#page38)
[Neustart des Ausführenden](#page39) [Verwaltung von Zertifikaten](#page39)
[Standardzertifikate](#page39) [Aktualisierung der Python-Ko](#page41)
[Bewerben Sie Betriebssystem](#page41) [Update KNIME Server](#page41)
[SSH-Zugriff auf KNIME Server](#page45) [Ändern der Instanzanfe](#page46)
[Aktualisierung/Abbau von](#page46) [Erhöhung der WorkflowRe](#page47)
[Schlüsseldrehung](#page47) [Servicelimits](#page47)
[Notunterhaltung](#page48) [Notfallpflege \(AWS\)](#page48)
[AZ Erholung](#page48) [Region Erholung.](#page48)
[Fehlerbehebung](#page49) [Unterstützung](#page50)
[Finden Sie Ihre Produkt](#page50) [Unterstützungskosten](#page50)

Einleitung

KNIME Server ist die Enterprise-Software für teambasierte Zusammenarbeit, Automatisierung, Management und Bereitstellung von Datenwissenschafts-Workflows, Daten und geführte Analytik. Nicht verfügbar Experten erhalten über KNIME WebPortal Zugriff auf die Datenwissenschaft oder können REST APIs nutzen, um Integration von Workflows als analytische Dienste in Anwendungen und IoT-Systeme. Eine vollständige Übersicht verfügbar [Hier](#).

Für einen Überblick über Anwendungsfälle finden Sie unsere [Lösungen Seite](#). Präsentationen auf den KNIME Summits über die Nutzung des KNIME Servers finden Sie [Hier](#).

Weitere Informationen

Wenn Sie auf der Suche nach detaillierten Erläuterungen zu den zusätzlichen Konfigurationsoptionen für KNIME Server, Sie können die [KNIME Leitfaden für die Verwaltung von Servern](#).

Wenn Sie den KNIME Server installieren möchten, sollten Sie zuerst die [KNIME Server Installationsanleitung](#).

Für Anleitungen zum Anschluss an KNIME Server von der KNIME Analytics Platform oder mit KNIME WebPortal verweist auf folgende Anleitungen:

- [KNIME Benutzerhandbuch des Servers](#)
- [KNIME WebPortal Benutzerhandbuch](#)

Eine zusätzliche Ressource ist auch die [KNIME Server Advanced Setup Guide](#).

Einsatz auf AWS

KNIME Server kann über den AWS Marketplace gestartet werden.

- [KNIME Server Medium \(PAYG\)](#)

Für eine vollständige Liste der Produktangebote, einschließlich der KNIME Analytics Platform, siehe [Hier](#).

KNIME Server Medium ist eine einzige AMI-Instanz und wird am einfachsten über die AWS gestartet Konsole. Wenn Sie mit dem AWS CLI vertraut sind, können Sie diese Bereitstellungsmethode auch verwenden.

Für Self-build-Einstellungen mit einem benutzerdefinierten Basisbild, sollten Sie die [KNIME Server Installationsanleitung](#).

Voraussetzungen

Die Person, die für die Bereitstellung von KNIME Server verantwortlich ist, sollte grundsätzlich vertraut sein

AWS-Funktionalität um EC2-Instanzen zu konfigurieren. KNIME Serververwaltung

erfordert grundlegende Linux-Systemverwaltungskompetenzen, wie die Bearbeitung von Textdateien über den CLI, und Start/Stop-Systemdienste.

KNIME Server Small, Medium und BYOL sind einzelne AMI-Bilder und enthalten alle Software Anforderungen.

Für selbst aufbauende Fälle konsultieren Sie bitte den Standard [KNIME Installationsanleitung für Server](#) .

AWS-Mittel

Die Einführung einer Instanz erfordert ein VPC und Subnet. Die Standard-Sicherheitsgruppe aktiviert HTTP-Zugriff auf Port 80 und HTTPS-Zugriff auf Port 443. SSH-Zugang zur Verwaltung der Server auf Port 22.

Weitere AWS-Dienste, die gegebenenfalls verwendet werden können, sind:

- [AWS KMS](#) : Zur Unterstützung der EBS/EFS-Volumenverschlüsselung.
- [AWS EFS](#) : Elastisches Dateisystem kann verwendet werden, um Server Failover-Setup zu unterstützen.
- [AWS ELB](#) : Load-balancer kann optional verwendet werden, um Server Failover-Setup zu unterstützen.
- [AWS ASG](#) : Auto-Scaling-Gruppe, die im Server Failover-Setup verwendet wird, oder um verteilt zu unterstützen
KNIME Ausführungsbeispiele für KNIME Server Large.

Vorinstallierte Software (AWS Specific)

Zur Bequemlichkeit haben wir installiert und vorkonfiguriert:

- [AWS CLI](#) .
- [Botanik](#) .

Unterstützte Regionen

Detailliertere Informationen über unterstützte Regionen sind verfügbar [Hier](#) .

Vorinstallierte Software

Zur Bequemlichkeit haben wir installiert und vorkonfiguriert:

- OpenJDK 11 (erforderlich)
- [Anaconda](#)
- R
- Chroniken (Um sicherzustellen, dass die Systemuhr synchronisiert wird)
- Postfix (Um KNIME Server zu ermöglichen, E-Mail-Benachrichtigungen zu senden)
- iptables (Redirects of Requests on port 80, 443 to Tomcat running on port 8080, 8443)

Erweiterungen installiert mit Ausführenden

Um zusätzliche Erweiterungen zu installieren, siehe Abschnitt [\[install-extensions\]](#).

Erweiterungen kommen alle von den folgenden Update-Seiten:

- <https://update.knime.com/analytics-platform/5.4>
- <https://update.knime.com/community-contributions/trusted/5.4>

Folgende Erweiterungen werden installiert:

- [org.knime.features.activelearning.feature.group](#)
- [org.knime.features.cloud.aws.mlservices.feature.group](#)
- [org.knime.features.cloud.aws.athena.feature.group](#)
- [org.knime.features.cloud.aws.feature.group](#)
- [org.knime.features.cloud.aws.redshift.feature.group](#)
- [org.knime.features.cloud.aws.redshift.driver.feature.group](#)
- [org.knime.product.desktop](#)
- [org.knime.features.audio.feature.group](#)
- [org.knime.features.arima.feature.group](#)
- [org.knime.features.cloud.azure.feature.group](#)
- [org.knime.features.google.cloud.storage.feature.group](#)
- [org.knime.features.chem.types.feature.group](#)
- [org.knime.features.bigdata.connectors.feature.group](#)

- [org.knime.features.ext.chem.tools.feature.group](#)
- [org.knime.features.parquet.feature.group](#)
- [org.knime.features.datageneration.feature.group](#)
- [org.knime.features.database.feature.group](#)
- [org.knime.features.dl.keras.feature.group](#)
- [org.knime.features.dl.tensorflow.feature.group](#)
- [org.knime.features.dl.onnx.feature.group](#)
- [org.knime.features.ext.dl4j.feature.group](#)
- [org.knime.features.distmatrix.feature.group](#)
- [com.knime.features.enterprise.slave.feature.group](#)
- [org.knime.features.expressions.feature.group](#)
- [org.knime.features.kafka.feature.group](#)
- [org.knime.features.bigdata.spark.feature.group](#)
- [org.knime.features.bigdata.fileformats.feature.group](#)
- [org.knime.features.bigdata.spark.local.feature.group](#)
- [org.knime.features.ext.h2o.mojo.spark.feature.group](#)
- [org.knime.features.browser.chromium.feature.group](#)
- [org.knime.features.ext.exttool.feature.group](#)
- [org.knime.features.exttool.feature.group](#)
- [org.knime.features.base.filehandling.feature.group](#)
- [org.knime.features.scm.git.feature.group](#)
- [org.knime.features.ext.h2o.feature.group](#)
- [org.knime.features.ext.h2o.mojo.feature.group](#)
- [org.knime.features.ext.h2o.spark.feature.group](#)
- [org.knime.features.ext.birt.feature.group](#)
- [org.knime.features.ext.lucene.feature.group](#)
- [org.knime.features.r.feature.group](#)
- [org.knime.features.ext.itemset/latest\[org.knime.features.ext.itemset.feature.group](#)
- [org.knime.features.js.views.feature.group](#)
- [org.knime.features.js.views.labs.feature.group](#)

- [org.knime.features.ext.jfreechart.feature.group](#)
- [org.knime.features.jpmm1.feature.group](#)
- [org.knime.features.ext.jython.feature.group](#)
- [org.knime.features.mli.feature.group](#)
- [org.knime.features.database.connectors.sqlserver.driver.feature.group](#)
- [org.knime.features.database.extensions.sqlserver.driver.feature.group](#)
- [org.knime.features.microsoft.r.feature.group](#)
- [org.knime.features.base.pmml.feature.group](#)
- [org.knime.features.mongodb.feature.group](#)
- [org.knime.features.neighborgram.feature.group](#)
- [org.knime.features.network.feature.group](#)
- [org.knime.features.network.distmatrix.feature.group](#)
- [org.knime.features.base.widedata.feature.group](#)
- [org.knime.features.ext.osm.feature.group](#)
- [org.knime.features.optimization.feature.group](#)
- [org.knime.features.ext.perl.feature.group](#)
- [org.knime.features.ext.webservice.client.pilot.feature.group](#)
- [org.knime.features.js.plotly.feature.group](#)
- [org.knime.features.base.pmml2.feature.group](#)
- [org.knime.features.base.pmml.translation.feature.group](#)
- [org.knime.features.python2.feature.group](#)
- [org.knime.features.quickform.legacy.feature.group](#)
- [org.knime.features.ext.r.bin.feature.group](#)
- [com.knime.features.gateway.explorer.feature.group](#)
- [com.knime.features.gateway.remote.feature.group](#)
- [com.knime.features.reporting.designer.feature.group](#)
- [org.knime.features.rest.feature.group](#)
- [org.knime.features.ext.md.feature.group](#)
- [org.knime.features.ext.parsio.feature.group](#)
- [org.knime.features.semanticweb.feature.group](#)

- [com.knime.features.explorer.serverspace.feature.group](#)
- [org.knime.features.ext.spotfire.feature.group](#)
- [org.knime.features.stats2.feature.group](#)
- [org.knime.features.core.streaming.feature.group](#)
- [org.knime.features.ext.svg.feature.group](#)
- [org.knime.features.ext.tableau.bin.feature.group](#)
- [org.knime.features.ext.tableau.hyper.bin.feature.group](#)
- [org.knime.features.ext.tableau.hyper.feature.group](#)
- [org.knime.features.ext.tableau.feature.group](#)
- [com.knime.features.explorer.sharedspace.feature.group](#)
- [org.knime.features.ext.textverarbeitung.feature.group](#)
- [org.knime.features.ext.textverarbeitung.dl4j.feature.group](#)
- [org.knime.features.ext.socialmedia.feature.group](#)
- [org.knime.features.virtual.feature.group](#)
- [org.knime.features.buildworkflows.feature.group](#)
- [org.knime.features.ext.webservice.client.feature.group](#)
- [org.knime.features.webanalytics.feature.group](#)
- [org.knime.features.ext.weka_3.7.feature.group](#)
- [com.knime.features.bigdata.knosp.feature.group](#)
- [org.knime.features.xgboost.feature.group](#)
- [jp.co.infocom.cheminfo.marvin.feature.feature.group](#)
- [org.erlwood.features.core.base.feature.group](#)
- [com.continental.knime.feature.feature.group](#)
- [com.sjwebb.knime.slack.feature.feature.group](#)

Optionale externe Abhängigkeiten

Optional KNIME Server Große Instanzen (nur über BYOL-Lizenz verfügbar) können

Wählen Sie eine Verbindung KNIME Server zu einem externen LDAP/AD-Service. Vollständige Details enthalten

in [KNIME Server Advanced Setup Guide](#) .

Veröffentlichungshinweise

Relevante Änderungen der KNIME Server Software finden Sie im [Anmerkungen](#) und [entsprechend](#) [Änderungen](#) .

Wichtige Änderungen der KNIME Analytics Platform finden Sie im [Änderungen](#) .

Im Folgenden finden Sie alle AWS-spezifischen Änderungen der AMIs, die an AWS Marketplace eingereicht wurden.

4.9.0

- Aktualisierte OS-Pakete
- Aktualisierte Anaconda Python Paketversionen
- Boto3 zur Umgebung von Anaconda hinzugefügt
- Hinzugefügt symbolische Verbindung (/opt/knime/knime-latest) um auf den KNIME Executor zu zeigen
[leichter.](#)
- Geänderter Strickserver, den KNIME Executor über den symbolischen Link referenzieren
[leichter.](#)
- Neue Erweiterungen (AWS ML Services, ONNX, MLI, Plotly, Erlwood Base, Continental, Slack, Marvin)

4.8.2

- Aktualisiert von Ubuntu 16.04 LTS auf 18.04 LTS.
- Aktualisierte OS-Pakete.
- Aktualisierte Anaconda Python Paketversionen (Standard Python ist jetzt Version 3.6).
- Inklusive Chrony, um sicherzustellen, dass die Serveruhr korrekt auf AWS intern synchronisiert wird Ressourcen.
- Änderte Lage von Server und Executor (/etc/fstab) in /opt.
- Name-Tag zu Server-Repo-Block-Gerät hinzugefügt, um einfachere Updates zu unterstützen.
- Minor Bugfixes.

4.7.2

- Neues Bild: KNIME Server Small (inkl. 30 Tage kostenlose Testversion)

- Aktualisierte OS-Pakete.
- Hinzugefügt Anaconda Python.
- Minor Bugfixes.

4.6.4

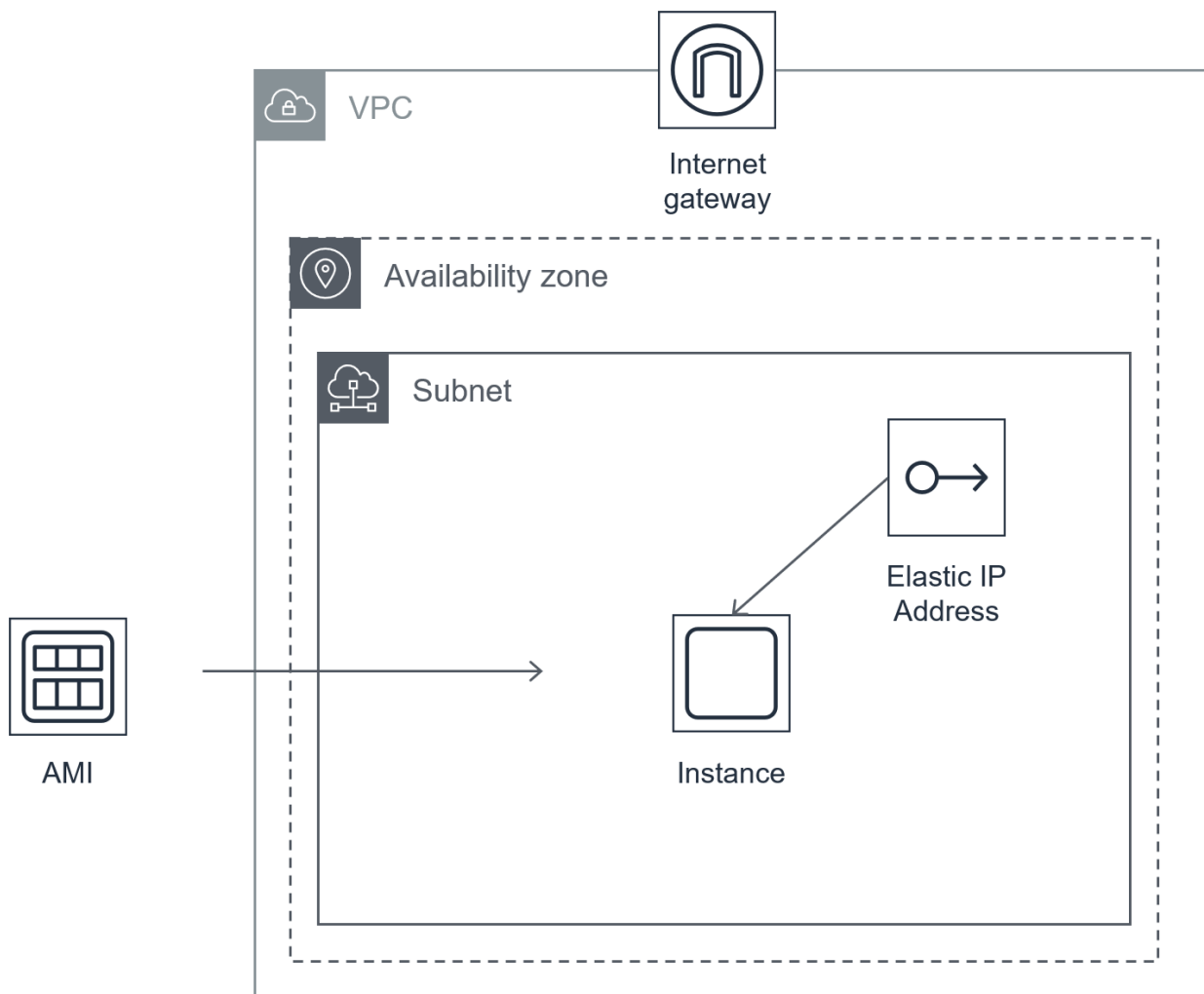
- Aktualisierte OS-Pakete
- Aktualisierte Systempakete.
- Aktualisierte Python Paketversionen.
- Minor Bugfixes.

Architektur im Überblick

Eine Übersicht über die allgemeine KNIME Server-Architektur wird geliefert. Weitere detaillierte Beschreibung von Software-Architektur finden Sie in der [KNIME Leitfaden für die Verwaltung von Servern](#).

KNIME Server Small/Medium (AWS)

KNIME Server Small und KNIME Server Medium laufen als einzige EC2 Instanz in einer einzigen Unternetz eines VPC. Die Verwendung eines elastischen IP ist bevorzugt, da es das Update/Upgrade vereinfacht Verfahren.



KNIME Server Large (AWS)

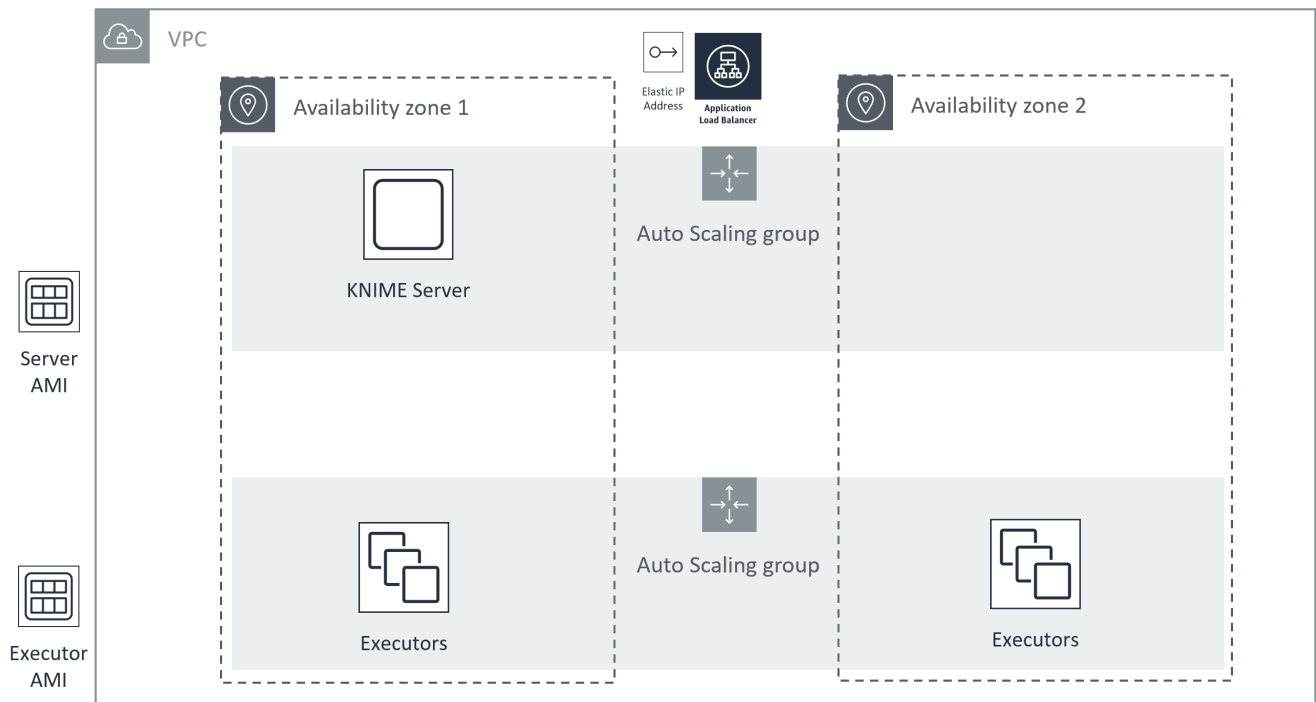
KNIME Server Large unterstützt Konfigurationen, die elastische Architekturen ermöglichen, die Unterstützung Skalierbarkeit und hohe Verfügbarkeit. Detaillierte Diskussionen zu diesen Themen finden Sie in

[Empfohlene AWS-Installation \(KN](#page24) Abschnitt.

Für Referenz haben wir hier eine der komplexeren Architekturen gezeigt, um Skalierung zu unterstützen

KNIME Ausführende und Versagen der Serverinstanz in eine neue Verfügbarkeitszone.

KNIME Server Large kann als einzige EC2-Instanz ausgeführt werden, wobei alle Anleitungen von oben kann verwendet werden.



Sicherheit

Detaillierte Beschreibung der allgemeinen Erwägungen für KNIME Server-Sicherheitskonfiguration
in der [KNIME Leitfaden für die Verwaltung von Servern](#)

Prüfpfad

KNIME Server-Log-Dateien sind über das KNIME Server AdminPortal akzeptabel oder durch Zugriff auf das
Dateien an ihren Standard-Standorten, wie in der [KNIME Leitfaden für die Verwaltung von Servern](#)

Spezielle Konfigurationen für den auf AWS laufenden KNIME Server werden nachfolgend beschrieben.

Zugang zu KNIME Server-Instanz

Root-Anmeldeinformationen sind nicht erforderlich, um auf die KNIME Server-Instanz zuzugreifen.

IAM Roles und Politik

IAM-Rollen/Politiken, die Zugang zu EC2-Instanzen ermöglichen und EBS-Volumen verwalten, sind
erforderlich, um den KNIME Server zu starten. Es wird angenommen, dass ein VPC mit einem Internet-Gateway ist
konfiguriert und verfügbar. Die Standardeinstellung erzeugt keine IAM-Rollen und Richtlinien
für KNIME Server und gewährt keinen öffentlichen Zugriff auf alle Ressourcen. Benutzer können die
IAM-Rolle für die EC2-Instanz von KNIME Server mit notwendigem Zugriff auf Basis ihrer Nutzung
Fälle, z.B. S3-Zugang.

Authentifizierung mit AWS

KNIME Server muss keine AWS-Dienste authentifizieren.

Schlüssel- und Rotationspolitik

Eine SSH-Taste für den Zugriff auf die KNIME Server-Instanz wird über die AWS erstellt oder ausgewählt
Management Console, oder CLI bei Instanz Start. Sie sind dafür verantwortlich, diesen Schlüssel als
je die Empfehlungen in Ihrer Organisation.

Gespeicherte Geheimnisse

Benutzer müssen bei der Bereitstellung der Lösung keine gespeicherten Geheimnisse erstellen oder verwalten.

EC2-Sicherheitsgruppen und VPC-Zugangskontrolllisten

Die Standard-Sicherheitsgruppe ermöglicht Zugriff auf den KNIME Server über HTTP und HTTPS auf Ports 80 und 443. Zusätzlich ist ein erweiterter Admin-Zugriff über den SSH-Port 22 aktiviert.

Es werden keine VPC-Zugangskontrolllisten definiert.

Konfiguration der Datenverschlüsselung

Es wird empfohlen, EBS-Verschlüsselung und EBS-Schnappverschlüsselung für alle zu aktivieren

KNIME Servervolumen. Vollständige Details in der [AWS Dokumentation](#).

Der einfachste Weg, um diese Verschlüsselung zu ermöglichen, ist die Auswahl der Festplattenverschlüsselung mit der Standardschlüssel, in der Phase, in der Sie die Volume-Optionen wählen können.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/sda1	snap	50	General Purpose SSD (gp2)	150 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted
EBS	/dev/sde		250	General Purpose SSD (gp2)	750 / 3000	N/A		

Add New Volume

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

Filter by attributes

KMS Key Aliases	KMS Key ID
Not Encrypted	
(default) aws/ebs	

Ressourcen erfassen

Sie können die EC2-Instanzen und Volumes für KNIME Server angeben, um z.

Eigentümer, Kostenzentrum, etc. Siehe [AWS Tagging Strategiedokument](#).

Standorte

Wir verwenden einen 'generischen' Ansatz, um Datei-Standorte zu beschreiben, und dann folgen Sie das mit dem Standard Einstellungen, die auf unseren Cloud-Marktplatzangeboten verwendet werden.

/srv/knime_server

Dieses Verzeichnis enthält sowohl permanente als auch ephemere Daten für den KNIME Server, für Beispiel Konfigurationsinformationen, Workflows, temporäre Daten. Sie möchten vielleicht Backup des gesamten Verzeichnisses, oder wenn Sie einen feinkörnigeren Ansatz wünschen, sollten Sie weiterlesen. Einige Teile dieses Verzeichnisses werden von Prozessen verwendet, die wahrscheinlich profitieren würden aus der Bereitstellung guter IO-Leistung. Für Details zu Festplattenwahlen, einschließlich

/opt/knime/knime-srv-4.11.x/apache-tomcat-9.0.x

Alle von der Apache Tomcat Installation benötigten Daten. Einschließlich Einstellungen für Authentifizierung, Sicherheit und Protokollierung. Die Sicherung dieses Verzeichnisses wird empfohlen.

/opt/knime/knime-4.2.x

Eine symbolische Verbindung zum neuesten KNIME Executor (im selben Verzeichnis). Enthält der ausführbare Ausführer und alle installierten Erweiterungen. Backup dieses Verzeichnisses ist empfohlen.

Für eine genauere Definition dessen, welche Art von Daten in jedem Ort gespeichert wird, lesen Sie bitte weiter.

Ständige Daten

/config

/srv/knime_server/config

Spezielle Konfigurationsinformationen für den KNIME Server. Zum Beispiel die Risse... server.config, knime.ini und die Anpassungsprofile. Backup dieses Verzeichnisses ist empfohlen.

/Erweiterungen

Enthält weitere Informationen, die für Erweiterungen zum KNIME Server erforderlich sind, wie die Vorschauversion des Workflow Hubs. Backup dieses Verzeichnisses ist empfohlen.

/jobs

Speichert Informationen über laufende Jobs. Praktisch sind die Arbeitsplätze eine Kopie eines Workflows aus dem Workflows-Verzeichnis, das dann zusätzliche Laufzeitdaten haben. Backup von diesem Verzeichnis wird empfohlen. Erhöhung der IO Bereitstellung für dieses Verzeichnis hilft Geschwindigkeit

die Schaffung neuer Arbeitsplätze für den Arbeitsablauf oder das Vertauschen von Arbeitsplätzen zu/aus dem Gedächtnis.

/licenses

Speichert die vom KNIME Server benötigte Lizenzdatei. Backup dieses Verzeichnisses ist empfohlen. Wenn Sie eine Lizenzdatei benötigen, kontaktieren Sie Ihre KNIME Kundenbetreuung

Vertreter oder Verkauf@knime.com

/trash

Der Standort von Workflows oder Datendateien, die auf den KNIME Server verschoben wurden 'Recycle Bin'. Sie möchten möglicherweise noch dieses Verzeichnis sichern, um sicherzustellen, dass versehentlich gelöschte Dateien sind möglich.

/workflows

Der Speicher aller Workflows, die auf den KNIME Server hochgeladen werden. Zusätzliche Metadaten wie die Berechtigungen auf den Workflows und ihre OpenAPI-Spezifikation werden gespeichert in diesem Verzeichnis. Die Sicherung dieses Verzeichnisses wird empfohlen. Erhöhung der IO-Vorbereitung für dieses Verzeichnis wird helfen, die Erstellung neuer Workflows zu beschleunigen und neue Jobs.

Ephemere Daten

/runtime

Speichert Informationen, die für lokal laufende Ausführer erforderlich sind. Dieses Verzeichnis wird nicht verwendet wenn die verteilte Ausführung verwendet wird. Die Sicherung dieses Verzeichnisses ist nicht erforderlich. Es wird nach Bedarf regeneriert. Erhöhung der IO Bereitstellung für dieses Verzeichnis wird helfen Arbeitszeit, insbesondere bei "IO-gebundenen" Workflows.

/temp

Dieses Verzeichnis wird als Zwischenspeicher für den Tomcat-Prozess des KNIME verwendet Server. Zum Beispiel beim Herunterladen großer Dateien vom Server. Backup dieses Verzeichnisses ist nicht erforderlich.

Größe

Es gibt keine "eine Größe passt alle" Antwort auf Fragen rund um die Dimensionierung von Bereitstellungen. Die Antwort hängt von Ihrer typischen Arbeitsbelastung, Anzahl der gleichzeitigen Benutzer, gewünschte Berechnung Zeit und so weiter. Wir bieten einige Empfehlungen, um zu helfen, zu starten.

KNIME Server Small/Medium

Berechnungen

Der rechnerischste intensive Teil des KNIME Servers führt Workflows durch. Als Beispiel, wenn Sie erwarten, dass 10 gleichzeitige Verbraucher den gleichen Analyse-Workflow auf Das KNIME Server etwa gleichzeitig. Der Workflow benötigt ca. 2GB von RAM, und führt in einer angemessenen Zeit mit 2 Kernen. Um die Arbeitsbelastung zu laufen ein einzelner Executor würde 20 GB RAM und 20 Kerne benötigen.

Darüber hinaus sollten Sie bis zu 4 Kerne, 4GB RAM für den Tomcat Server Prozess reservieren, die ist in erster Linie für den Austausch von Workflows und Daten verantwortlich.

Lagerbetrachtungen

Der Tomcat Server benötigt mindestens 30 GB für das Betriebssystem und die Anwendung selbst. Da der Tomcat Server auch das KNIME Server Workflow Repository hostet, a Mindestens 250 GB zusätzlicher Speicher wird auch zur Speicherung von Workflows empfohlen, und zusätzliche Laufzeitinformationen.

Die Speicherung einer größeren Anzahl von Workflows, Daten oder Jobs erfordert natürlich mehr Speicherplatz.

Für weitere Informationen, auf welchen Festplatten-Standorten welche Art von Informationen gespeichert werden, siehe die [Standorte](#page16) Abschnitt. Der Abschnitt dokumentiert auch, welche Speicherplätze sich verbessern können Anwendung Leistung durch erhöhte IO Bereitstellung.

KNIME Server groß

Da ein typischer Einsatz von KNIME Server Large die "Verteilt Bei der Auslegung einer Bereitstellung.

Berechnungen

Tomcat Server

Der Tomcat Server ist verantwortlich für die Verwaltung von Interaktionen mit dem KNIME Server Repository. Daher, wenn eine große Anzahl von Arbeitsabläufen parallel, oder wenn Sie eine große Anzahl von Arbeitsplätzen im Workflow-Repository, die die Größe dieses Servers sein muss erhöht. Beim Einsatz verteilter KNIME Executors verbraucht der Tomcat Server vier Kerne aus der KNIME Server-Lizenz. In der Mehrheit der Setups wird es ausreichen, 4 zu reservieren Kerne zum Tomcat Server. Eine Standardinstallation ordnet dem Tomcat-Prozess 2GB RAM zu, Obwohl es sinnvoll sein kann, den für Tomcat verfügbaren RAM auf 4-8 GB zu erhöhen.

RabbitMQ

RabbitMQ wird als Nachrichtenmakler für die Server-Executor-Kommunikation verwendet. Die Menge Verkehr durch die Warteschlange ist ziemlich begrenzt. Aus diesem Grund ist es möglich, nur 1 CPU zu reservieren Kern und 500Mb RAM zu diesem Dienst. In einigen Bereitstellungen kann es wünschenswert sein, dass Software auf die gleiche Maschine wie der Tomcat Server.

Ausführung

Um die Ausführung einer größeren Anzahl von Workflows zu unterstützen, ist es möglich, mehr als einen Executor. Die minimale Größe eines einzelnen Ausführenden sollte durch die Prüfung der CPU entschieden werden und RAM-Anforderungen zur Ausführung eines typischen Workflows und des gewünschten Workflows Parallelismus.

Betrachten Sie das folgende Beispiel. Sie erwarten, dass 20 gleichzeitige Verbraucher dasselbe ausführen Analyse-Workflow auf dem KNIME Server etwa gleichzeitig. Der Workflow benötigt ca. 2 GB RAM und führt in einer angemessenen Zeit mit 2 Kerne. Um die Arbeitsbelastung auf einem einzelnen Ausführenden zu betreiben, benötigen 40 GB RAM und 40 Kerne. Es gibt einen kleinen RAM Overhead für einen Executor-Prozess, um ~1-2GB auszuführen.

Wenn sich die Anzahl der Nutzer nun verdoppelt, könnte die Größe der Ausführende Maschine (doppeln Sie die Größe), oder einen zweiten Ausführenden, der gleichen Größe wie die Erster Executor.

Ein deutlicher Vorteil der Verwendung einer größeren Anzahl von Ausführenden besteht darin, dass dies Flexibilität für bei wechselnder Ausführeranforderung die Ausführenden hinzufügen/entfernen. Das muss gewogen werden gegen den begrenzten zusätzlichen RAM-Anforderung für den Betrieb eines Ausführenden.

Lagerbetrachtungen

Tomcat Server

KNIME Server Large hat die gleichen Speicherüberlegungen wie KNIME Server Small und Medium. Siehe Abschnitt Lagerbetrachtungen für alle Details.

RabbitMQ

Kaninchen MQ benötigt mindestens 200 MB Freiraum und kann typischerweise mit einem Wurzelvolumen von 50 GB.

Ausführung

KNIME Ausführende benötigen mindestens 30 GB für das Betriebssystem und die Anwendung selbst. Es ist auch eine Menge temporärer Speicherplatz erforderlich. Seit der Ausführung von einigen KNIME Workflows können IO gebunden sein (vor allem, wenn begrenztes RAM verfügbar ist), ist es empfohlen, dass die Ausführenden Zugriff auf SSD-Klasse-Speicher haben.

Dimensionierung (AWS)

KNIME Server Small und KNIME Server Medium werden beide über den AWS Marketplace mit gebaut in Lizenzen für 5 benannte Benutzer, und maximal 4 Kerne für Workflow-Ausführung.

Zusätzlich KNIME Server Medium ermöglicht 20 Verbrauchern Zugriff auf den KNIME Server

WebPortal nur über den Webbrowser. Bitte kontaktieren Sie uns

Verkauf@knime.com

wenn Sie eine größere

Anzahl der Nutzer, Verbraucher oder Kerne.

Für eine allgemeinere Diskussion über die Leimung von KNIME Server siehe Abschnitt

[Hier](#).

EG2 Entscheidung

Typischerweise kann Workflow-Ausführung Geschwindigkeit von zusätzlichen verfügbaren Instanz RAM profitieren.

Daher empfehlen wir die "R"-Instanztypen, da sie den besten Zugang zu

RAM.

Die R5a.2xlarge Instanz hat 64 Gb RAM zur Verfügung, und auch 8 CPU-Kerne sind damit die größten

Beispiel, von der KNIME Server Small und KNIME Server Medium Gebrauch machen können.

Vollständige Details zu EC2 Instanz-Typen finden Sie

[Hier](#).

Anmerkung zur Kernzählung

KNIME Die Ausführenden identifizieren einen Kern mit dem JVM. Bei AWS wird standardmäßig ein KNIME Executor Kern entspricht einer vCPU. In einigen Fällen können Sie finden, dass Sie besser pro Kern Workflow Ausführungsleistung, indem Sie die Anzahl der Fäden pro Kern optimieren. Z.B. mit einer R5a.4xlarge Instanz, mit Gewinden pro Kern auf 1. Diese Einstellungen erstellen Änderungen werden offensichtlich auf Kosten zusätzlicher Infrastrukturkosten (r5a.2xlarge vs. [r5a.4xlarge](#)). Siehe [AWS Dokumentation](#) für alle Details.

EBS Volumenauswahl

Das Standard-Root-Instanzvolumen ist 50Gb SSD (gp2), und in den meisten Fällen ist es nicht erforderlich, die Volumengröße erhöhen. Das zusätzliche Volumen hat eine Standardgröße von 250Gb SSD (gp2), die sollte für viele neue Anlagen angemessen sein.

Um die Volumengröße zu führen, müssen Sie beachten:

- Größe und Anzahl der zu speichernden Workflows
- die Anzahl der ausgeführten Arbeitsplätze und die Dauer der Arbeitserhaltung
- Anzahl und Größe der im Workflow-Repository gespeicherten zusätzlichen Dateien

Falls Sie später zusätzlichen Speicherplatz hinzufügen müssen, sehen Sie bitte den Abschnitt

[Workflow Repository EBS Volume](#page47)

KNIME Server groß

Sie können alle Empfehlungen für KNIME Server Small/Medium folgen. jedoch bei Verwendung ein KNIME Server Großer Einsatz mit verteiltem KNIME Ausführungsbeispiele, die Sie verwenden möchten mehrere homogene Instanztypen für die KNIME Executors, um die Flexibilität der Skalierung von Instanztypen abhängig von den erforderlichen Executor-Workloads. Sie können auch betrachten Skalierung in der KNIME Server-Instanz zu einem etwas kleineren Instanztyp (kein weniger als 4 vcpus), da die Ausführung nun an den KNIME Executors selbst geschieht. Seit RAM ist weniger relevant für die KNIME Server-Instanz, Sie können eine "m"-Instanz wählen anstatt eine 'r'-Instanz.

Kosten

Die Kosten für den Betrieb eines KNIME Servers variieren je nach Faktoren. Diese die gewünschte Workflow-Ausführungsleistung, die Menge der Daten, die geplant ist, werden gespeichert, die Backup-Strategie und geplante Failover-Setup.

Ihr KNIME Kundenbetreuer würde Ihnen gerne Beratung und Beratung anbieten wie diese Entscheidungen Ihr Setup beeinflussen. Im Folgenden geben wir einige Informationen über typische Setups, um eine Idee der Preisgestaltung zu geben.

Softwarepreise

Die Softwarepreise für den KNIME Server sind im AWS Marketplace definiert. Vgl.

[AWS](#)

[Marktplatz Preise](#) Fragen zur BYOL-Lizenz sollten auf

Verkauf@knime.com .

Hardwarepreise

Die Hardwarepreise werden durch AWS definiert. Vgl.

[AWS Pricing](#) .

[Alle mit Hilfe der](#)

[AWS Kostenrechner](#) .

KNIME Server klein

Erforderliche Dienste (Server Small/Medium - inkl. BYOL)

- EG2
- EBS Volumen
- Datenübermittlung in/out
- Optional: EBS Snapshots

Bereitstellung

Bereitstellung von KNIME Server Small und KNIME Server Medium ist über eine einzige virtuelle Maschine. KNIME Server Large bietet mehrere Optionen für hochverfügbare (HA) Bereitstellungen.

KNIME Installation von Servern

Umgebungen KNIME Die Serverinstallation wird im Detail mit der [KNIME Server Installationsanleitung](#) . Darüber hinaus bieten wir vorkonfigurierte Bilder über Marketplaces an, die in dieser Anleitung abgedeckt.

Testen der Bereitstellung

Einfache Prüfung der Bereitstellung kann durch die Anmeldung in KNIME Server WebPortal über den Webbrowser. Bestimmte Funktionalität ist nur über die KNIME Analytics zu testen Plattform.

Verbindung über den Browser

Sobald Sie den KNIME Server AMI gestartet haben, startet die resultierende Instanz automatisch den KNIME Server. Das KNIME Server WebPortal ist im Browser unter <https://knime>

Verbindung über die Analytics Plattform

Der Zugang zum KNIME Server von der KNIME Analytics Platform erfolgt über den KNIME Explorer.

Vollständige Dokumentation ist in der [KNIME Benutzerhandbuch des Servers](#) Verwenden Sie die Mount Point Adresse:
`https://knime`

Prüfung der Workflow-Ausführung

Klicken Sie auf einen beliebigen Workflow aus dem WebPortal-Repository-Baum und warten Sie auf die Seite zu laden. Wenn

'Start' Taste erscheint dann Workflow Ausführung funktioniert wie erwartet. Für automatisierte Tests

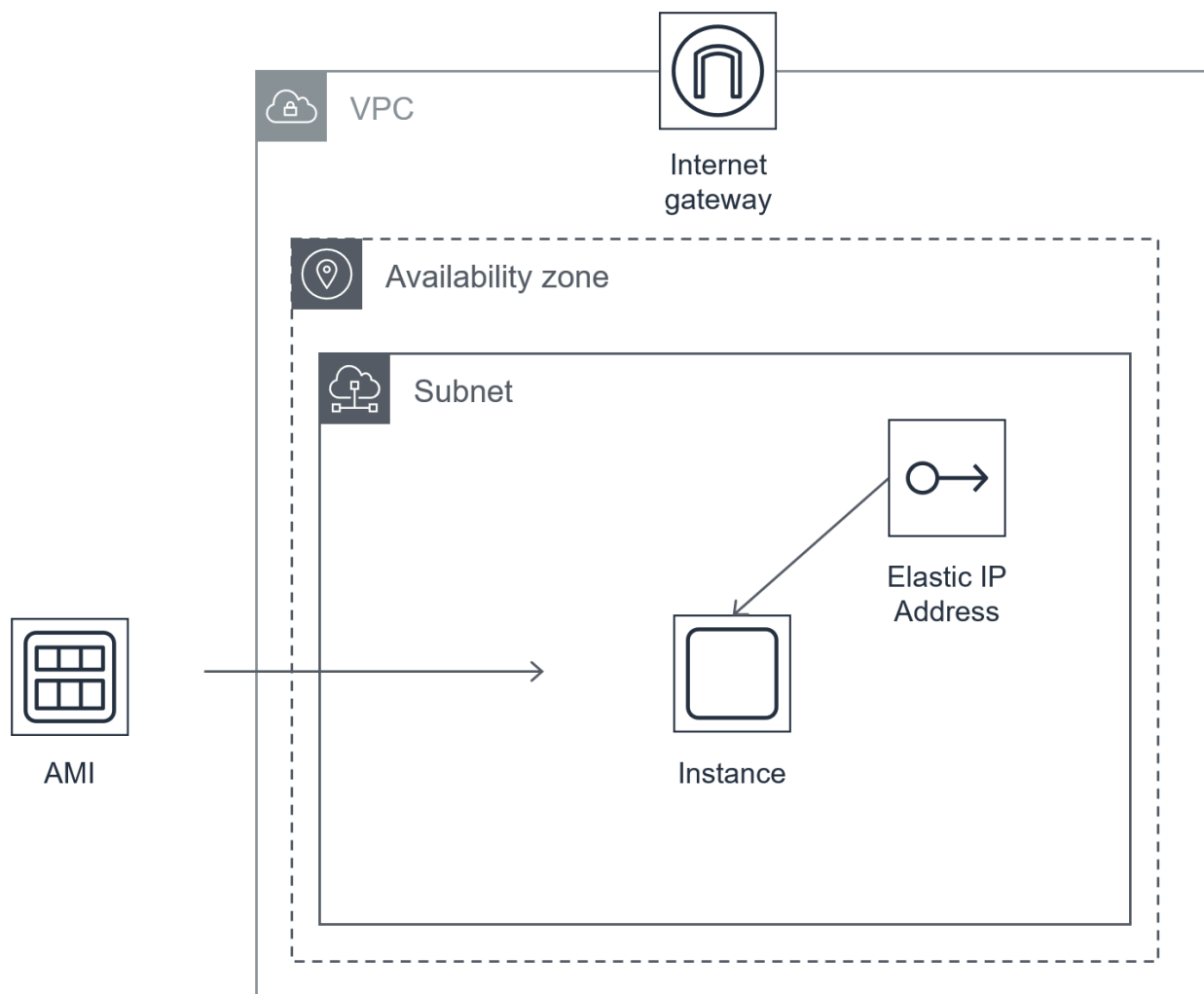
[Start](#) [Taste](#) erscheint dann Workflow-Ausführung funktioniert wie erwartet. Für automatisierte Tests

Empfohlene AWS-Bereitstellung (KNIME Server Klein/Medium/Large (via BYOL))

Einfache Bereitstellung

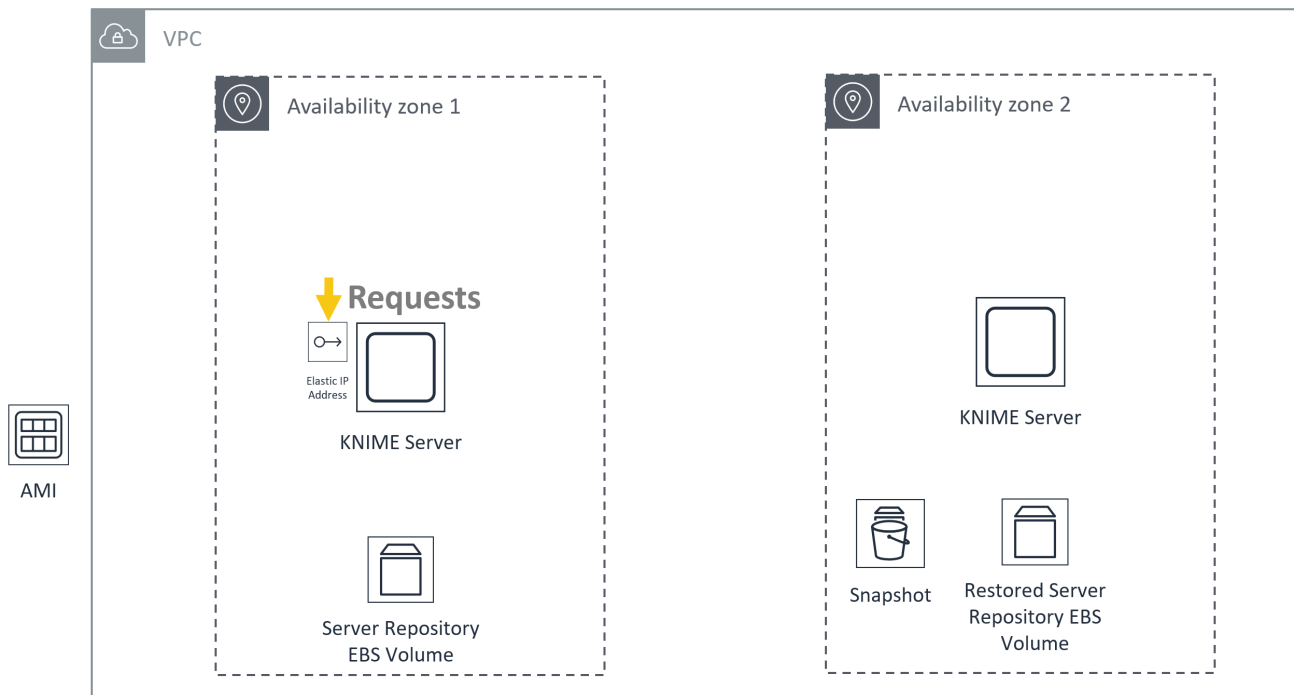
Eine einfache Bereitstellung gemäß den Größenleitlinien der vorherigen
wie:

[Abschnitt sieht aus](#page20)



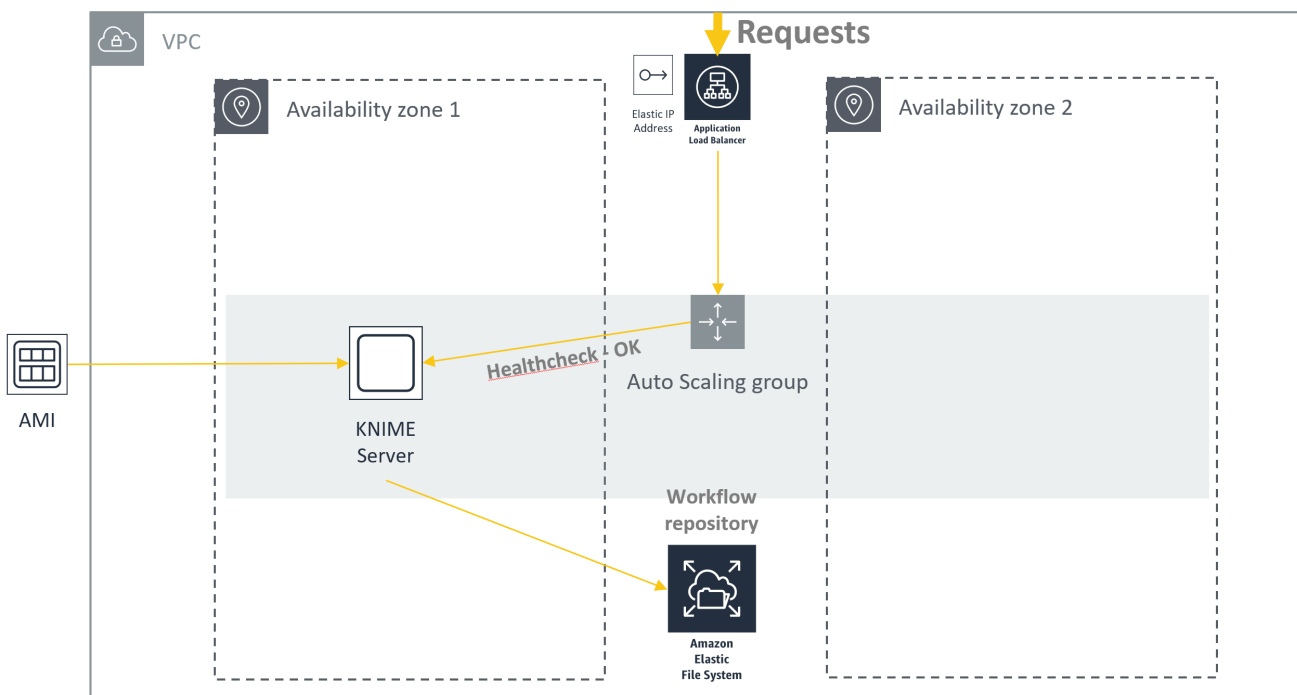
Einfaches Failover

Eine einfache, nicht automatisierte Bereitstellung, um eine Katastrophenrückgewinnung bei z.B. einem Fall zu gewährleisten
korruptiert zu werden, ist, einen Snapshot der Instanz (in regelmäßiger Basis) zu nehmen. Dann
Fall, dass Katastrophenrettung erforderlich ist, schalten Sie einfach die Elastische IP von der alten Instanz auf
die neue Instanz.



Kalt-Standby (EFS)

Durch Verschieben der Inhalte des /srv-Verzeichnisses von einem EBS-Volume auf [AWS EFS](#), es wird ein Failover-System, das den neuesten Zustand des KNIME Servers bereiten kann Workflow Repository.



Failover dauert ein paar Minuten, während die neue Instanz Stiefel und lädt das Repository Inhalt.

Mit diesem Setup gibt Widerstand gegen beide Instanz-Versagen, und Availability-Zone-Versagen.

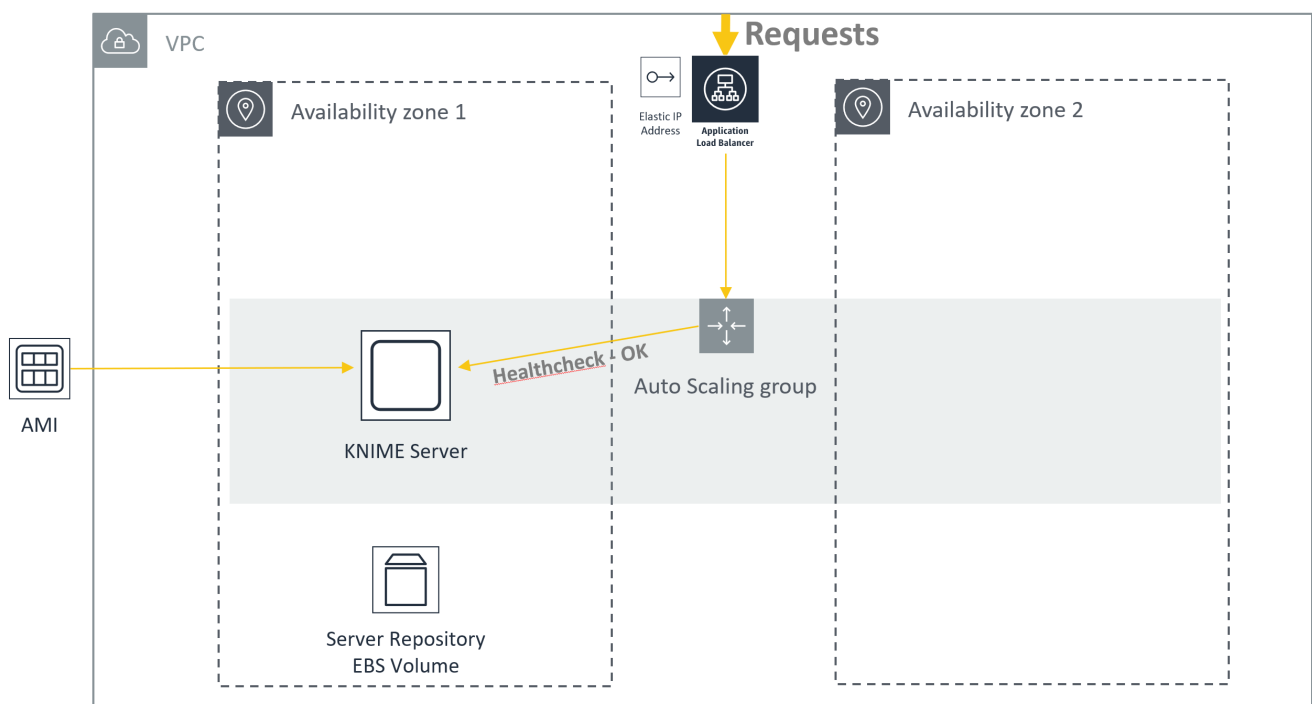
Schlussfolgerung

Während es technisch möglich ist, zwei KNIME Server-Instanzen mit dem gleichen Workflow-Repository, so wird zu Datenverlust führen.

Kalt-Standby (EBS)

Der Einsatz auf AWS macht es sehr einfach, bei einem Beispiel Kälte-Standby zu bieten.

Scheitern. In diesem Fall können Sie entweder die EBS-Volume des Workflows migrieren Repository, oder Sie müssen eine regelmäßige Snapshot der Datenblockgeräte.



Stellen Sie sicher, dass die Snapshots in der Region verfügbar sind, die Sie die Kälte starten wollen- Standby-Server.

Bei der Implementierung dieses Setups mit den AWS Marketplace-Bildern müssen Sie ändern

das AMI, so dass das Admin-Passwort nicht auf Instanz Failover zurückgesetzt wird. Wenn Sie das BYOL-Bild verwenden und eine KNIME Server Große Lizenz, die empfohlen wird, das Problem zu lösen ist, durchzuführen

Authentifizierung mit LDAP/Active Directory. Für KNIME Server Small/Medium, wo LDAP ist

nicht verfügbar, Sie müssen das Passwort-Set-Skript durch Umbenennen deaktivieren

/opt/init_db.sh

Sie müssen Benutzerdaten-Skripte implementieren, die das Anbringen/Defizieren der EBS verwalten

Volumen, um auf die Failover-Instanz migriert werden, und wenn Sie auf eine neue Verfügbarkeit migrieren müssen

Zone müssen Sie auch die Instanz schnappen. Aufgrund dieser Komplexität und der zusätzlichen

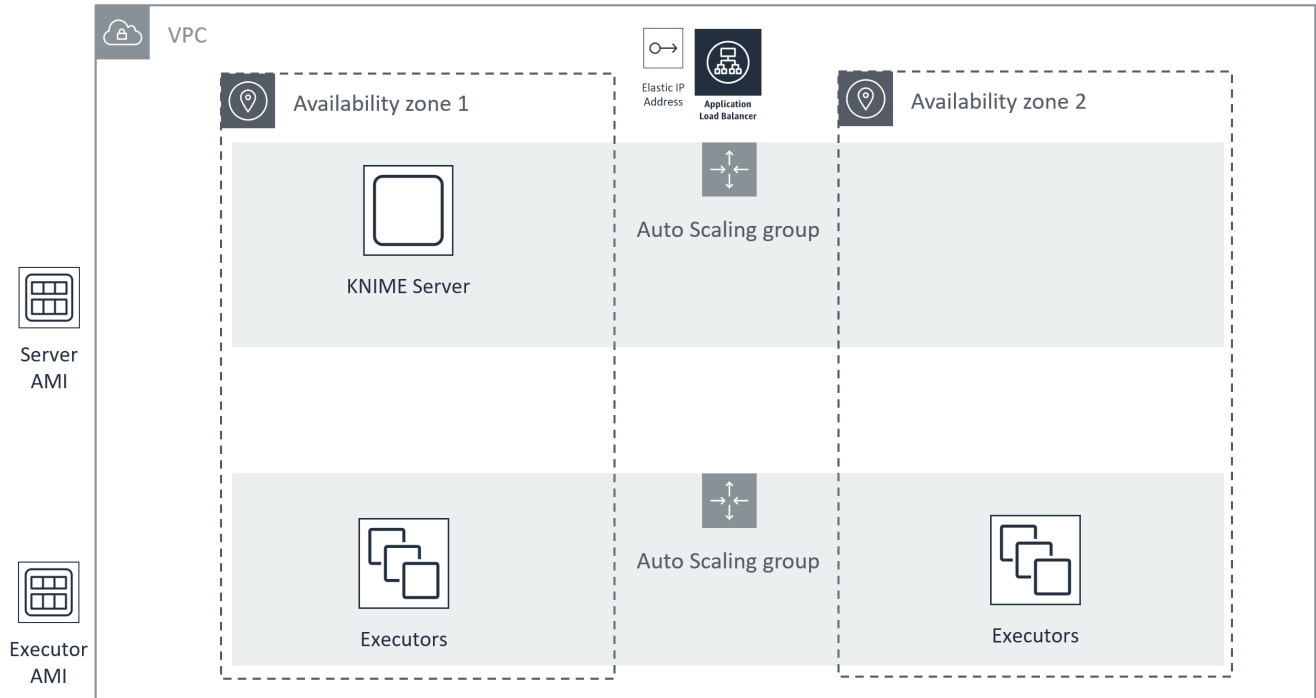
Komplexität, um die Verfügbarkeit Zone Ausfall Widerstandsfähigkeit zu gewährleisten, Kältestandby mit AWS EFS ist

in den meisten Fällen bevorzugt.

Empfohlene AWS-Bereitstellung (KNIME Server Large)

Ein typischer Einsatz nach den Größenvorgaben in den vorherigen
wie:

[<a href="#page18" style="color: #ff6600;](#page18)
Abschnitt sieht aus



Standard KNIME Server-Passwort

Wir setzen kein festes Standard-Passwort für den KNIME Server, da dies ein bekannter Fehler ist Sicherheitspraxis. Daher wird das Standard-Passwort auf die Instanz-ID des AMI auf der Erster Start des KNIME Servers. Dies kann auf der AWS-Konsole oder durch Ausgabe der Befehl aus der KNIME Server AMI-Instanz:

```
`curl http://169.254.169.254/latest/meta-data/instance-id`
```

Beim ersten Login werden Sie empfohlen, einen neuen Admin-Benutzer zu erstellen, und dann die Kimeadmin Benutzer.

Lizenzdatei auf KNIME Server auf AWS (BYOL) anwenden

Für KNIME Server (BYOL) müssen Sie Ihre Lizenzdatei anwenden. Dies kann durch einen Besuch geschehen <https://knime> von Ihrem Webbrowser.

Einloggen mit dem Benutzernamen admin: knimeadmin, und Passwort: , wird umleiten Sie auf die Lizenz-Upload-Seite. Hier können Sie Ihre Lizenzdatei anwenden. Eine gültige Lizenzdatei wird

sofort angewendet, und Sie können beginnen, alle KNIME Server-Funktionalität zu verwenden.

In der AWS-Konsole (EC2-Seite) finden Sie die für den gewünschten Fall. Oder

Sie können sich an die Instanz über SSH anmelden und den Befehl ausgeben:

```
Curl http://169.254.169.254/latest/meta-data/instance-id
```

Vorlage der Bereitstellung

Insbesondere bei KNIME Server Large, wo es potenziell mehrere Instanzen gibt die eingesetzt werden müssen, mit zusätzlichen Einstellungen wie Netzwerken, Sicherheitsgruppen und Zugriffsrichtlinien, empfehlen wir dringend, die Verwendung von Templated-Bereitstellungen zu berücksichtigen. Vorlagen-Einstellungen können sicherstellen, dass keine wichtigen Einstellungen verpasst werden, und dass die Ein Einsatz kann bei Bedarf identisch repliziert werden.

Templatiermaschine

In dieser Dokumentation beschreiben wir eine Methodik, die das "native" Temporierungstool für Ihre Cloud-Plattform der Wahl, z.B. CloudFormation für AWS oder Azure Resource Manager (ARM) für Azure. Sie können auch ein Drittanbieter-Tool wie Terraform betrachten. Wir zeigen nicht spezifische Beispiele [Hashicorp Terraform](#), aber Übersetzung aus den nativen Beispielen sollte einfach sein.

Vorlage VM Aufbau

In dem Fall, dass Sie wählen, um Ihr eigenes Bild zu bauen, möchten Sie fast sicher automatisieren dass aus den gleichen Gründen wie die Automatisierung des Infrastruktureinsatzes. Es ist mit einem Werkzeug wie [Hashicorp Packer](#).

Neben Marktplatzbildern für KNIME Server (klein, mittel, groß oder BYOL) Sie haben die Möglichkeit, KNIME Server von Grund auf zu installieren und zu konfigurieren. In diesem Fall Ein Installationsprozess wird im [KNIME Installationsanleitung für Server](#). Zusätzlich Konfigurationen von KNIME Server werden im [KNIME Leitfaden für die Verwaltung von Servern](#).

In dem Fall, in dem Sie den Aufbau eines KNIME Server 'Golden Image' automatisieren möchten, können Sie die oben genannten zwei Dokumente als die erforderlichen Informationen zu betrachten. Es wird dann notwendig, um Ihren internen Build-Prozess anzupassen, um diesem Verfahren folgen. Keine der Werkzeuge Sie sind erforderlich, und Sie können wählen, Alternativen zu verwenden.

Wir beschreiben in Kürze die Schritte, die wir bei KNIME verfolgen, um den Azure Marketplace zu bauen Bilder. Wir verwenden das Werkzeug [Hashicorp Packer](#) den Prozess zu automatisieren.

Die Beschreibung soll keine vollständige Liste sein, sondern eine Übersicht über die Arten von Dingen dass Sie berücksichtigen müssen.

Packerschritte für KNIME Server Small/Medium

Wir folgen Schritten wie:

- Definieren Sie 'Basis' Bild. Wir wählen Ubuntu 22.04 LTS.
- Neueste OS-Patches anwenden
- Konfigurationsdateien hochladen (preferences.epf, knime.ini, License.xml, autoinstall.xml, etc.)
- VM Benutzer erstellen (knime)
- Installieren Sie die erforderliche Abhängigkeit (Java JDK 11)
- Automatisches KNIME laufen Server-Installation
- Installieren Sie optionale Abhängigkeiten (Python, R, Chrony, etc.)
- Port Forwarding/Firewall oder Frontend Webserver konfigurieren
- Reinigungsbild und Verallgemeinerung

Packerschritte für KNIME Server groß

Folgen Sie den Schritten für KNIME Server Small/Medium, um das "Server"-Bild zu erstellen. Sie möchten vielleicht

Deaktivieren Sie die Teile des Builds, die den Executor installieren. Dann folgen Sie Schritten wie:

- Definieren Sie 'Basis' Bild. Wir wählen Ubuntu 22.04 LTS.
- Neueste OS-Patches anwenden
- Konfigurationsdateien hochladen (knime.ini, Executor Launch script)
- VM Benutzer erstellen (knime)
- Ausführen von KNIME Server
- Installieren Sie optionale Abhängigkeiten (Python, R, Chrony, etc.)
- Reinigungsbild und Verallgemeinerung

AWS CloudFormular Template-Bereitstellung

Während es einfach und bequem ist, den KNIME Server über die AWS-Konsole einzusetzen, gibt es

sind starke Argumente für die Verwendung einer Template-Bereitstellung mit

[CloudFormation Vorlagen](#)

AWS CloudFormation Templates erlauben es, den vollen Zustand der endgültigen Bereitstellung so zu beschreiben

dass sie in Zukunft gleich eingesetzt werden kann. Ebenso ist es möglich, einen Master zu ändern

Template, die eine einfache Wiederverwendung von freigegebenen Konfigurationen und Anpassungen ermöglicht, in denen erforderlich.

Für einen Überblick über die möglichen Arten von Bereitstellungen mit einer CloudFormation Vorlage siehe [die AWS-Dokumentation](#) [Quickstart-Vorlagen](#) .

Erstellung einer CloudFormationsvorlage

Die CloudFormation Vorlage beschreibt die AWS-Infrastruktur, die benötigt wird für den KNIME Server gestartet.

KNIME Ausführende mit Auto Scaling Groups

Für Auto Scaling KNIME Executors auf AWS werden drei Komponenten benötigt:

- Ein VPC, der KNIME Server, RabbitMQ und die KNIME Executors beherbergt
- Eine AWS Auto Scaling Group, die das Starten und Abschalten von KNIME verwaltet

Ausführung



KNIME Server sollte in einem öffentlichen Subnetz eingesetzt werden, während KNIME Executors sollte in einem privaten Subnetz eingesetzt werden.

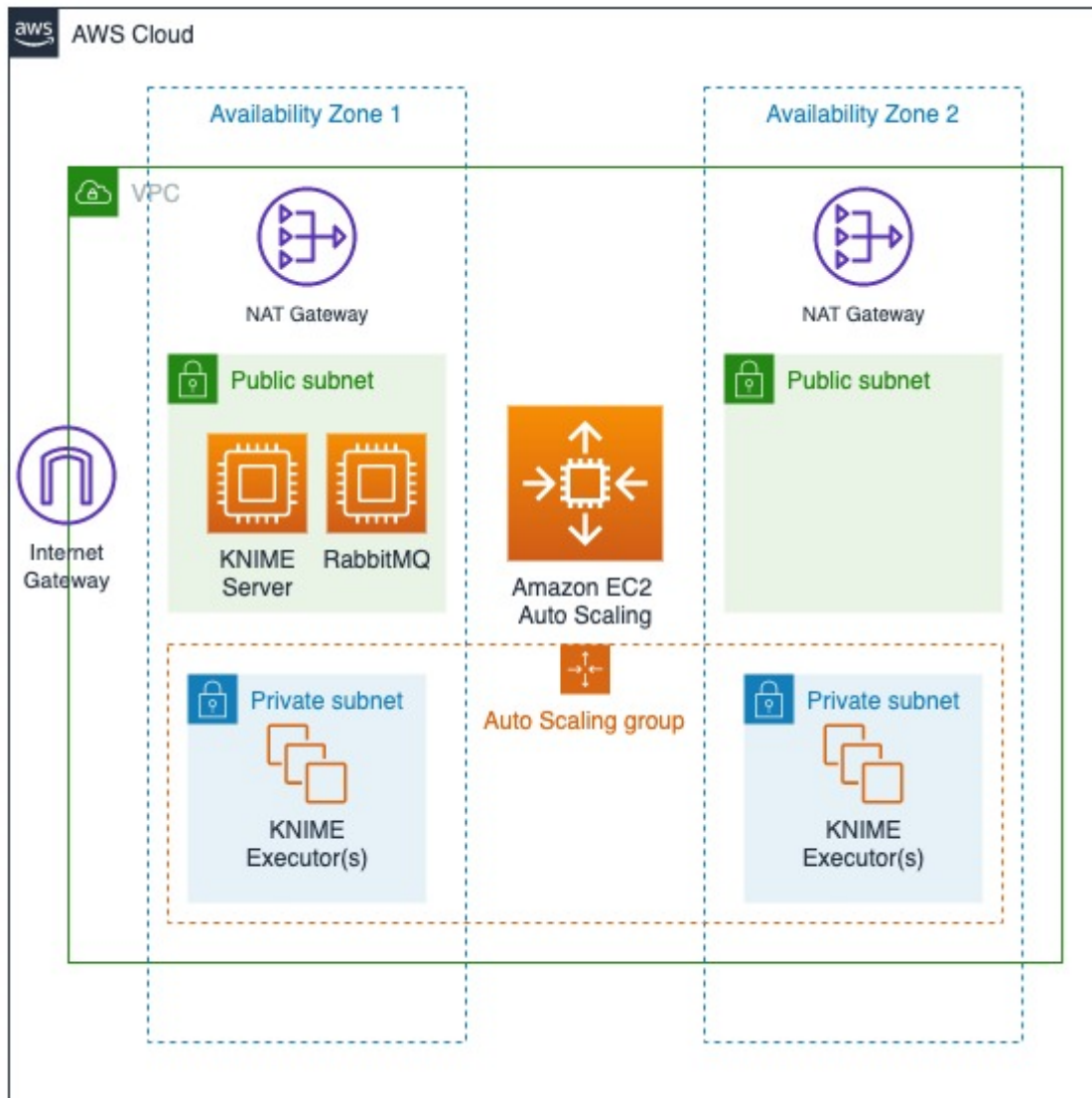
Einrichtung des VPC

Eine Virtual Private Cloud (VPC) innerhalb der AWS Cloud-Plattform bietet eine sichere Vernetzung Umgebung für KNIME Software. Jeder VPC bietet eine eigene Netzwerk-Domain, die bietet Zugang zu und Schutz von der externen Netzwerk-Welt. Die Divisionen sind gut definiert und Schichten der Netzsicherheit vorgesehen sind.

Bei der Bereitstellung des KNIME Servers innerhalb eines VPCs wird dieser üblicherweise in einem öffentlichen Subnetz eingesetzt. Dadurch können Verbindungen zum Server aus dem externen Internet hergestellt werden. Sicherheit Gruppen können verwendet werden, um den Zugriff auf den KNIME Server zu steuern (d.h. offene Ports für spezifische IP Adressen oder eine Reichweite von IP-Adressen). Es gibt Szenarien, in denen der KNIME Server sein kann in einem privaten Subnetz eingesetzt. Die Anwendungsfälle für diese Bereitstellung umfassen: alle Zugriffe auf die Server wird innerhalb des VPC erstellt oder der VPC ist über ein VPN an ein Unternehmen angeschlossen privates Netz.

KNIME Die Ausführenden werden in der Regel innerhalb privater Subnetze eingesetzt. Die Begründung hier ist, dass die Außenwelt muss sich nicht mit den Ausführenden verbinden. Ausführende kommunizieren nur mit KNIME Server und RabbitMQ. Wenn Sie innerhalb einer Auto-Scaling-Gruppe (ASG) laufen, Die Executors sollten innerhalb von Subnetzen ausgeführt werden, die in verschiedenen Verfügbarkeitszonen enthalten sind. Das gibt den Ausführenden ein Niveau der Widerstandsfähigkeit. Wird die Vernetzung innerhalb eines AZ nicht verfügbar, wird die ASG automatisch neue Executors im noch funktionierenden AZ initiieren. Während Die Gesamtberechnungskapazität kann in der Periode gesenkt werden, während die neuen Ausführenden beginnen bis, der Dienst der Ausführenden ist nicht gestört. Und innerhalb kurzer Zeit die gewünschte Das Niveau der Rechenleistung wird wiedererlangt.

Das folgende Diagramm zeigt den typischen Einsatz von KNIME Server mit KNIME Executors innerhalb eines VPC in AWS.



CloudFormation Vorlagen

KNIME bietet [CloudFormation](#) Vorlage zusammen mit den AWS Marketplace AMIs

Vereinfachung der Erstellung der AutoScalingGroup.

Ausführer.yaml

Dieser Stack erstellt die AutoScalingGroup, die die KNIME-Ausschnitte regelt, sowie die Sicherheitsgruppe für diese Fälle; die AMIs für KNIME Executor Instanzen sind auf den AWS-Marktplatz.

Öffnen Sie die yaml-Datei in einem Texteditor und bearbeiten Sie die RegionMap, die das KNIME angibt Ausführender AMI. Stellen Sie sicher, auch die richtige AWS-Region einzustellen.

Füllen Sie die Vorlage wie in der CloudFormation-Konsole veranlasst aus und erstellen Sie den Stack. Wählen

die VPC, in der die Serverinstanz ist. Im Abschnitt Executor Configuration müssen Sie festlegen die private IP, die der Serverinstanz zugewiesen wurde; geben Sie nicht die öffentliche IP seit KNIME Ausführende Instanzen dürfen nicht außerhalb des VPC kommunizieren. Setzen Sie die zielmittlere CPU-Auslastung für die Skalierungsgruppe.



Die Verwendung eines zu niedrigen Wertes könnte dazu führen, dass die ASG zweimal sogar skaliert wenn nur eine Skala nach oben gewünscht wird. Ein Wert von 90% sollte angemessen sein Ausgangspunkt. Es ist möglich, diese Einstellungen an einem späteren Punkt in der AutoSkalierung Gruppenkonsole.

Darüber hinaus müssen Sie die Mindest- und Höchstgrenzen angeben, wie viele Ausführende sollte gestartet werden. Falls Sie nicht dynamisch skalieren möchten, sondern BYOL verwenden Ausführer, Mindest- und Höchstwerte sollten auf denselben Wert festgesetzt werden. Für diese Szenarien ASG ermöglicht es Ihnen, die korrekte Anzahl der Ausführenden bei Ausfall zu halten.

Nachdem der Stack erstellt wurde, wird die angegebene Mindestanzahl von KNIME Executor Instanzen Sie sollten automatisch mit RabbitMQ auf dem Server verbunden werden.

Nach diesem Schritt sind die Auto Scaling KNIME Executors gebrauchsfertig.

KNIME beenden Ausführung

Wenn Sie mit diesem Setup aufhören möchten, löschen Sie einfach den Stack in der CloudFormation Abschnitt von AWS.



Dies bedeutet auch, dass alle Daten auf der Serverinstanz verloren gehen, es sei denn, es wurde vor der Löschung gesichert.

Operationen

Im Rahmen einer KNIME Server-Bereitstellung sollten Sie überlegen, Ihren Service für Verfügbarkeit. KNIME Server hat mehrere Endpunkte, mit denen das System ermittelt werden kann Gesundheit.

Anwendungsfehler

Ein einfacher REST-Aufruf auf den bereitgestellten KNIME Server sollte immer eine 200 Antwort mit einer Last ähnlich:

```
Curl https://knime/rest
```

rest_response

```
{
  "@controls": {\cHFFFF},
  "self": {\cHFFFF},
  "href": "https://knime/rest/",
  "Verfahren": "GET"
},
  "knime:v4": {\cHFFFF},
  "href": "https://knime/rest/v4",
  "Titel": "KNIME Server API v4",
  "Verfahren": "GET"
},
  "Version": {\cHFFFF},
  "major": B,
  "Minor": 8,
  "Revision": 0,
  "Qualifier": ""
},
  "mountId": "",
  "@namespaces": {\cHFFFF},
  "Knie": {\cHFFFF},
  "Name": "http://www.knime.com/server/rels#"
},
},
}
```

Eine andere Antwort gibt ein Konfigurationsproblem oder Anwendungsfehler an.

Es ist auch möglich, auf Executor-Availability zu testen. Dies erfordert eine Authentisierung gegen die KNIME Server und rufen den folgenden REST-Endpunkt an.

```
curl -X GET "https://knime/rest/v4/repository/Examples/Test Arbeitsabläufe  
- Test Basic Workflow - Daten  
Blending:execution?reset=true&timeout=300000" -H "Accept:application/vnd.mason+json"
```

AZ-Fehler

Seit KNIME Server Small/Medium läuft in einem einzigen AZ ein AZ Fehler wird durch die

Ein nachfolgend beschriebenes Applikationsfehlererkennungsverfahren.

[Kalt-Standby \(EFS\)](#) ein Beispiel für eine Architektur zu sehen,

Verfügbarkeit Zonenfehler.

Rechtssache

Ein Fehler kann mit AWS Cloudwatch erkannt werden, um die Instanzgesundheit zu überwachen. Vgl.

der Abschnitt

schuld.

Speicherkapazität

Sie können die Speicherkapazität der beiden EBS-Volumen (Wurzel und Daten) mithilfe von

Standardtechniken und Services wie AWS CloudWatch. Für weitere Details siehe

[Hier](#).

Wir empfehlen einen Alarm bei <5% Freiraum auf beiden Volumen auszulösen.

Ablauf der Sicherheitsbescheinigung

Zertifikatsablauf wird erwischt, wenn der Basisserver-Check mit einem HTTP 400-Status ausfällt

Code.

Backup und Recovery

Backup

KNIME Server kann unter den in der

[Verwaltungshandbuch](#) .

[KNIME Server](#)

Wichtige Datenstandorte im Abschnitt

<#page16>

Typischerweise ist die einfachste Backup-Lösung eine Snapshot des OS-Volumens zu nehmen, und eine zweite Snapshot des Datenvolumens.

Erholung

Bei der Verwendung der oben genannten 'Whole Volume Snapshot' Backup-Methode, Restaurierung der System wird am besten durch die Einführung einer neuen Instanz aus den Snapshot-Bildern.

Backup (AWS)

Es wird empfohlen, die AWS EBS Snapshot Funktionalität zu nutzen. Siehe die AWS

[Dokumentationsabschnitt](#)

[mit EBS Snapshots](#) .

Erholung (AWS)

Um einen AWS EBS Snapshot wiederherzustellen, siehe den AWS-Dokumentationsabschnitt auf

[Volumen](#) .

[Wiederherstellung von EBS](#)

Routine Wartung

Beginn der KNIME Server

KNIME Server startet automatisch, wenn die Instanz mit Standard systemd beginnt Befehle. Sobald die Tomcat-Anwendung erfolgreich gestartet wurde, wird sie automatisch starten und ausführen. Dies bedeutet, dass Sie im normalen Betrieb nicht die unten Befehl.

In dem Fall, dass Sie einen gestoppten KNIME Server starten müssen, kann es mit dem folgendes Kommando am Terminal:

```
sudo systemctl start knime-server.service
```

KNIME stoppen Server

KNIME aufhalten Server durch Ausführung des Befehls:

```
sudo systemctl stop knime-server.service
```

Neustart von KNIME Server

Neustart der KNIME Server durch Ausführung des Befehls:

```
sudo systemctl restart knime-server.service
```

Bootnote, für Versionen älter als KNIME Server 4.7

Beachten Sie, dass das Starten, Stoppen und Neustarten von Version 4.7 und älter von KNIME Server abweicht, wo knime-server. Service wurde durch apache-tomcat ersetzt.service

Neustart des Executors (KNIME Server Small/Medium/Large)

Es ist möglich, den Ausführenden neu zu starten, indem er den folgenden Befehl ausgibt:

```
sudo -u knime touch /srv/knime_server/rmirestart
```

Dies wird einen neuen Ausführenden starten, der den bestehenden Ausführenden laufen lässt. Alle vorhandenen Arbeitsplätze werden weiter auf dem alten Staatsanwalt laufen, und alle neuen Jobs werden auf den neuen Staatsanwalt gestartet.

Dies ist hilfreich bei der Aktualisierung von Executor-Präferenzdateien, ohne dass vorhandene Dateien unterbrechen müssen Jobs laufen. Wenn die rmirestart-Datei automatisch gelöscht wird, wurde der neue Executor gestartet.

Es ist möglich, eine harte Tötung auf einer laufenden Instanz durchzuführen, indem Sie den Befehl ausgeben:

```
sudo -u knime kill -9
```

wenn ist die Prozess-ID des laufenden Ausführenden. Sie können die durch Laufen:

```
ps aux | grep knime
```

und auf der Suche nach den Prozess(en), die nicht die apache-tomcat-Instanz sind.

Neustart des Ausführenden (KNIME Server Large - Distributed Executors)

In den meisten Fällen möchten Sie die gesamte Instanz neu starten, die den Executor betreibt. Aber in bestimmte Fälle, in denen Sie dies tun möchten, indem Sie die Executor-Anwendung selbst neu starten. Das kann entweder durch Stoppen des Ausführenden Prozesses und ab dem Terminal wieder abgeholt werden. Oder durch Neustart des systemd service, wenn es verwendet wird.

```
sudo systemctl restart knime-executor.service
```

Verwaltung von Zertifikaten

Detaillierte Schritte zur Verwaltung des SSL-Zertifikats für KNIME Server finden Sie im [KNIME Serververwaltung Leitfaden](#)

[KNIME](#)

Standardzertifikate

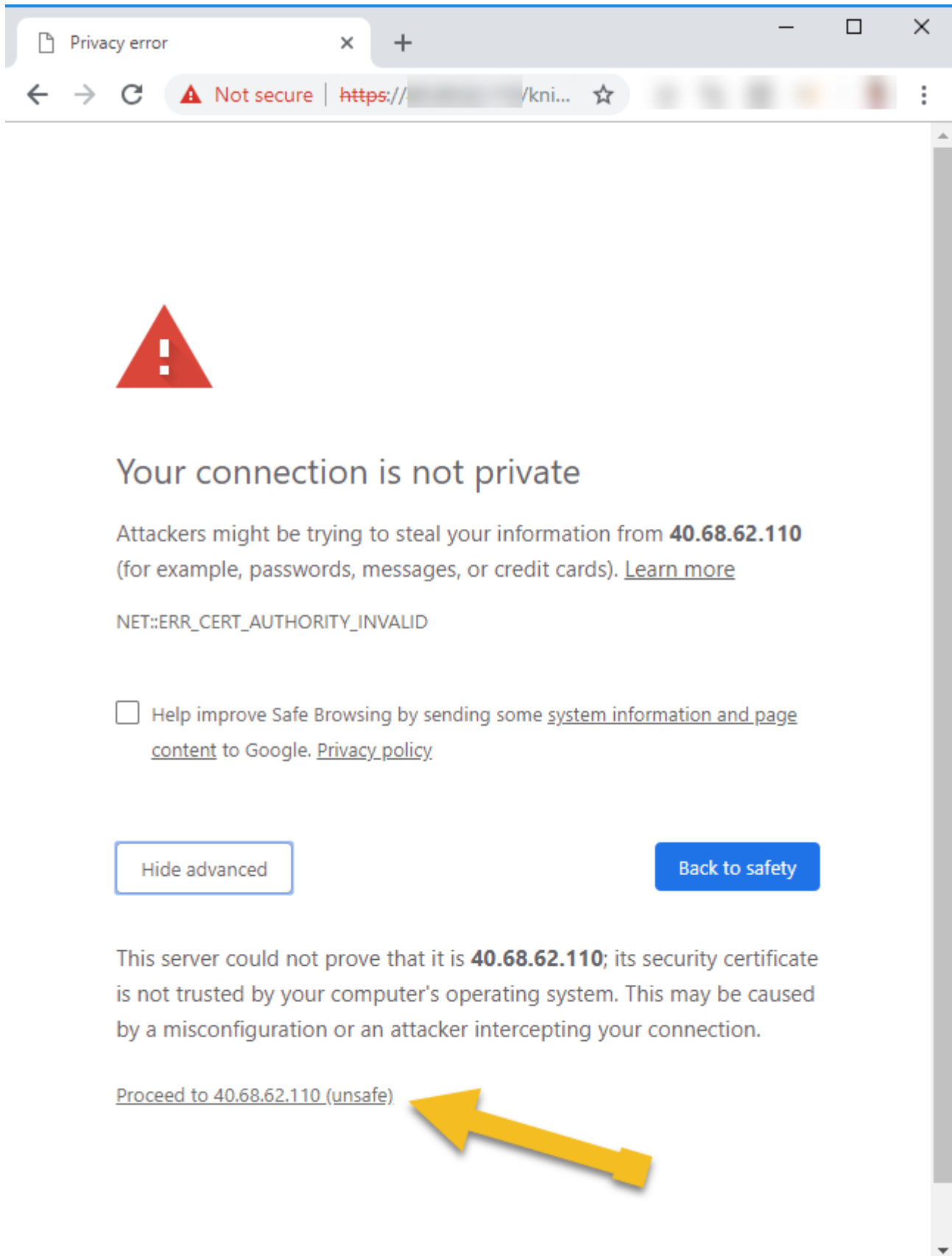
KNIME Serverschiffe mit Standard-SSL-Zertifikat. Dies ermöglicht eine verschlüsselte Kommunikation zwischen Client und Server. Da die Bescheinigung jedoch nicht im Voraus erstellt werden kann, der Server, auf dem Sie laufen, wird er nicht als gültiges Zertifikat anerkannt. Daher wir empfehlen, Ihr eigenes Zertifikat gemäß den Richtlinien in den

[#page39](#)

[Zertifikate](#page39)

Beim Testen mit dem Standardzertifikat werden moderne Browser wie unten eine Warnung ausstellen.

Mit der Wahl, die Warnung zu ignorieren, können Sie auf das KNIME WebPortal zugreifen.



Python Konfiguration aktualisieren

Wir verwenden Anaconda Python, um eine Standard-Python-Umgebung zu definieren. Die aktuelle yaml-Datei kann in `/home/knime/python/py36_knime.yaml` gefunden werden.

Detaillierte Dokumentation zur Verwaltung von Anaconda finden Sie unter [Anaconda Dokumentation](#).

Im Folgenden wird eine beispielhafte yaml-Datei angezeigt. Wir wählen Pakete, die wir wissen, sind gut verwendet, oder sind für die Verwendung der [KNIME Tiefen lernen](#) Paket. Meistens haben wir gepinnte Version Zahlen, um Kompatilität zu gewährleisten. Sie können wählen, um die Versionsnummern zu deaktivieren. Zusätzlich kann ein neues Python-Paket hinzufügen möchten, in welchem Fall Sie das Paket dem yaml hinzufügen können Datei und den Befehl ausführen:

```
sudo -u knime /home/knime/python/anaconda/bin/conda env update -f
/home/knime/python/py36_knime.yaml --prune
```

[py36_knime.yaml](#), [Quelle](#), [yaml](#)

Apply Betriebssystempatches

Die KNIME Server 4.18 AMIs basieren auf Ubuntu Server 20.04 LTS. Das Betriebssystem sollte regelmäßig mit den Standard-Ubuntu-Verfahren gepatcht.

Nach einem Java JDK-Update muss der KNIME Server neu gestartet werden.

Update KNIME Server

Updates und Patches zum KNIME Server werden auf dem KNIME Server Forum bekannt gegeben. Du [kannst dich abonnieren](#) [Thema](#).

Vor der Anwendung eines Feature-Updates (z.B. Version 4.8.2 → 4.9.0) Sie sollten die [KNIME Server Release Notes und Update Guide](#). Dies wird Änderungen der Parameter dokumentieren, Funktionen und Einstellungen, die Ihre Installation beeinflussen könnten.

Bei einem Patch-Update (z.B. Version 4.8.1 → 4.8.2) keine Änderungen an Einstellungen erforderlich.

Es gibt zwei Strategien zur Anwendung von Features oder Patch-Updates von KNIME Server. Der erste ist: befolgen Sie die Anweisungen im KNIME Server Update Guide über das Terminal (in place update). Die zweite ist, einen Snapshot des Workflow-Repository-Blockgeräts auf ein neues KNIME zu migrieren Server-Instanz (Swap-Update deaktivieren).

In place update

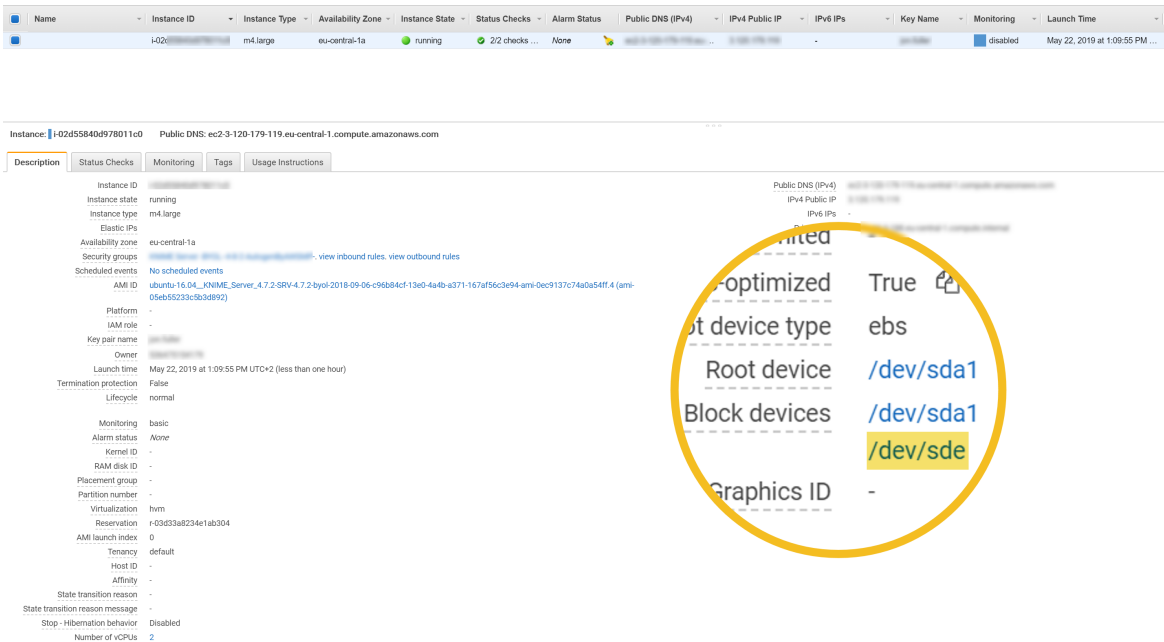
Um ein Feature-Update zu machen, haben Sie die Möglichkeit, die Anweisungen in der [Update Guide](#)

[KNIME Server](#)

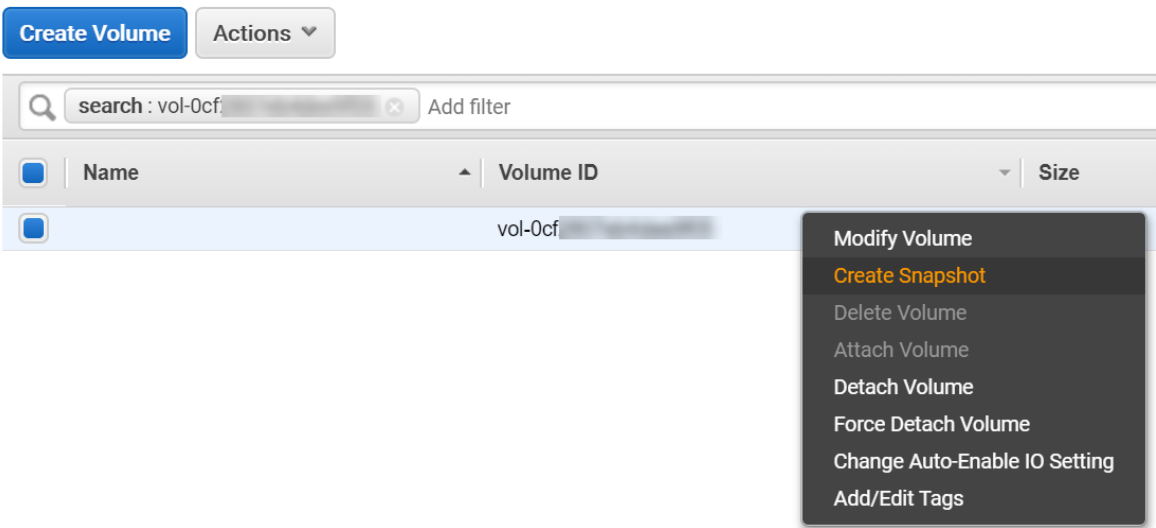
Disk Swap Update (AWS)

- Melden Sie sich an den vorhandenen Server und stoppen Sie den KNIME Server-Service. Siehe Abschnitt [KNIME Server](#)
- Erstellen Sie einen Snapshot des Workflow-Repository-Blockgeräts.

Finden Sie das Blockgerät, das Schnappschüsse benötigt (in der AWS EC2 Konsole)



Erstellen Sie einen Snapshot des Blockgeräts



Geben Sie dem Snapshot einen Namen

Volumes > Create Snapshot

Create Snapshot

Volume

vol-0cf

Description

update-knime-server

Encrypted

Not Encrypted

Key

(127 characters maximum)

Value

(255 characters maximum)

This resource currently has no tags

Choose the Add tag button or [click to add a Name tag](#)

Add Tag

50 remaining

(Up to 50 tags maximum)

* Required

Cancel

Create Snapshot

- Starten Sie eine neue Instanz von KNIME Server vom AWS Marketplace, indem Sie die zuvor erstellt Snapshot.

☐ Neue Instanz auf der AWS-Konsole starten

Open for Innovation

KNIME

KNIME Server Medium for AWS

Use the KNIME Analytics Platform to automate advanced analytics, machine learning, or data prep/ETL tasks. Productionize your data science applications using the KNIME Server Medium for AWS and extend your analytical applications to your entire organization.

KNIME complements Amazon AWS services such as SageMaker, Kinesis, RedShift, Comprehend, ...

[More info](#)

[View Additional Details in AWS Marketplace](#)

Product Details

By

KNIME.com

Customer Rating

★★★★★ (0)

Latest Version

4.8.2

Base Operating System

Linux/Unix, Ubuntu 18.04 LTS

Delivery Method

64-bit (x86) Amazon Machine Image (AMI)

License Agreement

[End User License Agreement](#)

On Marketplace Since

10/17/16

AWS Services Required

Amazon EC2, Amazon EBS

Highlights

Pricing Details

Hourly Fees

Instance Type	Software	EC2	Total
t2.large	\$3.973	\$0.107	\$4.08/hr
t2.xlarge	\$3.973	\$0.214	\$4.187/hr
t2.2xlarge	\$3.973	\$0.429	\$4.402/hr
t3.large	\$3.973	\$0.096	\$4.069/hr
t3.xlarge	\$3.973	\$0.192	\$4.165/hr
t3.2xlarge	\$3.973	\$0.384	\$4.357/hr
m5a.large	\$3.973	\$0.104	\$4.077/hr
m5a.xlarge	\$3.973	\$0.208	\$4.181/hr
m5a.2xlarge	\$3.973	\$0.416	\$4.389/hr
m5d.large	\$3.973	\$0.136	\$4.109/hr
m5d.xlarge	\$3.973	\$0.272	\$4.245/hr
m5d.2xlarge	\$3.973	\$0.544	\$4.517/hr
m5d.metal	\$3.973	\$6.528	\$10.501/hr
m5.large	\$3.973	\$0.115	\$4.088/hr
m5.xlarge	\$3.973	\$0.23	\$4.203/hr
m5.2xlarge	\$3.973	\$0.46	\$4.433/hr
m4.large	\$3.973	\$0.12	\$4.093/hr
m4.xlarge	\$3.973	\$0.24	\$4.213/hr

Cancel

Continue

☐ Zuweisen Snapshot zu neuer Instanz

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-0c3e8f872eb44a6d	50	General Purpose SSD (gp2)	150 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted
EBS	/dev/sde	knime	250	General Purpose SSD (gp2)	750 / 3000	N/A	<input type="checkbox"/>	<input type="checkbox"/>

Add New Volume

Description

update-knime-server

Snapshot ID

snap-096935a01f...

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

- Für Updates von Marktplatzbildern Version 4.7.2 ist es erforderlich, die Anweisungen im folgenden Abschnitt.

☐

Anmerkung zum Upgrade auf 4.8.2 +

Zur besseren Unterstützung neuerer EC2-Instanztypen wie der m5, c5 und t3 Serie

Fälle, in denen NVMe-Blockgeräte verwendet wurden, waren einige Änderungen erforderlich.

Das bedeutet, dass bei der Aktualisierung von einer älteren Instanztyp.

Nicht durchzuführen, wird bedeuten, dass die Tomcat-Management-Seite auf

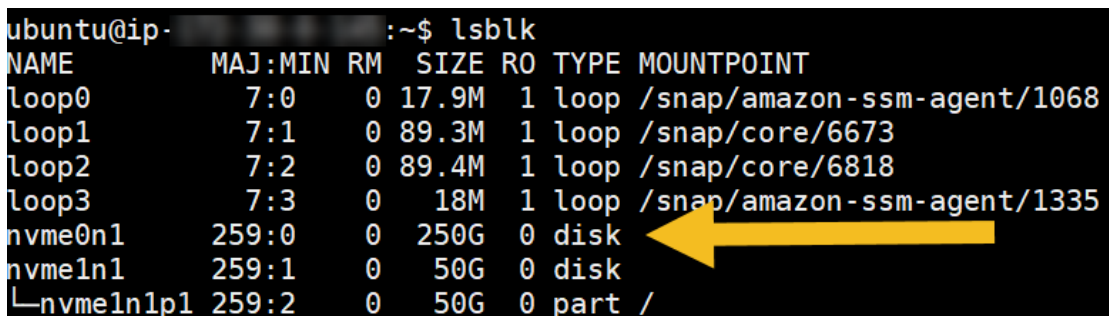
`http://` wird zugänglich sein, aber der KNIME Server bei

`https://hostname/knime` nicht zugänglich sein.

Details zu NVMe Geräten auf AWS sind verfügbar

[Hier](#).

Lsblk



```
ubuntu@ip-:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0        7:0    0 17.9M  1 loop /snap/amazon-ssm-agent/1068
loop1        7:1    0 89.3M  1 loop /snap/core/6673
loop2        7:2    0 89.4M  1 loop /snap/core/6818
loop3        7:3    0  18M   1 loop /snap/amazon-ssm-agent/1335
nvme0n1      259:0    0 250G   0 disk
nvme1n1      259:1    0  50G   0 disk
└─nvme1n1p1  259:2    0  50G   0 part /
```

Aus dem Screenshot können Sie das Blockgerät identifizieren, das in der Form sein wird

`nvmeXnY`. Hier ist es `nvme0n1`, die dem 250GB-Gerät entspricht.

Um das Volumenlabel einzustellen, montieren Sie das Workflow-Repository vorübergehend, werde sein

sicher zu bearbeiten `Nvme Xn Y` auf den Wert aus dem vorherigen Schritt.

```
sudo mount -t /dev/nvmeXnY /test
```

Anschließend das 'knime-repo'-Volumenlabel auf das Blockgerät auftragen. Genau

'knime-repo' wird bedeuten, dass Sie diesen Schritt in späteren

Aktualisierungen.

```
sudo e2label /dev/nvmeXnY knime-repo
```

Wenn Sie ein Volume-Label verwenden, das anders ist als 'knime-repo', dann müssen Sie auch bearbeiten

`/etc/fstab`, um das entsprechende Volumenlabel zu verwenden, um das Blockgerät zu montieren

`/srv`.

Bearbeiten `/srv/knime_server/config/knime-server.config`

die bestehenden

Zeile zu:

```
com.knime.server.executor.knime_exe = /opt/knime/knime-3.7.2/knime
```

Starten Sie den Server neu, um die Änderungen anzuwenden.

```
sudo shutdown -r jetzt
```

SSH Zugriff auf KNIME Server auf AWS

Zugang zur KNIME Server-Instanz über SSH folgt der [Allgemeine Hinweise](#) bereitgestellt durch AWS.

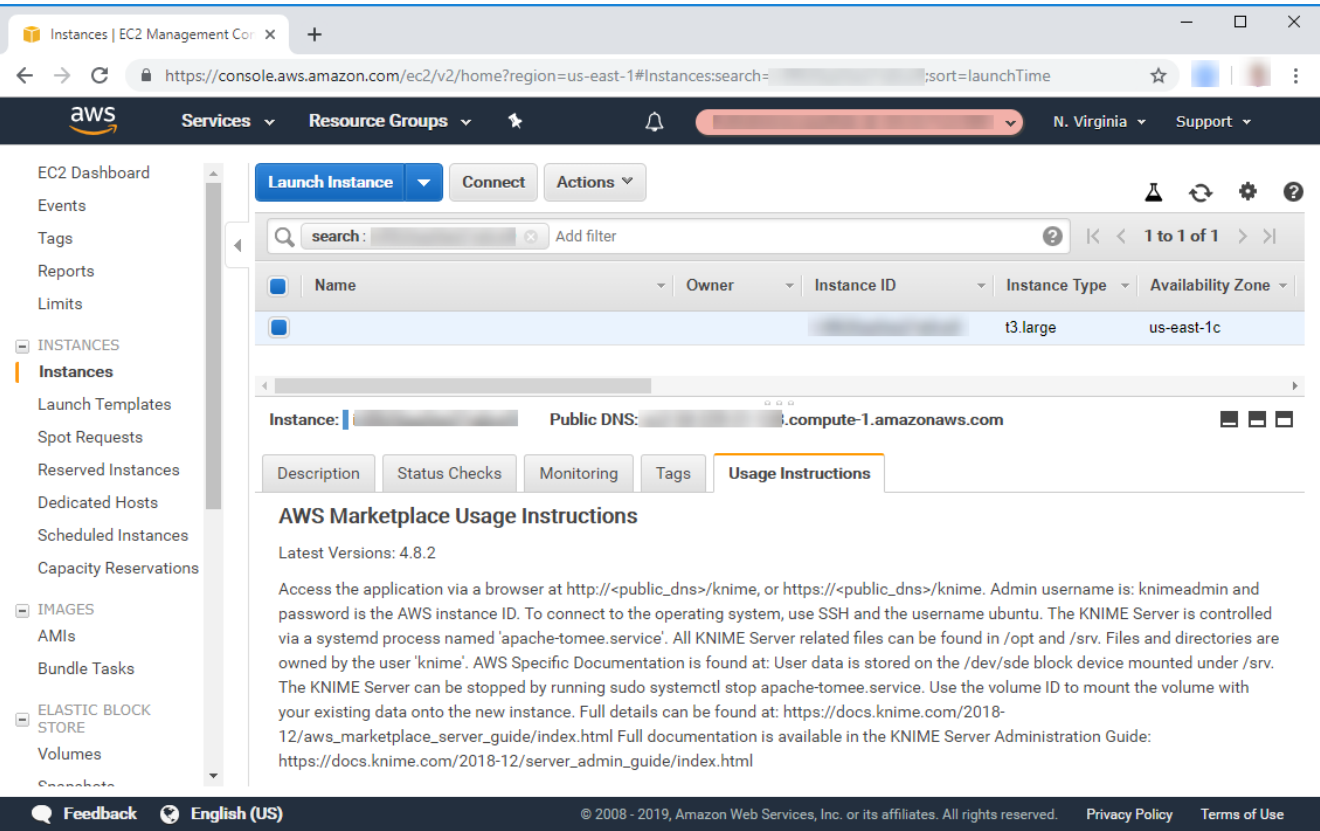
Verbindung zum KNIME Server-Instanz verwendet immer die `ubuntu` Benutzer und SSH-Taste angegeben zum Beispiel Startzeit. Ein Beispiel SSH-Verbindungsstring ist:

```
Ssh -i
```

Alle relevanten KNIME Server-Installation und Laufzeitdateien gehören zu den `knimeadmin` Benutzer. In Ordnung Um Änderungen an diesen Dateien vorzunehmen, ist es erforderlich, die Identität der `knimeadmin` Benutzer:

```
sudo su knime
```

Gebrauchsanweisungen sind immer im Rahmen der AWS-Konsole enthalten [Gebrauchsanweisung](#) Tab.



Skalierung des KNIME Server-Instanz

Falls Ihre KNIME Server-Instanz dauerhaft bei hoher CPU-Last läuft, werden Sie kann auf eine größere Instanz Größe skalieren wollen. Umgekehrt kann es der Fall sein, dass Sie begonnen haben mit einer zu großen Instanzgröße. In beiden Fällen ist es möglich, den KNIME Server zu stoppen und erneut mit einem anderen Instanztyp starten.

Dazu sollten Sie den Anweisungen folgen [Hier.](#) .

Erhöhung der Workflow Repository EBS Volumenkapazität

Es ist möglich, die Kapazität des Workflow-Repository EBS-Volume zu erhöhen (Standardgröße: 250 Gb) nach dem Start einer Instanz. Folgen Sie den Anweisungen [Hier.](#) .

Eine Änderung der Größe des Wurzelvolumens sollte nicht erforderlich sein. Das Verfahren ist schwieriger als die Größe des Workflow-Repository EBS-Volumes zu erhöhen.

Schlüsseldrehung

Verwaltung von SSH-Tasten für den Zugriff auf den KNIME Server [Hier.](#) .

ServiceLimits

Verwaltung von AWS-ServiceLimits [Hier.](#) .

Notwendige Wartung

Wenn KNIME Server REST API nicht verfügbar ist, ist ein Neustart des Tomcat Servers erforderlich.

Wenn die REST API verfügbar ist, aber die Ausführung API nicht wie beabsichtigt funktioniert, dann der Executor muss zuerst neu gestartet werden, und wenn das nicht funktioniert, dann wird Tomcat neu gestartet. erforderlich.

[Sichere AZ's Routine-Wartung](#page36)
für Details.

Notfallwartung (AWS)

Fall KNIME Server ist aufgrund der abgebauten Leistung einer Verfügbarkeitszone nicht verfügbar (AZ), EC2-Instanzfehler usw. Es ist möglich, einen Snapshot wiederherzustellen und eine neue Instanz zu starten.

AZ Erholung

Die AZ-Recovery wird durch die Einführung einer neuen Instanz in ein unangetastetes AZ mit einem [Momentaufnahme](#page37)

Dann befestigen Sie die elastische IP von der betroffenen Instanz an die neue Instanz.

Erholung der Region

Die Wiederherstellung der Region wird durch die Einführung einer neuen Instanz in eine unangetastete Region, mit einer [aktuelles foto](#page37)

Dann befestigen Sie die elastische IP von der betroffenen Instanz an die neue Instanz.

Fehlerbehebung

Um Probleme mit der KNIME Server-Instanz zu beheben, wenden Sie sich bitte an

[KNIME Leitfaden für die Verwaltung von Servern](#)

und

[Amazon EC2 Dokumentation zur Fehlerbehebung](#)

.

Unterstützung

KNIME Server Kleine Unterstützung wird durch die Einreichung von Fragen in der

[KNIME Server Forum](#).

KNIME Server Medium und KNIME Server Large Support werden zusätzlich über

Kontakt der Support@knime.com E-Mail-Adresse.

Bei Kontakt mit KNIME Unterstützung, die Sie benötigen, um Ihre Produkt-Code, Instanz-ID, und AWS Account ID. Wir wollen Ihre Frage in weniger als 48 Stunden beantworten.

Finden Sie Ihre Produktdetails.

Finden Sie Ihre Produkt-Code, Instanz-ID und AWS-Konto-ID:

Die [AWS Dokumentation](#) erklärt, wie man Zugang zu den EC2-Metadaten erhält, die die Informationen zu Ihrer Instanz. Sie können auch diese Informationen aus dem EC2 Web bestimmen Managementkonsole.

Kosten für die Unterstützung

Wenn Sie zusätzliche Unterstützung benötigen, kontaktieren Sie bitte Verkauf@knime.com für weitere Informationen.

KNIME AG
Talacker 50
8001 Zürich, Schweiz
www.knime.com
Info@knime.com