Engineer

Deployment-Rollback

Rückkehr zur vorherigen Bereitstellung nach einem Fehler

Was es ist

Ein Deployment-Rollback setzt nach einer problematischen Änderung eine frühere bereitgestellte Version oder Konfiguration wieder ein. Was dabei geschieht, hängt vom System ab. Eine Plattform kann den Datenverkehr auf die vorige Version zurückschalten. Ein Orchestrator kann neue Anwendungsinstanzen mit einer gespeicherten Revision starten. Beim Rollback eines Kubernetes Deployments wird die Pod-Vorlage einer früheren Revision wiederhergestellt. Die Datenbank und andere abhängige Systeme laufen auf ihrem eigenen Stand weiter.

Ein Revert im Quellcode erzeugt eine neue Revision, die die Wirkung einer Änderung rückgängig macht. Damit diese Revision im Dienst ankommt, muss man sie bauen und bereitstellen. Ein Rollback kann ein Paket einsetzen, das bereits vorliegt. Nach dem Eingriff muss der deklarierte Sollzustand wieder mit dem beabsichtigten Betrieb zusammenpassen. Andernfalls kann die nächste normale Bereitstellung versehentlich erneut die problematische Version aktivieren.

Auch nach der Rückkehr können Datensätze, Dateien und Ereignisse vorhanden sein, die die neuere Version geschrieben hat. Wenn das frühere Programm sie lesen kann, bleiben aktuelle Eingaben erhalten. Eine Wiederherstellung der Datenbank auf einen früheren Zeitpunkt ist eine eigene Maßnahme. Sie braucht eine Entscheidung über den gewünschten Datenstand und eigene Kontrollen. Beschreiben Sie vorab, welche Teile zurückgesetzt werden und welche Daten weiter verfügbar sein müssen.

Das Konzept für eine Entscheidung nutzen

Nutzen Sie „Was ist ein Deployment-Rollback?“ für eine konkrete Entscheidung. Wir können helfen, das Konzept mit dem System in Verbindung zu bringen, das Sie entwickeln oder einkaufen. Geprüft, veröffentlicht und rückgängig zu machen: Änderungen erreichen Ihre Kunden am Dienstagnachmittag statt spät am Freitagabend.

Unsere Leistungen entdecken Kontakt aufnehmen →

Die Annahmen hinter dem Begriff prüfen

Nutzen Sie die Erklärung und die verlinkten Arbeiten, um Fragen für Ihr Projekt zu formulieren. Wir können die Annahmen gemeinsam mit Ihnen prüfen, bevor ein Werkzeug ausgewählt wird. GIS-SaaS-Deployments mit GitHub automatisiert — Bereitstellung, Daten und Berichte ohne Handarbeit

GIS-SaaS-Deployments mit GitHub automatisiert Kontakt aufnehmen →

Warum es wichtig ist

Wenn nach einer Bereitstellung Fehler auftreten, muss das Team entscheiden, welche Aktion den Dienst wieder funktionsfähig macht. Ein verfügbares älteres Paket und eine geübte Umschaltung bieten eine Möglichkeit. Sie kann schneller sein, als unter Zeitdruck einen Fix zu bauen. Das Zurückschalten kann neue Fehler verhindern, während bereits betroffene Daten eine weitere Korrektur brauchen. Benennen Sie das Ziel der Maßnahme, damit diese beiden Arbeiten erkennbar bleiben.

Eine gestartete Anwendung ist ein schwaches Signal für die vollständige Wiederherstellung. Prüfen Sie einen Vorgang, den die Nutzerinnen und Nutzer tatsächlich brauchen. Wenn der Bestellprozess betroffen war, kann eine kontrollierte Bestellung mit Testkonto helfen. Prüfen Sie, welche Version den Aufruf beantwortet und was die Verarbeitungswarteschlange meldet. Geben Sie seltenen Fehlern ausreichend Zeit, wieder aufzutreten, oder verwenden Sie einen passenden Testfall.

Die Rückkehr hängt von gespeicherten Paketen und gültigen Zugängen ab. Eine Plattform kann eine frühere Revision anzeigen, obwohl das referenzierte Container-Image gelöscht wurde. Auch ein Zertifikat oder eine Berechtigung kann fehlen. Bewahren Sie die Voraussetzungen für den vereinbarten Wiederherstellungszeitraum auf. Üben Sie mit den Rechten und Werkzeugen, die im Bereitschaftsdienst tatsächlich zur Verfügung stehen.

Wie es in der Praxis aussieht

Das Fallbeispiel zu wiederholbaren Bereitstellungen in Azure beschreibt einen festen Weg durch Entwicklung, Test und Produktion. Eine bekannte Abfolge erleichtert die Planung der Wiederherstellung, weil das Team die betroffenen

Versionen und Umgebungen benennen kann. Das Fallbeispiel beschreibt den Weg zur Bereitstellung. Das folgende Beispiel stellt einen eigenständigen möglichen Ablauf der Wiederherstellung dar.

Angenommen, Version B eines Bestelldienstes verursacht Fehler beim Einsatz gespeicherter Adressen. Version A funktionierte unmittelbar zuvor. Das Paket und seine Einstellungen sind noch verfügbar. Prüfen Sie zuerst, ob B Daten geschrieben hat, die A nicht lesen kann. Sind beide mit dem aktuellen Zustand kompatibel, können Sie den Verkehr wieder zu A leiten. Führen Sie eine kontrollierte Bestellung aus. Beobachten Sie den öffentlich erreichbaren Dienst und Fehler in der Warteschlange und prüfen Sie die Kennung der Version, die die Anfrage verarbeitet hat. Prüfen Sie zusätzlich, ob ein bereits vor der Umschaltung begonnener Auftrag später genau einmal abgeschlossen wird. So fällt auf, wenn die früheren Hintergrundprozesse einen Zustand anders deuten als die neue Anwendung.

Wenn B dagegen eine von A benötigte Spalte entfernt hat, kann der Wechsel zusätzliche Fehler erzeugen. Eine Möglichkeit besteht darin, die betroffenen Schreibvorgänge anzuhalten und eine Korrektur bereitzustellen, die mit dem vorhandenen Schema funktioniert. Planen Sie Änderungen so, dass zuerst eine neue Struktur hinzugefügt werden kann. Stellen Sie lesende und schreibende Komponenten schrittweise um. Entfernen Sie die alte Struktur erst, wenn sie keine zu unterstützende Version mehr braucht. Prüfen Sie während des Übergangs beide Versionen.

Wo es schiefgeht

Eine erfolgreiche Verwaltungsaktion besagt zunächst, dass das System einen Auftrag angenommen hat. Prozesse können noch im Start sein, und der Datenverkehr kann noch einem alten Pfad folgen. Prüfen Sie die tatsächlich laufenden Instanzen anhand der vereinbarten fachlichen Kriterien. Bleibt der Fehler bestehen, muss das Team die Diagnose erneut öffnen.

Funktionsschalter, Verkehrsregeln und geplante Aufgaben können unabhängig vom installierten Paket wirken. So kann neuer Code abgeschaltet worden sein, während ein Hintergrundprozess die neue Schreibvariante weiter verwendet. Ordnen Sie die beteiligten Steuerungen den zuständigen Personen zu und prüfen Sie die Reihenfolge. Berücksichtigen Sie auch Clients und Prozesse, die nach der Umschaltung noch Anfragen im neueren Format senden.

Ein automatisch abgeglichener Sollzustand kann die problematische Version erneut aktivieren. Stimmen Sie die Änderung der deklarierten Version mit der Wiederherstellung von Paket und Verkehr ab. Erfassen Sie die Versionskennungen, die maßgeblichen Zeiten und die Folgen für die Nutzer. Sobald eine Korrektur bereitsteht, kann das Team diese wieder über den normalen kontrollierten Ablauf bereitstellen.

Weiterführende Lektüre

- Kubernetes über Deployments und Rollbacks erklärt Revisionen und den Umfang der Pod-Vorlage.
- DORA über Bereitstellungsmetriken beschreibt Lieferfluss und Wiederherstellung.
- PostgreSQL über Tabellenänderungen dokumentiert Änderungen am Schema.
- Helm rollback zeigt die Auswahl einer früheren Revision.

Glossar

Für Ihre tatsächlichen Rahmenbedingungen auswählen

Beschreiben Sie den Bereitstellungsweg, die Betriebsrisiken und die Bereitschaftsverantwortlichen. Wir können eine Änderung mit Prüfungen, einem Wiederherstellungsweg und klaren Zuständigkeiten abgrenzen.

Preise ansehen Kontakt aufnehmen →

Die Begriffe praktisch nutzbar machen

Fragen Sie, wie „Was ist ein Deployment-Rollback?“ auf Ihre Organisation zutrifft. Ein konkretes Beispiel aus Ihrem bestehenden System reicht aus, um das Gespräch zu beginnen.

Unsere Leistungen entdecken

welcome@engineer.company +45 50 10 61 84 Kopenhagen, Dänemark

Nehmen Sie diese Seite mit

Kontaktkarte

Mit Sorgfalt und Liebe gestaltet von D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – heute · Engineer ApS · Alle Rechte vorbehalten.

Gedruckt von https://engineer.company/de/glossary/deployment-rollback/