Präsentiert von

veeam

Datensicherung und Wiederherstellung für Kubernetes

für dummies®
A Wiley Brand

- Warum Kubernetes-native Backups wichtig sind/
- Informationen zum Schutz Ihrer Kubernetes-Anwendungen/
- Erstellung von automatisierten Kubernetes Backup-Richtlinien

Steve Kaelble

3. Sonderausgabe von Veeam®

Über Veeam

Veeam ist ein führender Anbieter von Backup- und Disaster-Recovery-Lösungen für Kubernetes. Veeam Kasten hilft Unternehmen dabei, die Herausforderungen des Backup-Managements von „Day 2"-Operationen zu meistern, um Anwendungen sicher auf Kubernetes auszuführen. Weitere Informationen finden Sie unter https://www.veeam.com/products/cloud/kubernetes-data-protection.html?ad=menu-solutions. Lesen Sie die neuesten Pressemitteilungen zu Veeam Kasten unter http://docs.kasten.io. Folgen Sie Veeam auf X unter https://twitter.com/veeam.

Datensicherung und Wiederherstellung für Kubernetes

3. Sonderausgabe von Veeam®

Steve Kaelble

für dummies®
A Wiley Brand

Datensicherung und Wiederherstellung für Kubernetes für Dummies®, 3. Sonderausgabe von Veeam®

Veröffentlicht von
John Wiley & Sons, Inc.
111 River St., Hoboken, NJ 07030-5774
www.wiley.com

ISBN 978-1-394-35365-1 (pbk); ISBN 978-1-394-35366-8 (ebk); ISBN 978-1-394-35367-5 (ePub)

Allgemeine Informationen zu unseren anderen Produkten und Dienstleistungen oder zur Erstellung eines individuellen *Für Dummies*-Buches für Ihr Unternehmen oder Ihre Organisation erhalten Sie von unserer Abteilung Business Development in den USA unter Tel. 877-409-4177, E-Mail info@dummies.biz oder auf www.wiley.com/go/custompub. Für Informationen zur Lizenzierung der *Für Dummies*-Marke für Produkte oder Dienstleistungen kontaktieren Sie bitte BrandedRights&Licenses@Wiley.com.

Danksagung des Verlags

Die folgenden Personen haben bei der Erstellung dieses Buches mitgewirkt:

Development Editor: Rebecca Senninger

Acquisitions Editor: Traci Martin

Editorial Manager: Rev Mengle

Business Development Representative: Matt Cox

Production Editor: Saikarthick Kumarasamy

Inhaltsverzeichnis

Einführung

Niemand muss Ihnen sagen, dass der Cloud die Zukunft gehört. Die Akzeptanz und Nutzung cloudbasierter Anwendungen steigt kometenhaft an (Entschuldigung, das Wortspiel war beabsichtigt). Kubernetes ist überall. Die Lösung ist auf dem besten Weg, zur bevorzugten Enterprise-Plattform und zur Grundlage der unterschiedlichsten Anwendungen zu werden – und das aus gutem Grund. Sie ist portabel, agil, skalierbar und äußerst zuverlässig, was sie zum Traum eines jeden Entwicklers macht.

Eines ist sie jedoch *nicht*: der Heilige Gral für den Schutz Ihrer Daten. Einige der Vorzüge der Architektur bringen nämlich auch neue Herausforderungen in Bezug auf das Management und den Schutz von Daten mit sich. Der Schutz, den sie Ihren Anwendungen bietet, erstreckt sich nicht automatisch auf Ihre Daten, und traditionelle Backup-Architekturen lassen sich nicht einfach für ein cloudbasiertes Ökosystem nachrüsten.

Deshalb brauchen Sie eine echt cloudnative Backup-Lösung, die nicht nur die Sprache von Kubernetes spricht, sondern direkt in dieser neuen und aufregenden Welt zu Hause ist. Und Sie müssen die Best Practices kennen, mit denen Sie eine reibungslose Umstellung auf die Nutzung und Einführung cloudbasierter Anwendungen erreichen können.

Über das Buch

Die Veeam-Sonderausgabe *Datensicherung und Wiederherstellung für Kubernetes für Dummies* ist Ihr Handbuch für diese Lösung. Es vermittelt Ihnen die Grundlagen und erklärt, wie Kubernetes entstanden ist. Es zeigt Ihnen klar und deutlich, warum Ihr Backup morgen nicht mehr so funktionieren kann wie gestern.

Nach der Lektüre dieses Buches werden Sie ein besseres Verständnis für die Erstellung und Ausführung der Backups Ihrer Kubernetes-Anwendungen haben. Unter anderem finden Sie anwendungsorientierte Ansätze rund um die Themen Wiederherstellbarkeit, Gewährleistung der Sicherheit, Arbeiten in mandantenfähigen Umgebungen, Unterstützung bei der Wiederherstellung und optimale Nutzung der cloudbasierten Anwendungsmobilität.

Dieses Buch steckt voller nützlicher Tipps zum Schutz Ihrer Daten in einem sich ständig ändernden Ökosystem. Gleichzeitig erfahren Sie, wie Sie Entwicklern und Bedienern das Leben erleichtern können. Es zeigt Ihnen auch, wie Sie von einer revolutionären Veränderung profitieren können, ohne das Boot zu sehr zum Schaukeln zu bringen oder es in die falsche Richtung zu steuern.

Leichtfertige Annahmen

Beim Verfassen dieses Buches haben wir einige Annahmen über Sie, den Leser dieser Seiten, getroffen.

- Möglicherweise sind Sie eine technische Fachperson, die sich mit DevOps und cloudbasierten Anwendungen auskennt.
- Vielleicht haben Sie es eher mit geschäftlichen Belangen zu tun, interessieren sich jedoch für die Möglichkeiten von cloudbasierten Lösungen oder sind besorgt wegen ihrer Risiken.
- Auf jeden Fall setzen Sie sich dafür ein, dass Ihr Unternehmen in diesem hochdynamischen Bereich erfolgreich ist.

In diesem Buch verwendete Symbole

Um Ihnen die Navigation in diesem Buch zu erleichtern, haben wir an den Rändern einige Symbole eingefügt. Diese Symbole sollen als Wegweiser dienen, die Sie auf besonders wichtige Passagen hinweisen.

Sie können gerne von einem Kapitel zum nächsten springen, doch bitte schenken Sie diesen Abschnitten, die besonders wichtige Informationen enthalten, besondere Aufmerksamkeit.

Wir haben Ihnen nützliche Informationen versprochen. Neben diesem Symbol finden Sie einige besonders hilfreiche Ideen.

Die Datensicherung kann mitunter eine Herausforderung sein. Dieses Symbol weist auf Bereiche hin, in denen besondere Vorsicht geboten ist, um Probleme zu vermeiden.

In diesen Abschnitten wird näher auf die Details eingegangen. (Wenn Sie technische Fakten mögen, sind diese Abschnitte genau das Richtige für Sie!)

Zusätzliche Informationen

Dieses Buch soll zum Nachdenken anregen und als eine Einführung in einige wichtige Konzepte dienen. Das Ganze ist mit vielen interessanten Details gespickt. Wenn Sie das Buch verschlungen haben und Lust auf mehr haben, finden Sie unter `https://www.veeam.com/products/cloud/kubernetes-data-protection.html` zusätzliche Ressourcen, Hintergrundinformationen und White Papers.

IN DIESEM KAPITEL

- Cloud- und containerbasierte Anwendungen
- Warum Kubernetes?
- Die Vorteile auf den Punkt gebracht
- Mythen rund um Kubernetes

Kapitel 1
Kubernetes und cloudbasierte Anwendungen

Der mittlerweile geschützte iPhone-Werbeslogan von Apple „There's an app for that" ist ein geflügeltes Wort geworden. Und hat sich als äußerst vorausschauend erwiesen: Für eine zunehmende Anzahl alltäglicher Aufgaben gibt es jetzt eine App, die Verbrauchern und Unternehmen das Leben erleichtert. App-Entwickler entscheiden sich heute zunehmend für containerbasierte Architekturen und stellen ihre Anwendungen in der Kubernetes-Umgebung bereit.

In diesem Kapitel betrachten wir die zunehmende Nutzung von container- und cloudbasierten Anwendungen und erläutern, wie Kubernetes so erfolgreich wurde. Es beinhaltet die Vorteile einer Kubernetes-nativen Architektur und räumt mit einigen Mythen auf, die sich um die mittlerweile effektivste Methode zur Bereitstellung moderner Geschäftsanwendungen ranken.

Der Aufstieg cloudbasierter Anwendungen

Wo fügen sich containerbasierte Anwendungen derzeit in Ihr Unternehmen ein? Was hält die Zukunft bereit? Aus Studien geht hervor, dass cloud- und containerbasierte Technologien in den letzten Jahren einen enormen Aufschwung erlebt haben.

Kubernetes hat sich dank seiner Fähigkeit, die Bereitstellung, Skalierung und Verwaltung containerbasierter Anwendungen zu automatisieren, inzwischen weitreichend etabliert. Laut einer ESG-Umfrage vom April 2023 ist Kubernetes reifer geworden: 66 Prozent der Befragten gaben an, dass sie Kubernetes bereits zur Verwaltung und Orchestrierung ihrer Container verwenden (https://go.veeam.com/wp-ar-enterprise-kubernetes-protection).

Die jährliche Studie der Cloud Native Computing Foundation (https://www.cncf.io/reports/cncf-annual-survey-2023/) aus dem Jahr 2023 kommt zu ähnlichen Ergebnissen. Demnach wird Kubernetes von 84 Prozent der Befragten genutzt oder evaluiert – ein Beleg dafür, dass sich die Plattform als Kerntechnologie etabliert hat. Auch in Verbindung mit Hybrid Cloud-Implementierungen erfreut sich Kubernetes zunehmender Beliebtheit.

Auf die Frage nach der größten Herausforderung führten 46 Prozent der Teilnehmer, die noch keine Container in der Produktion einsetzten, und 28 Prozent, die sie nur in begrenztem Umfang nutzten, das fehlende Schulungsangebot an. Bei Unternehmen, in denen Container bereits für die meisten Anwendungen genutzt werden, rückt die Sicherheit als größte Herausforderung an die erste Stelle.

Wie es um die Anwendungsbereitstellung in Kubernetes (stateful oder stateless) bestellt ist, geht aus dem Data on Kubernetes 2021 Report (https://dok.community/dokc-2021-report/#:~:text=Key%20Findings,2x%20or%20greater%20productivity%20gains) klar hervor. 90 Prozent der Befragten waren der Meinung, dass Kubernetes für Stateful-Workloads geeignet sei. Eine überwältigende Mehrheit (70 Prozent) setzte derartige Workloads in der Produktion ein, wobei Datenbanken ganz oben auf der Liste standen. Als Schlüsselfaktoren führten Unternehmen erhebliche Vorteile hinsichtlich der Standardisierung, Konsistenz und Verwaltung an.

Ausführung auf Kubernetes

Kubernetes wurde 2015 als Open-Source-Software eingeführt und entwickelte sich schnell zur gängigsten Plattform für die Planung und Orchestrierung von Containern. Die Technologie avanciert zunehmend zur Grundlage für die meisten Anwendungen, unabhängig davon, wo sie bereitgestellt werden sollen. Kubernetes ist auf dem besten Weg, sich neben Linux und vSphere als bevorzugte Enterprise-Plattform zu etablieren.

Mit Kubernetes können Anwendungen bereitgestellt und gewartet werden. Die Skalierung erfolgt auf der Grundlage unterschiedlicher Metriken wie CPU und Arbeitsspeicher. Die Bausteine der Plattform werden als *Primitives* (Befehle) bezeichnet und ihre Rechen- und Speicherressourcen als Objekte definiert. Die wichtigsten API-Objekte in Kubernetes sind:

- Cluster
- Nodes
- Labels und Selectors
- Replikationssatz
- Deployment

Kubernetes ist eine robuste Lösung, mit der Benutzer containerbasierte Anwendungen einfach bereitstellen, skalieren und verwalten können. Dank ihrer Erweiterbarkeit und Portierbarkeit erfreut sich die Technologie im Cloud-Computing-Ökosystem größter Beliebtheit. Kubernetes bietet Benutzern zudem eine hohe Flexibilität hinsichtlich der Programmiersprache und des Frameworks und gibt ihnen die Möglichkeit, Fehler nachzuverfolgen und zu protokollieren.

KUBERNETES – EIN KURZER RÜCKBLICK

An dieser Stelle lohnt es sich, einen Blick auf die Entwicklung von Kubernetes zu werfen. Der Name Kubernetes stammt aus dem Griechischen und bedeutet *Steuermann* oder *Lotse*. Etymologisch ist er mit dem Wort Kybernetik (Regelungstechnik) verwandt. Das Projekt wurde 2014 von einer Gruppe von Google-Ingenieuren ins Leben gerufen und war ursprünglich unter dem Namen Project 7 bekannt.

(Hier noch ein interessanter Fakt für Science-Fiction-Fans: Der ursprüngliche Codename war eine Hommage an die *Star Trek*-Figur Seven of Nine, die einst zum mächtigen Borg-Kollektiv gehörte, und spielt darauf an, dass Kubernetes von einem älteren Google-System namens Borg beeinflusst wurde.)

NICHT VERGESSEN

Dank dieser Vorteile lassen sich nicht nur erhebliche Produktivitätssteigerungen und Kostensenkungen erzielen, auch Anwendungen können schneller in der Produktion eingesetzt werden. Kubernetes automatisiert zahlreiche, mit der Bereitstellung, Verwaltung und Skalierung verbundene Prozesse. Benutzer können sich die Fähigkeiten

von Kubernetes zunutze machen, um Host-Cluster zur Ausführung von Containern zu erstellen, und diese Cluster über Public, Private oder Hybrid Clouds hinweg zu verwalten. Dabei dürfen sie davon ausgehen, dass alles hervorragend funktioniert und dass sie sich keine Sorgen über Ausfallzeiten machen müssen.

Kein Wunder, dass sich Kubernetes so schnell durchgesetzt hat! Im nächsten Abschnitt werden einige der Vorteile der Plattform ausführlicher erläutert. Wie bei den meisten Technologien, die schnell an Beliebtheit gewinnen, kommt es jedoch auch mal vor, dass viele auf den Zug aufspringen, ohne wirklich zu verstehen, worauf sie sich eigentlich einlassen.

Im Fall von Kubernetes hat dies zu einigen Herausforderungen in der „Day 2"-Production geführt, vor allem im Hinblick auf Datenmanagement, Sicherheit und Observability. Ein Großteil der mit der Hochverfügbarkeit und Skalierbarkeit von Anwendungsdiensten verbundenen Probleme wird zwar beseitigt, doch diese Vorteile können nicht automatisch auf die Daten übertragen werden. Deshalb muss der Verwaltung von Anwendungsdaten bei Kubernetes besondere Priorität eingeräumt werden.

Die Vorteile erkennen

Entwicklungsteams wissen die durch Kubernetes gewonnene höhere Agility, Portierbarkeit und Zuverlässigkeit zu schätzen. Es überrascht also nicht, dass eine zunehmende Anzahl von Anwendungen auf der Plattform ausgeführt wird: Dabei handelt es sich nicht nur um Stateless-, sondern auch um Stateful-Anwendungen, einschließlich jener, die auf NoSQL-Datenbanken basieren, sowie um Anwendungen, die eine relationale Datenbank für ihre Backend-Prozesse verwenden.

Die Nutzung einer cloudbasierten Infrastruktur und Kubernetes bringt u. a. die folgenden Vorteile mit sich:

- Es ist leicht, auf alle Computer-, Speicher- und Netzwerkressourcen zuzugreifen, die für ein schnelles Wachstum erforderlich sind.
- Die Speicherung ist einfach und der Self-Service reibungslos. Mit Kubernetes lassen sich relationale und NoSQL-Datenbanken einfach und nahtlos integrieren.
- Änderungen an containerbasierten Anwendungen können problemlos bereitgestellt werden. Selbst bei komplexen Anwendungen können schnell Verbesserungen und Updates vorgenommen werden.

» Die Plattform kann Skalierungsanforderungen fast sofort erfüllen.

ABBILDUNG 1-1: Das Kubernetes-Ökosystem.

Abbildung 1-1 bietet einen Überblick über das Kubernetes-Ökosystem.

Mythen um Kubernetes

Häufig wird die Meinung vertreten, dass auf der Kubernetes-Plattform ausgeführte Anwendungen „stateless", also temporär sein sollten. Als Kubernetes noch in den Kinderschuhen steckte, traf dies auch weitgehend zu. Im Bereich der Technologie vollziehen sich Veränderungen jedoch in einem rasanten Tempo, und was einst als Tatsache galt, entspricht heute oft nicht mehr der Realität.

Die Unterstützung für Speicher- und Stateful-Anwendungen hat mittlerweile einen hohen Reifegrad erreicht. Kubernetes hat sich zu einer idealen Plattform für Stateful- und Stateless-Anwendungen entwickelt. Um Ihre Stateful-Kubernetes-Anwendung zu schützen, benötigen Sie jedoch eine Kopie der Daten an einem anderen, völlig unabhängigen Speicherort.

WARNUNG

Und genau das kann zum Problem werden. Es ist zweifellos eine gute Sache, dass Anwendungsentwickler von einer reibungslosen und dynamischen Self-Service-Speicherbereitstellung profitieren können. Allerdings ist in Kubernetes-Clustern, die nicht unbedingt für Stateful-Anwendungen ausgelegt sind, eine umfangreiche Speichernutzung keine Seltenheit.

Das ist deshalb problematisch, weil Ihre Daten aufgrund der umfassenden Nutzung relationaler und nicht relationaler Datenbanken auf Kubernetes ungeschützt sein können, wenn Sie nicht über geeignete

Datenverwaltungssysteme verfügen. Für das Unternehmen stellt dies ein erhebliches Risiko dar.

Viele sind sich dieses Risikos jedoch nicht bewusst, und die Gründe sind leicht nachzuvollziehen. Einer der Pluspunkte bei der Ausführung von Anwendungen auf Kubernetes ist die Hochverfügbarkeit bei Software- und Serverproblemen sowie bei regionalen Ausfällen. Mit einer Stateful-Anwendung ist es viel einfacher, Datenreplikationsdienste über Fehlerdomänen hinweg zu betreiben.

NICHT VERGESSEN

Hochverfügbarkeit und Backup sind jedoch zwei unterschiedliche Dinge. Hochverfügbarkeit bedeutet nicht, dass es an einem anderen Standort ein Backup Ihres Workloads gibt. Die Replikation verbessert zwar die Datenverfügbarkeit und schützt vor einem teilweisen Ausfall der Infrastruktur, aber nicht vor dem Verlust oder der Beschädigung von Daten, ganz gleich, ob vorsätzlich oder versehentlich. Wie Sie den erforderlichen Schutz erhalten, erfahren Sie hier in diesem Buch.

IN DIESEM KAPITEL

- **Warum native Backups wichtig sind**
- **Berücksichtigung neuer Bereitstellungsmuster**
- **Einordnung in die DevOps-Welt**
- **Weniger Stress für Bediener**
- **Die Lücken bei der Datensicherung schließen**
- **Nutzung mehrerer Datenservices**

Kapitel 2
Einrichtung einer nativen Datensicherung für Kubernetes

Ihr Unternehmen ist auf die Verfügbarkeit seiner Anwendungen angewiesen. Mit Kubernetes kann diese sehr gut sichergestellt werden. Das hört sich schon einmal gut an, doch was wäre, wenn Sie alle Daten in Ihrem Unternehmen verlieren würden?

In diesem Kapitel erfahren Sie, warum Sie Kubernetes-native Backup-Lösungen brauchen. Es beleuchtet die zahlreichen Faktoren, die eine derartige Lösung unverzichtbar machen – von den unterschiedlichen Bereitstellungsmustern bis hin zu den als DevOps bezeichneten Praktiken. Lesen Sie weiter, um zu erfahren, wie cloud-native Backups Ihren Bedienern viel Stress ersparen und ihnen mehr Freiraum für Innovationen verschaffen. Erfahren Sie außerdem, wie Sie Lücken bei der Datensicherung schließen und dafür sorgen können, dass auch dann alles reibungslos läuft, wenn Ihre Anwendungen mit mehreren unterschiedliche Datenservices verbunden sind.

Bedarf erkennen

Natürlich braucht Ihnen niemand zu sagen, wie wichtig Backups sind. Ganz gleich, mit welcher Art von Technologie Sie arbeiten, haben Sie wahrscheinlich schon einmal einen Albtraum gehabt, in dem es um katastrophale Datenverluste ging.

WARNUNG

Es gibt zahlreiche Ausfallszenarien, in denen dieser böse Traum zur Realität werden könnte, zum Beispiel durch versehentliche Löschung, mangelndes Verständnis der verwendeten Plattform, Ransomware und andere Angriffe. Ihre Daten sind einer Vielzahl von Gefahren ausgesetzt. Wie würde Ihr Unternehmen ohne diese Daten überleben?

NICHT VERGESSEN

Ich möchte Ihnen nicht noch mehr schlaflose Nächte bereiten, doch eines lässt sich nicht verleugnen: Kubernetes ist eine völlig neue Welt, in der das bereits bestehende Risiko für Datenverluste in der Cloud noch erhöht werden können. Kubernetes ist komplex, die wenigsten sind damit vertraut und die Verwaltungsaufgaben sind weniger zentralisiert. Daher kann es mit höherer Wahrscheinlichkeit zu Zwischenfällen kommen.

WARNUNG

Für virtualisierungsbasierte Infrastrukturen gibt es bereits hervorragende Tools auf dem Markt. Für nicht virtualisierte Umgebungen werden jedoch ganz neue Tools benötigt. Man mag versucht sein, jedem Anwendungsteam einen Teil der Verantwortung zu übertragen, doch durch eine geteilte Backup-Verantwortung erhöht sich nur das Risiko, sondern auch die Wiederherstellungsdauer. Und Zeit ist ja bekanntlich Geld. In diesem Fall belaufen sich die durchschnittlichen Kosten pro Ausfallstunde bei kritischen Anwendungen auf etwa eine halbe Million Euro.

Mit einer cloudnativen Backup-Lösung fahren Sie also deutlich besser. Um Kubernetes-basierte Anwendungen zu sichern und zu schützen, benötigen Sie eine Kubernetes-native Backup-Lösung.

Unterschiedliche Bereitstellungsmuster

Dass Kubernetes ein Game-Changer ist, muss wohl nicht extra erwähnt werden. Das System ist revolutionär und populär – eine ganz neue Art der Architektur, bei der die Abstraktionsebene verlagert wurde, um die Flexibilität zu erhöhen und die Ausführung von Workloads zu verbessern. Die Kubernetes-Plattform unterscheidet sich grundlegend von fast allen früheren Recheninfrastrukturen.

Eines hat sich jedoch nicht geändert: die Anforderungen rund um die Datensicherung. Administratoren sollten für jede Plattform einen Backup-Plan haben, nicht nur für Kubernetes.

Eine zeitlose Regel, mit der jedes Ausfallszenario effektiv angegangen werden kann, ist die 3-2-1-Backup-Regel. Dieser Ansatz hilft bei der Beantwortung zweier wichtiger Fragen: Wie viele Backup-Dateien sollte ich haben und wo soll ich sie speichern? Die 3-2-1-Backup-Regel liefert die Antworten:

- **3:** Fertigen Sie mindestens drei Kopien Ihrer Daten an.
- **2:** Speichern Sie die Kopien auf zwei unterschiedlichen Medien.
- **1:** Bewahren Sie ein Backup an einem externen Ort auf.

NICHT VERGESSEN

Betrachten wir zunächst die Tatsache, dass Sie containerbasierte Anwendungen weder Servern noch virtuellen Maschinen (VMs) zuordnen müssen. Im Gegensatz zu VMs benötigt ein Container lediglich ein Betriebssystem, unterstützende Programme und Bibliotheken sowie Systemressourcen, um ein bestimmtes Programm auszuführen.

Daher können Sie mit Containern zwei- oder dreimal so viele Anwendungen auf einem einzigen Server unterbringen wie mit einer VM. Mithilfe von Containern können Sie auch eine portable, konsistente Betriebsumgebung für die Entwicklung, das Testen und die Bereitstellung erstellen. Kubernetes verteilt Anwendungskomponenten mit seiner eigenen Platzierungsrichtlinie auf alle Server, um die Leistung und Fehlertoleranz zu erhöhen.

Wenn man in solchen Situationen ein herkömmliches Datenmanagementsystem verwendet, ist der Misserfolg vorprogrammiert. Sie können vielleicht ein Backup erstellen, aber wenn Sie Tools verwenden, die nicht für die Cloud entwickelt wurden, wird es spätestens bei der Wiederherstellung schwierig.

Hinzu kommt, dass cloudbasierte Anwendungen von der Dynamik ihrer Umgebung profitieren. Zur Verbesserung des Lastausgleichs kann bei Containern im laufenden Betrieb eine Umplanung oder Skalierung auf unterschiedlichen Knoten vorgenommen werden. Es gibt ständig neue Bereitstellungen, bei denen Komponenten hinzugefügt und entfernt werden.

NICHT VERGESSEN

Mit anderen Worten: Die Anwendung verändert sich kontinuierlich. Deshalb brauchen Sie eine Backup-Lösung, die cloudbasierte Architekturmuster versteht, ohne feste IP-Adressen auskommt und mit Veränderungen ebenso gut zurechtkommt wie Ihre cloudbasierte Kubernetes-Anwendung.

Herkömmliche Backup-Lösungen, die in einer Umgebung mit Servern und VMs zuverlässig funktionieren, sind für eine Kubernetes-Umgebung wahrscheinlich weniger tauglich. Zur Erfüllung von Anforderungen wie dynamische Anwendungserkennung, sofortige Backups, Plattformintegration, Wiederherstellung und die Fähigkeit, den gesamten Anwendungskontext zu erfassen, brauchen Sie Kubernetes-native Backups.


TECHNISCHES

Um sich diese Entwicklung etwas genauer vor Augen zu führen, sollten Sie bedenken, dass physische Systeme einen agentenbasierten Ansatz zum Schutz der Daten und des Betriebssystems erforderten. Als die Virtualisierung Einzug hielt, verlegten viele der Backup-Anbieter ihre Agenten einfach in virtuelle Umgebungen. Dadurch wurden die VMs bei der Durchführung von Backups zusätzlich belastet, weil sie in ihrer neuen Umgebung einfach wie physische Maschinen behandelt wurden.

Es zeigte sich jedoch schnell, dass sich diese virtuellen Workloads am besten auf Virtualisierungsebene über APIs schützen ließen. So war es möglich, anwendungskonsistente Backups schnell und effizient zu erstellen, ohne die Leistung der virtuellen Maschine zu beeinträchtigen.

Dasselbe Szenario spielt sich in der Kubernetes-Welt noch einmal ab. Theoretisch könnten Sie Ihren virtualisierten oder physischen Backup-Prozess verwenden, um einen Teil Ihrer Kubernetes-Umgebung und -Daten zu schützen. Dabei kann jedoch nicht alles geschützt werden, was bei der Wiederherstellung zu Problemen führt.

Ansatz der Linksverschiebung (Shift Left)

Viele stellen sich die DevOps-Philosophie wie ein Unendlichkeitszeichen vor – die liegende „8“, die eine unendliche Bewegungsschleife von links nach rechts, von rechts nach links usw. erzeugt. Auch der Vergleich mit einer Rennstrecke ist treffend, da es bei DevOps um extrem schnelle Anwendungsentwicklungszyklen geht.


NICHT VERGESSEN

Wenn man sich DevOps als Unendlichkeitszeichen vorstellen will, ist die Entwicklung üblicherweise links angeordnet und den Betrieb rechts. Wenn die DevOps-Philosophie auf die Kubernetes-Welt übertragen wird, werden die Bedürfnisse und Anforderungen von Entwicklern und Betriebsabläufen auf eine Weise kombiniert, wie es bisher nicht der Fall war. Auf der linken Seite der Schleife passiert mehr, daher die Linksverschiebung (Shift Left).

Der Fokus von Kubernetes ist ganz klar auf Entwickler und ihre Anwendungen gerichtet – und die ständige Hetzjagd durch die Entwicklungszyklen. Aufgrund des Plattformdesigns müssen Backup-Lösungen in erster Linie anwendungs- und nicht infrastrukturorientiert sein.

Entwickler, die in dieser Umgebung arbeiten, legen sowohl Anwendungskomponenten in Form von Code als auch Infrastrukturanforderungen wie Speicher und Lastverteiler fest. Die Anforderungen für die Datensicherung müssen integriert werden und im Einklang mit dem Code stehen. So können Schutzpläne erstellt werden, anstatt die Verantwortung wie bisher an die Infrastruktur- oder Backup-Administratoren zu übertragen.


TIPP

Sie sollten in der Lage sein, Datenmanagementaufgaben in den täglichen Entwicklungsprozess einzubeziehen, sodass die Verantwortung nicht mehr nur beim Operations-Team liegt. Dies lässt sich am besten durch einen API-Ansatz erreichen. Die Backup-Plattform muss API priorisieren und über eine cloudbasierte API verfügen.

Die Rede ist von einer Kubernetes-nativen API, nicht von den älteren REST- oder SOAP-APIs. Die Authentifizierung und Autorisierung erfolgen nahtlos und Sie können eine einfachere Anwendungs- und Workflow-Integration erreichen. Entwickler und Bediener können vertraute Tools wie kubectl verwenden.

Bedienerprobleme überwinden

Bei der Virtualisierung wird physische Hardware abstrahiert und diese Abstraktion dann zur Erstellung mehrerer virtueller Maschinen genutzt, in denen monolithische Workloads untergebracht werden sollen. Bei cloudbasierten Workloads hingegen liegt der Fokus vollständig auf der Anwendung und nicht auf der VM. Das Management wird dadurch vereinfacht, dass Anwendungen als operative Einheiten im Mittelpunkt stehen, während die Infrastruktur oder die Datenspeicher abstrahiert werden.


TIPP

Verwenden Sie unbedingt ein Backup-Tool, das Ihnen die nötige Flexibilität gibt, um selbst über die angemessene Vorgehensweise zu entscheiden. Nicht jedes Unternehmen benötigt eine API für seine Backup-Workflows. Einige brauchen ein Dashboard, um durch die Kubernetes-Umgebung zu navigieren.

Eine tiefgreifende Kubernetes-Integration kann die Komplexität der zugrundeliegenden Plattform in den Hintergrund treten lassen. Sie können manuelle Aufgaben oder Integrationsarbeit vermeiden oder

zumindest reduzieren, wenn Sie sich auf die Benutzererfahrung konzentrieren und sich die Backup-Workflows für cloudbasierte Anwendungen noch einmal genauer ansehen.

Denken Sie daran, dass in der Vergangenheit für die Ausführung einer einzelnen Anwendung oft mehrere virtuelle Maschinen erforderlich waren. Heute bestehen containerbasierte Anwendungen in der Regel aus Hunderten von unterschiedlichen Kubernetes-Ressourcen, wozu auch Konfigurationen, Festplatten und Secrets gehören.

Und wir sprechen hier nur von einer Anwendung. Stellen Sie sich nun alle Anwendungen in einem Cluster vor. In diesem Fall muss Ihr Bediener Millionen von Komponenten verstehen und schützen – sofern die Anwendung nicht die operative Einheit für das Backup ist.


WARNUNG

Ein traditionelles Backup-System legt den Fokus in der Regel auf die Infrastruktur wie Festplatten und Volumes und ignoriert Kubernetes-Ressourcen. Damit sind Fehler bei Recovery-Playbooks mit fehlenden Beziehungen vorprogrammiert, was letztendlich zu einer sehr langen Wiederherstellungsdauer führt.

In dieser unerfreulichen Situation wäre dann ein manueller Prozess erforderlich, um herauszufinden, welche Backups für die Wiederherstellung benötigt werden. Und dann ein weiterer komplexer manueller Prozess, um die wiederhergestellten Volumes wieder mit den Kubernetes-Anwendungen zu verbinden. Das ist mit einer enormen betrieblichen Belastung verbunden, selbst wenn es zwischen Sicherung und Wiederherstellung keine Abweichungen bei den Kubernetes-Objekten gegeben hat (und das ist unwahrscheinlich).

Die Lücken schließen

Wenn es darum geht, Ihre Anwendungen selbst bei teilweisen Infrastrukturausfällen am Laufen zu halten, ist Kubernetes kaum zu überbieten. Die Fehlertoleranz ist ein starkes Verkaufsargument. Man darf sich jedoch nicht in falscher Sicherheit wiegen und vergessen, wie wichtig Sicherung und Wiederherstellung, Disaster Recovery, Anwendungsmobilität und Ransomware-Schutz sind.


NICHT VERGESSEN

Auch hier gilt: Hochverfügbarkeit und Replikation sind nicht dasselbe wie ein Backup. Das Risiko einer Beschädigung oder Löschung von Daten besteht weiterhin, sei es durch einen Zwischenfall oder durch eine böswillige Handlung. Sollte dies passieren, könnte sich das Problem auf alle Replikate ausbreiten und zu einem katastrophalen Datenverlust führen.

Kubernetes wird oft in Public Clouds ausgeführt. Wird die Speicherung dadurch nicht ausfallsicher? Nein, das ist ein Mythos. Die zuverlässigsten Cloud-Storage-Lösungen versprechen zwar eine sehr hohe Verfügbarkeit, doch für den Schutz Ihrer Daten sind Sie selbst verantwortlich.

Wie sieht es mit Anbietern von lokalen Speicherlösungen aus – die können doch Volume-Snapshots bereitstellen, oder? Ja, aber diese Snapshots sind trotzdem oft anfällig für Hardwareausfälle. Und wird ein Volume gelöscht, werden damit zusammenhängende Snapshots meist automatisch mitgelöscht.

Hier kommt die 3-2-1-Backup-Regel ins Spiel, die bereits im Abschnitt „Unterschiedliche Bereitstellungsmuster" erwähnt wurde. Diese Regel schützt Sie vor den meisten, wenn nicht allen Ausfallszenarien, besonders wenn Sie damit beginnen, Ihre externen Kopien durch Immutability (Unveränderlichkeit) zu schützen.


TECHNISCHES

Ohne erweiterte Kubernetes-Sicherheitsberechtigungen werden Aktionen wie das Stilllegen des Dateisystems normalerweise nicht zugelassen. Wenn Sie jedoch über ein Kubernetes-natives Backup-System mit klar definierten Berechtigungen und rollenbasierter Zugriffskontrolle verfügen, können Sie Quiesce-Hooks für Datenbanken und Kubernetes-Workloads nutzen, um dasselbe Ergebnis ohne Sicherheitskompromisse zu erzielen.

Wichtig ist, dass das Entwickler- und Operations-Team zusammenarbeitet – oder zumindest enger als zuvor. Sie alle haben ein gemeinsames Ziel, nämlich das Richtige für die Daten zu tun.

Die Disaster Recovery ist ein weiteres Element, das in Betracht gezogen werden sollte. Für Ausfallszenarien, bei denen eine Wiederherstellung innerhalb desselben Clusters möglich ist, sind Backups unverzichtbar. Bei der Disaster Recovery werden Ihre Workloads jedoch an einem völlig anderen Ort aufgerufen.

Integration der Ökosysteme


TECHNISCHES

Der Begriff *Polyglot Persistence* bezieht sich auf die Verwendung mehrerer Datenspeichertechnologien für unterschiedliche Speicheranforderungen in einer Anwendung oder in kleineren Komponenten einer Anwendung. Dabei geht es nicht nur um relationale und nicht relationale Datenbanken, sondern auch um Speicherbereiche wie Batch/Daten-Streaming und Message Queues.

Unterschiedliche Speicheranforderungen kann es in einem Unternehmen mit mehreren Anwendungen oder in einzelnen Komponenten

einer Anwendung geben, die Daten unterschiedlich speichern müssen. Polyglot Persistence nimmt mit der zunehmenden Verbreitung von Kubernetes zu.

Die gute Nachricht ist, dass es trotz dieser Komplexität möglich ist, umfassendere Backups für diese Workloads zu erstellen, wenn Kubernetes zur automatischen Workload-Erkennung integriert wird. Das bedeutet, dass die Backup-Lösung die Anforderungen der Anwendung berücksichtigen und den E/A-Befehl auswählen kann (z. B. Volume-Snapshots, anwendungskonsistente Backups und logische Speicherabbilder).

Die Zeiten, in denen nur ein einziger Datenservice genutzt wurde, sind vorbei. Mit Kubernetes-Metadaten kann die Backup-Lösung automatisch Beziehungen zwischen mehreren unabhängigen Datenservices erkennen.

Da Kubernetes die Anwendungstopologie vollständig erfasst, kann die Kubernetes-native Backup-Lösung eine konsistente Kopie des gesamten Anwendungsstacks erfassen, sowohl innerhalb einzelner Services als auch serviceübergreifend. Dadurch können Daten von Replikaten identifiziert und erfasst werden, um die Anwendungsauswirkung zu reduzieren. Die Leistung und Effizienz werden verbessert, während die Parallelverarbeitungsfähigkeit von Kubernetes die Wiederherstellungsleistung optimiert.

Hinsichtlich der Komplexität muss erwähnt werden, dass immer mehr Unternehmen eine große Anzahl von Kubernetes-Clustern in unterschiedlichen Umgebungen betreiben. Die Backup-Plattform muss daher mit dem Rest des cloudbasierten Ökosystems und seiner Infrastruktur zusammenarbeiten.

Letztendlich profitieren Sie von einer besseren Benutzererfahrung, helfen Operations-Teams dabei, effizienter zu arbeiten, und senken gleichzeitig die Kosten. Ein Vorteil dieser verbesserten Benutzererfahrung ist die Tatsache, dass Entwickler und Bediener weiterhin die ihnen vertrauten cloudbasierten Tools verwenden können. Sie können zum Beispiel Prometheus zur Überwachung und Benachrichtigung und Kubernetes-APIs für die rollenbasierte Zugriffssteuerung (RBAC) integrieren, um die für die Ursachenanalyse erforderliche Protokollierung und Prüfung durchzuführen.

IN DIESEM KAPITEL

- Betrachtung der Anwendung als Ganzes
- Untersuchung und Skalierung der Architektur
- Gewährleistung der Wiederherstellbarkeit
- Vereinfachung von Abläufen
- Sicherheit in mandantenfähigen Umgebungen
- Erfolgreiche Wiederherstellung und Portierbarkeit

Kapitel 3
Best Practices für Kubernetes-Backups

Wie in den Kapiteln 1 und 2 beschrieben, ist eine Kubernetes-Umgebung eine Welt für sich. Ihr bisheriger Backup-Ansatz reicht nicht mehr aus oder funktioniert womöglich überhaupt nicht.

In diesem Kapitel werden einige Best Practices zur Sicherung Ihrer Kubernetes-Anwendungen vorgestellt. Es wird erklärt, welche Architekturunterschiede einen anwendungsorientierten Fokus erfordern, wie die Wiederherstellbarkeit gewährleistet werden kann und warum ein natives Backup-System das Skalieren von Anwendungen erleichtert. Das Kapitel zeigt, wie wichtig Sicherheitsaspekte sind, besonders angesichts der mit der Mandantenfähigkeit einhergehenden Schwierigkeiten. Außerdem werden die Herausforderungen von sich ständig weiterentwickelnden Objekten und APIs sowie die Bedeutung der Portierbarkeit thematisiert.

Erfassung der Anwendung

Grundsätzlich steht die Anwendung immer im Mittelpunkt. Die Infrastruktur ist zwar auch wichtig, aber letztendlich ist sie dazu da, die Anwendung verfügbar und flexibel zu machen. Das war schon immer

der Fall. Mit Kubernetes erhält diese Tatsache jedoch eine neue Dimension, da der Fokus hier auf Entwicklern und den von ihnen erstellten Anwendungen liegt, sowie auf der Geschwindigkeit, mit der sie diese Anwendungen entwickeln und aktualisieren.

Für eine Plattform, die in erster Linie auf Entwickler und Anwendungen ausgerichtet ist, wird eine Backup-Lösung benötigt, die ebenfalls anwendungsorientiert ist. Wie in den Kapiteln 1 und 2 erläutert, muss Ihr Backup Kubernetes-nativ sein, um containerbasierte Anwendungen in der Kubernetes-Welt richtig unterstützen zu können.


TIPP

Anwendungsorientiert bedeutet, Kubernetes-Konstrukte zu verstehen, anstatt sich auf die Infrastruktur zu konzentrieren. Dazu ist eine vollständige Erfassung der Anwendung und der Schutz aller Komponenten erforderlich – und es dürfen keine Ressourcen, Filter oder Labels übersehen werden.

Erschließung der Architektur

In diesem Buch geht es speziell um Kubernetes-Backups, die eine wesentliche Rolle für den Erfolg auf dieser Plattform spielen. Ein Deep Dive in die Kubernetes-Architektur ist an dieser Stelle nicht erforderlich. Es ist jedoch hilfreich, sich ein wenig mit der Architektur zu befassen, um ein Gefühl dafür zu bekommen, wie man eine gute Backup-Strategie am besten umsetzt.

Verbindung der Komponenten

Abbildung 3-1 zeigt einige der Hauptkomponenten einer Kubernetes-Anwendung. Dazu gehören unter anderem Pods, Services, Zertifikate, Secrets und persistente Volumes.

Eine Anwendung, die in einer Produktionsumgebung eingesetzt wird, besteht aus Hunderten solcher Komponenten. Somit stellt sich die Frage, wie man Daten und alle anderen internen Komponenten am besten schützt und wiederherstellt. Und wie man das in großem Maßstab realisieren kann.


TIPP

Die gute Nachricht ist: Sie müssen nicht alles im Alleingang machen. Mit der richtigen Backup-Plattform können Sie über den API-Server automatisch eine Schnittstelle zur Kubernetes-Steuerungsebene herstellen. Dank dieser Integration erkennt die Backup-Lösung die auf dem Cluster ausgeführten Kubernetes-Anwendungen und integriert sie dann in die zugrundeliegende Rechen-, Netzwerk- und Speicherinfrastruktur.

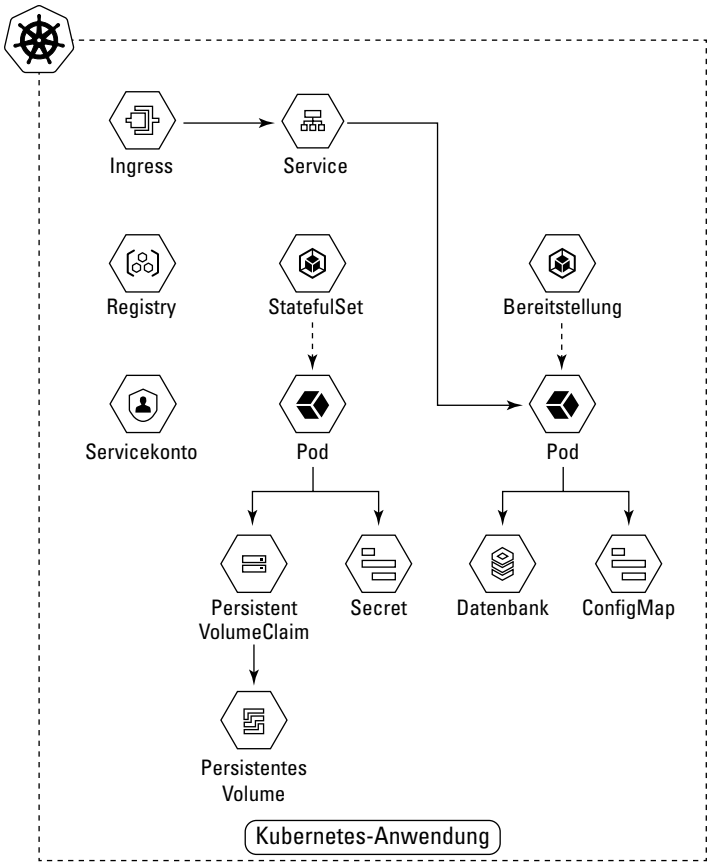

ABBILDUNG 3-1: Eine einfache Kubernetes-Anwendung ist eine Gruppe miteinander verbundener Komponenten, die alle mit dem Backup- und Wiederherstellungssystem verbunden sein müssen.

Zunächst einmal müssen Sie die Beziehung zwischen Speicher und Anwendungen verstehen. Dann gilt es herauszufinden, wie sich die auf persistenten Volumes gespeicherten Anwendungsdaten und alle zugehörigen Anwendungsressourcen am besten erfassen lassen. Diese Aufgabe muss effizient und konsistent durchgeführt werden.

Als Nächstes wird der Speicherort für die Backup-Daten festgelegt. Dieser kann sich im Speichersystem befinden, um eine schnelle Wiederherstellung zu ermöglichen. Wenn Sie für die Ausführung einen der großen Cloud-Provider nutzen, empfiehlt sich die Verwendung dauerhafter Snapshots. In den meisten Fällen ist es jedoch besser, Backup-Daten in einem Objektspeichersystem zu speichern, das sich in einer anderen Fehlerdomäne befindet und eine Georeplikation für die Disaster

Recovery ermöglicht. „Vorsicht ist besser als Nachsicht“ heißt hier die Devise.

TIPP

Die in Kapitel 2 besprochene 3-2-1-Backup-Regel ist eine gute Richtlinie, die Sie bei der Planung Ihrer Datensicherung befolgen sollten. Sie brauchen mindestens drei Kopien Ihrer Daten, die auf zwei unterschiedlichen Medien gespeichert werden müssen. Eine dieser Kopien sollte sich an einem externen Ort befinden. Wenn Sie diesen grundlegenden Ansatz nicht befolgen, sind Ihre Daten einem deutlich höheren Risiko ausgesetzt.

Bei der Speicherintegration in Kubernetes müssen Sie auch noch Folgendes beachten:

- Speicher werden als persistente Volumes dargestellt, die den Containern zur Verfügung gestellt werden. Der Schutz dieser wichtigen Geschäftsdaten ist von entscheidender Bedeutung.
- Wo wollen Sie diese Daten speichern? In einem lokalen Blockspeicher? Wenn Sie Kubernetes lokal oder extern ausführen, können Sie eine Objektspeicherplattform wie Amazon S3 oder Microsoft Azure Blob Storage in Betracht ziehen. Bei der Auswahl eines sekundären Speichers für Backup-Daten müssen Flexibilität, Auswahlmöglichkeiten und Benutzerfreundlichkeit berücksichtigt werden.

Ihre grundlegende To-do-Liste in diesem Bereich umfasst also:

- das Verständnis der Beziehung zwischen Anwendungen und deren Speicherung
- die Entscheidung, wie Anwendungsdaten auf persistenten Volumes gespeichert werden sollen
- die Entscheidung, wo Sie das Backup aufbewahren sollten, um die Einhaltung der 3-2-1-Regel zu gewährleisten

NICHT VERGESSEN

Unabhängig davon, welche Plattform Sie zum Schutz von Kubernetes-Anwendungen verwenden, muss diese automatisch alle im Cluster ausgeführten Komponenten erkennen und die jeweilige Anwendung als Kernelement behandeln. Die Anwendung sollte unbedingt den Status aller Speicher-Volumes und Datenbanken sowie die konfigurierbaren Daten in Kubernetes-Objekten, wie ConfigMaps und Secrets, berücksichtigen.

Hochskalierung

Bei Anwendungen, die auf Kubernetes ausgeführt werden, spielt sich viel hinter den Kulissen ab. Dank Microservices und dem Kubernetes-Support für jeden Aspekt von der Konfiguration bis zur Handhabung von

Secrets werden Anwendungen auf Hunderte Einzelkomponenten mit eigenen Lebenszyklen heruntergebrochen. Diese Komplexität ist meist nur für Kubernetes sichtbar.

Es bedarf einer cloudnativen Backup-Lösung, um die Millionen von Komponenten in großen Clustern zu bewältigen und die Beziehungen zwischen Anwendungen, den von ihnen verwendeten Daten und dem zugehörigen Kubernetes-Status zu verstehen. Nur eine cloudbasierte Lösung kann das alles in großem Maßstab bewältigen.


NICHT VERGESSEN

Kubernetes und cloudbasierte Anwendungen sind für eine einfache, bedarfsgerechte Skalierung nach oben und unten ausgelegt. Backup-Lösungen müssen hierbei mithalten können. Ihre Backup-Lösung sollte die folgenden Anforderungen erfüllen:

- Anwendung derselben cloudbasierten Architekturmuster, damit entsprechend den Anwendungs- und Clusteränderungen skaliert werden kann
- Skalierung nach unten bei Inaktivität
- automatische Ausführung ohne manuelle Eingriffe


NICHT VERGESSEN

Mit einer Backup-Plattform, die mit dem Cluster mitwächst, erzielen Sie die beste Leistung. Gleichzeitig sparen Sie aber auch Geld, da der Ressourcenverbrauch an den aktuellen, sich ständig ändernden Bedarf und nicht an Spitzenlasten gekoppelt ist. Da das Backup-System linear mit der Anwendung und dem Clusterwachstum skaliert wird, läuft es außerdem viel reibungsloser. Es gibt keine sprunghaften Veränderungen wie bei einem Appliance-basierten Modell.

Die Herausforderungen bei der Skalierung werden mit der zunehmenden Nutzung von Kubernetes-Multi-Clustern nur noch komplexer. Es kann Tausende von Namespaces pro Cluster und Hunderte Kubernetes-Ressourcen pro Namespace geben.

In unterschiedlichen Umgebungen wie Entwicklungs-, Staging- und Produktionsumgebungen sind mehrere Cluster zu finden. Aber es gibt auch Aufteilungen über die Grenzen von Anwendungen, Sicherheitsmechanismen und Teams hinweg. Cluster können außerdem in mehreren Verfügbarkeitszonen, Regionen, Clouds und lokalen Rechenzentren bereitgestellt werden.

Wenn sie alle manuell verwaltet werden müssten, wäre das ein wahrer Albtraum. In der Realität wäre es auch gar nicht möglich, es sei denn, man verfügt über eine cloudnative Backup-Plattform, die Multi-Cluster-Vorgänge unterstützt und umfassende Einblicke bietet.

Hier sind die wichtigsten Schritte, die Sie im Zusammenhang mit der Skalierung befolgen sollten:

- Stellen Sie sicher, dass Ihre Backup-Lösung den zu schützenden Anwendungen entsprechend nach oben und unten skaliert werden kann.
- Ermitteln Sie, wie Ihre Lösung auf die Herausforderungen des Multi-Clustering von Kubernetes reagieren wird.

Planung für die Wiederherstellbarkeit

Wiederherstellbarkeit ist der Heilige Gral, um den sich alles in diesem Buch dreht. Sie erfordert eine umfangreiche Planung und Ausführung. Hierbei geht es aber um weit mehr als nur um die Neuerstellung von Kubernetes-Objekten und Speicher-Volumes.


TIPP

Ihre containerbasierte Anwendung mit all ihren Kubernetes-Komponenten ist komplex. Der erste Schritt ist daher die Erstellung eines Ausführungsplans, der Folgendes regelt:

- Überprüfung von Clusterabhängigkeiten
- Erstellung neuer Kubernetes-Ansichten der wiederherzustellenden Daten
- Festlegung der Recheninfrastruktur und des Kubernetes-Clusters, in dem die Wiederherstellung initiiert werden soll, z. B. eine Wiederherstellung über mehrere Verfügbarkeitszonen hinweg

Wenn dieser Plan fertig ist, müssen Sie die Backup-Datenquellen identifizieren, z. B. Objektspeicher, Snapshots und Backups. Der Zielspeicher muss vorbereitet werden, was unter anderem die Neuzuordnung der Speicherklasse und Änderungen der Speicherplattform erforderlich macht.


TIPP

Ermitteln Sie, ob der Plan geändert werden muss. Berücksichtigen Sie dabei Dinge wie die Erneuerung von TLS-Zertifizierungen, DNS-Änderungen und die Bearbeitung veralteter Secrets. Anschließend müssen die Kubernetes-Anwendungskomponenten aktualisiert werden, damit sie die neuen Speicherressourcen berücksichtigen, die im Rahmen der Wiederherstellung erstellt werden.

Wenn all diese Planungen abgeschlossen sind, muss die Backup-Plattform den Plan in die entsprechenden Kubernetes-API-Aufrufe übersetzen, damit die benötigten Ressourcen erstellt werden können (zum Beispiel Aufrufe zur Erstellung eines Lastverteilers oder zur Neuerstellung eines Secrets). Alle zu einer cloudbasierten Anwendung gehörenden Ressourcen und Microservices müssen mit der richtigen Konfiguration neu bereitgestellt werden. Siehe Abbildung 3-2 zur Veranschaulichung dieses Prozesses.

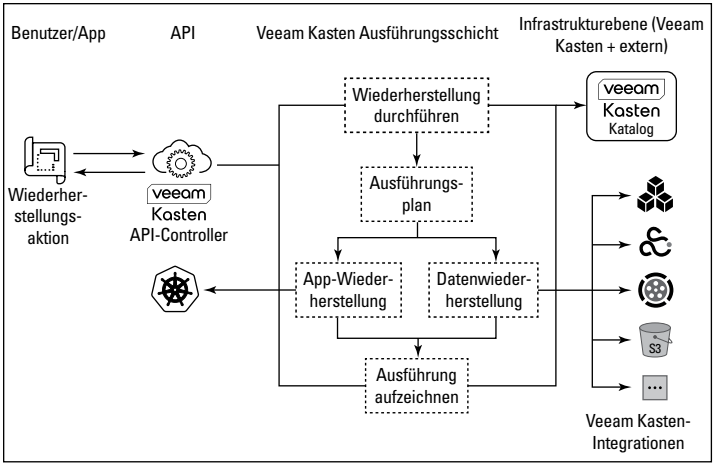

ABBILDUNG 3-2: Ein Blick auf den Wiederherstellungsprozess, einschließlich der API-Aufrufe, die alle erforderlichen Ressourcen erstellen.

Das Entscheidende ist, dass alle Anwendungskomponenten an jedem von Ihnen gewünschten Ort wiederhergestellt werden können. Außerdem sollten Sie über die erforderliche Granularität verfügen, um Anwendungen bei Bedarf nur teilweise wiederherzustellen – zum Beispiel das Datenvolumen. Stellen Sie sicher, dass Ihre Backup-Lösung Ihnen die Möglichkeit bietet, die entsprechende Point-of-Time-Kopie der Anwendung auszuwählen. Ist es zu viel verlangt, dass das alles so einfach wie möglich vonstattengehen sollte? Ganz und gar nicht.

Kurz gesagt, erfordert die Disaster Recovery eine sorgfältige Planung sowie das richtige Tool zur Umsetzung des Plans:

- Erstellen Sie Ihren Ausführungsplan mit Schwerpunkt auf Clusterabhängigkeiten und Ansichten der Daten sowie dem Ort, an dem die Wiederherstellung durchgeführt werden soll.
- Identifizieren Sie Datenquellen und den Zielspeicher.
- Entscheiden Sie, ob eine Transformation erforderlich ist und wie die Komponenten aktualisiert werden sollen.
- Wandeln Sie alles in Kubernetes-API-Aufrufe um.

Fokus auf den Betrieb

Führen Sie sich einige der Dinge vor Augen, die Kubernetes so populär machen: Entwickler können Anwendungen schnell und einfach bereitstellen und aktualisieren – und das alles in großem Maßstab. Das

Letzte, was Sie wollen, ist eine Backup-Plattform, die diese Effizienz beeinträchtigt. Im besten Fall wäre das frustrierend, im schlimmsten Fall könnte es Entwickler dazu veranlassen, Best-Practice-Prozesse zu umgehen.

TIPP

Natürlich ist es keine einfache Aufgabe, dafür zu sorgen, dass sich alle an Best Practices halten, besonders wenn neue Tools, Services und Funktionen in eine dynamische Infrastruktur eingeführt werden. Ihre Kubernetes-native Backup-Plattform muss:

- in großem Maßstab einsetzbar sein;
- dem Operations-Team die nötigen Workflow-Funktionen zur Verfügung stellen;
- die vielfältigen Anforderungen in Bezug auf Compliance und Monitoring erfüllen;
- für Entwickler unproblematisch sein.

Aus Entwicklersicht ist es wichtig, dass keine Änderungen in Bezug auf Code, die Erstellung von Paketen, die Toolkette und die Bereitstellung erforderlich sind. Bediener sollten Entwicklern auch Self-Service-Funktionen zur Verfügung stellen.

NICHT VERGESSEN

Entwickler sollten zum Beispiel in der Lage sein, ihre eigenen Anwendungen wiederherzustellen und Backup-Prozesse für ihre Datenservices anzupassen und zu erweitern. Sie sollten die Kontrolle über die serviceübergreifende Koordination und das Quiescing sowie über die Verwendung ihrer eigenen oder von Datenbankanbietern bereitgestellten Tools haben. Außerdem müssen die Interaktionen der Entwickler mit der Backup-Plattform API-gesteuert sein.

Bediener wiederum sind froh, dass sie sich dank der Backup-Lösung nicht mehr mit den Hunderten zur Anwendung gehörenden Kubernetes-Komponenten befassen müssen. Sie wünschen sich Backup-Richtlinien, die vollständig automatisiert sind und mit denen sie sich auf das große Ganze der Anwendung konzentrieren können, anstatt auf einzelne Ressourcen und die Speicherinfrastruktur.

Was die minutiösen Details der Richtlinie und alle zu schützenden Anwendungskomponenten betrifft, sollte diese Arbeit erst bei der Umsetzung der Richtlinie erforderlich sein. Von diesem Zeitpunkt an sollten keine manuellen Aktualisierungen mehr nötig sein, um alle Komponenten zu erfassen, wenn sich die Anwendung ändert (was natürlich immer der Fall ist).

BACKUP-JOBS ODER INTELLIGENTE RICHTLINIEN

Warum ist es besser, intelligente Backup-Richtlinien zu erstellen, anstatt spezifische Backup-Jobs einzurichten? Betrachten Sie die Unterschiede, um sich ein eigenes Bild zu machen.

Backup-Jobs beschreiben jeden Schritt zum Software-Stack, einschließlich des Zeitpunkts, zu dem der Job ausgeführt werden soll. Ein Backup-Job weiß natürlich nichts über die Rechenzentrumsarchitektur und braucht Ihre Unterstützung, damit er weder die Workloads noch die Netzwerkkapazität beeinträchtigt.

Richtlinien hingegen sind eher ergebnisorientiert. Anstatt Zeitpläne oder notwendige Änderungen vorzuschreiben, legen Sie ein Recovery Point Objective (RPO) oder ein gewünschtes Ergebnis fest. Wenn eine Richtlinie einmal erstellt wurde, ist sie immer aktiv, immer auf der Suche nach Problemen und Anforderungen und stets darauf ausgerichtet, das gewünschte Ergebnis zu erzielen.

Sie können deutlich sehen, welche Option stärker automatisiert ist und welche mehr menschliche Eingriffe erfordert. Abgesehen davon, dass diese Eingriffe Zeit kosten, führen sie leider auch oft zu Fehlern, wenn es viele Dinge zu berücksichtigen oder zu konfigurieren gibt.

TIPP

Hier ist eine weitere Best-Practice-Empfehlung, die dabei hilft, die Arbeit so einfach wie möglich zu gestalten. Erstellen Sie umfassende, labelbasierte Backup-Richtlinien, die neue Anwendungen automatisch erkennen können, sobald sie verfügbar sind.

Sie sollten zum Beispiel eine Richtlinie für alle Anwendungen haben, die MongoDB verwenden, oder eine Richtlinie für Anwendungen, die über das Helm-Tool bereitgestellt werden. Wenn Sie alles richtig machen, muss das Operations-Team keine manuellen Prozesse zur Änderungskontrolle einrichten. Sie müssen sich auch darauf verlassen können, dass Anwendungen bei ihrer Erstellung oder Entfernung stets die Service Level Agreements für Backups einhalten.

Im Klartext heißt das: Entscheiden Sie sich für eine Backup-Lösung, die allen das bietet, was sie sich wünschen und was sie brauchen. Eine Plattform, die den Anforderungen von Container-Operations-Teams und Entwicklern gleichermaßen gerecht wird, ist kein Ding der Unmöglichkeit. Geben Sie sich also nicht mit weniger zufrieden.

In Bezug auf den Betrieb sind die folgenden Punkte besonders wichtig:

- Stellen Sie sicher, dass Ihre Backup-Lösung tatsächlich eine Lösung ist und sich nicht als Problem entpuppt. Sie muss zur Effizienzsteigerung beitragen, Arbeitsabläufe optimieren und Anforderungen erfüllen.
- Finden Sie heraus, wo intelligente Richtlinien Erleichterung schaffen können und die Erstellung spezifischer Backup-Jobs überflüssig machen.
- Erstellen Sie Richtlinien, die selbstständig neue, ihrem Fokus entsprechende Anwendungen entdecken.

Gewährleistung der Sicherheit

Ganz gleich, ob die Bereitstellung in einer Public Cloud, in einer lokalen Infrastruktur oder in einer Hybridumgebung erfolgt – die Sicherheit muss immer im Vordergrund stehen. Die Schlagzeilen machen nahezu jeden Tag deutlich, dass Sicherheit heute wichtiger ist denn je. Die zunehmende Mandantenfähigkeit ist ein weiterer Grund, das Thema Sicherheit im Blick zu behalten.

Mehrere Schutzebenen

Die gute Nachricht ist, dass Kubernetes viele Sicherheitsfeatures bietet, darunter Netzwerkrichtlinien, die interne Anwendungskomponenten und die damit verbundenen Datenservices schützen. Kubernetes verhindert nicht nur den Zugriff auf diese Komponenten von außerhalb des Clusters, sondern stellt auch eine Barrikade für nicht vertrauenswürdige Anwendungen auf, die im selben Cluster ausgeführt werden.

NICHT VERGESSEN

Das ist zwar gut, doch es bedeutet auch, dass Sie keine Backup-Lösungen außerhalb Ihrer Kubernetes-Cluster ausführen können. Sie können Anwendungen weder erkennen noch auf sie zugreifen, um sie etwa in einen Backup-fähigen Zustand (Quiescing) zu versetzen, ohne dabei die Isolationsrichtlinien zu verletzen. Abhilfe schafft hier eine sorgfältig konzipierte, Kubernetes-native Lösung, die sich in die Steuerungsebene einbetten lässt.

Eine weitere Herausforderung könnte die Bereitstellung von Self-Service-Funktionen darstellen, weil Entwickler immer mehr Infrastrukturaufgaben übernehmen. Ihr Backup-System muss Kontrollen für das Identity and Access Management bereitstellen, und auf die rollenbasierte Zugriffskontrolle (RBAC) können Sie auf keinen Fall verzichten. Anhand dieser Kontrollen können Sie festlegen, welchen Benutzern und Gruppen bestimmte Zugriffsebenen, Einschränkungen oder Benutzerrechte in Bezug auf die Backup-Plattform zugewiesen werden sollen.

Der Zugriff sollte unter Anwendung der von Kubernetes definierten Rollen und Tools gewährt werden. Das ist besser, als weitere Rollenmanagementsysteme einzuführen, die Ihre Teams erst erlernen müssen.

Für das Operations-Team empfiehlt sich ein Least-Privilege-Ansatz zur Ausführung allgemeiner Aufgaben, z. B. das Monitoring von Backups, die Verifizierung ihrer erfolgreichen und vollständigen Durchführung sowie die Durchführung angeforderter Wiederherstellungen. Sie können bei Bedarf spezifische Anwendungsfälle erstellen und Entwicklern die Berechtigung erteilen, schnelle Wiederherstellungen und Klone von Snapshots zu erstellen, während der Zugriff auf Backups in externen Sekundärspeichern nur bestimmten Personen vorbehalten ist.

Ihre cloudnative Backup-Plattform muss darüber hinaus umfassend in die Systeme für das Identity and Access Management, das Schlüsselmanagement oder das Zertifikatsmanagement in der Cloud integriert sein. Ein Datenmanagementsystem, das wirklich Kubernetes-nativ ist, lässt sich außerdem in die Authentifizierungslösung des Cloud-Providers integrieren, was zusätzliches Benutzer- oder Gruppenmanagement, neue Tools oder APIs für RBAC-Richtlinien überflüssig macht.

TIPP

Beachten Sie, dass Kubernetes die Datenverschlüsselung an das zugrundeliegende Speichersystem und die Backup-Plattform delegiert. Anwendungsdaten dürfen daher niemals im Klartext gespeichert oder übertragen werden. Aus diesem Grund muss die Backup-Plattform:

- das Zertifikatsmanagement von Kubernetes verstehen;
- mit in den Speicher integrierten Schlüsselmanagementsystemen zusammenarbeiten;
- kundenseitig verwaltete Verschlüsselungsschlüssel über die Kubernetes Secrets-Schnittstelle unterstützen.

Nehmen wir zum Beispiel die Objektspeicherung: Wenn Sie eine lokale Kubernetes-Anwendungsbereitstellung haben, die Backups an AWS S3 überträgt, werden Ihre Daten über eine externe Internetverbindung von hier nach dort übertragen. Die Backup-Plattform muss sicherstellen, dass die Daten über ein Protokoll wie TLS verschlüsselt werden.

Die Frage der Verschlüsselung erübrigt sich auch dann nicht, wenn die Daten an dem Ort angekommen sind, an dem sie letztendlich gespeichert werden. Ihr System muss dafür sorgen, dass sie auch dort verschlüsselt werden: Daten, die bei der Speicherung nicht verschlüsselt sind, bleiben ungeschützt. Es reicht nicht aus, RBAC und damit zusammenhängende Richtlinien anzuwenden. Sie benötigen auch zuverlässige Verschlüsselungsalgorithmen wie AES-256-GCM mit Verschlüsselungsschlüsseln pro Anwendung. Andernfalls besteht die Gefahr, dass

Daten versehentlich in falsche Hände geraten oder sogar in böswilliger Absicht kopiert werden.

WARNUNG

Und da wir gerade von böswilligen Absichten sprechen: Ihr System muss immer für Kubernetes-spezifische Malware und Ransomware-Angriffe gewappnet sein. Die Bedrohungen nehmen ständig zu, da viele Kubernetes-Anwendungen extern für Kunden zugänglich sind.

Aus diesem Grund sollten Sie sich mit einem Schutzmechanismus vertraut machen, der als *Immutability* (Unveränderlichkeit) bekannt ist. Dieser Mechanismus schützt vor der Änderung und Löschung Ihrer Backups im Objektspeicher – auch vor böswilliger Verschlüsselung oder der Löschung von Daten. Er hilft dabei, die Bedrohung durch Ransomware zu neutralisieren.

Alle diese Faktoren machen deutlich, dass eine Backup-Lösung benötigt wird, die Kubernetes-nativ ist, aber auch zuverlässige Backups erstellen kann, die nicht von Kubernetes oder dem Speichersystem abhängen. Sie muss tiefgreifende Integrationen ermöglichen, damit Wiederherstellungen schnell und automatisiert erfolgen können.

Sicherheit ist wie immer von entscheidender Bedeutung, weshalb folgende Punkte berücksichtigt werden müssen:

- Integrieren Sie eine Kubernetes-native Backup-Lösung in die Steuerungsebene.
- Erstellen Sie Self-Service-Funktionen, die die Sicherheit nicht beeinträchtigen.
- Stellen Sie sicher, dass Ihre Lösung immer Verschlüsselungsfunktionen und weitere Schutzebenen bietet.

Anmerkungen zur Mandantenfähigkeit

Herkömmliche Backup-Systeme sind in der Regel auf Administratoren ausgerichtet, und oft gibt es nur wenige Personen, die das gesamte System nutzen. Einige Benutzer haben nur Lesezugriff, während eine viel kleinere Gruppe über umfassende Berechtigungen verfügt. Für Wiederherstellungen muss möglicherweise ein Ticket eingereicht oder jemand gefragt werden.

Kubernetes-Cluster sind meist mandantenfähig, und Entwickler und Entwicklerteams werden dem System ständig dynamisch hinzugefügt und wieder entfernt. Sie haben ihre eigenen Bereiche und Anwendungen, und jeder muss sich um seine eigenen Angelegenheiten kümmern.

TIPP

Ein Backup-System muss in dieser Umgebung über Self-Service-Funktionen verfügen, damit Entwickler die Kontrolle über das Netzwerk, die Firewall, die Speicherbereitstellung und Wiederherstellungen haben

– allerdings nur für ihre eigenen Anwendungen. Das System muss so ausgelegt sein, dass Entwickler nur die Anwendungen einsehen und auf sie zugreifen können, für die sie zuständig sind.

Ein Entwickler, der zum Beispiel für Anwendung X verantwortlich ist, sollte nur Anwendung X sehen, aber auf keinen Fall Anwendung Y, die im Zuständigkeitsbereich eines anderen Entwicklers liegt. Der Entwickler von Anwendung Y wiederum sollte keinen Einblick in oder Zugriff auf Anwendung X haben.

Aber wie lässt sich das bewerkstelligen? Sie haben unter anderem eine rollenbasierte Zugriffskontrolle (wie bereits in diesem Kapitel erwähnt). Einige Entwickler benötigen eine Berechtigung für die Sicherung und Wiederherstellung, während andere vielleicht nur Zugriff auf Wiederherstellungsfunktionen haben und die Backups von jemand anderem verwaltet werden. Einige haben nur Lesezugriff. Wichtig ist, unterschiedliche Gruppen mit angemessenen Zugriffsrechten auszustatten.

Wie kann das automatisiert werden? Mit Methoden wie Open Policy Agent (OPA) können Sie Richtlinien im Wesentlichen als Code ausdrücken.

TIPP

Dabei ist entscheidend, dass der Ansatz Kubernetes-nativ ist. Getrennte Systeme für die Benutzerverwaltung, z. B. ein RBAC-System außerhalb der Plattform, sind nicht sinnvoll. Das Backup-System muss die APIs übernehmen, damit keine zusätzlichen Tools oder Benutzermanagement erforderlich sind. Alles andere wäre zu aufwendig.

Folgende Punkte sind in Bezug auf die Mandantenfähigkeit zu beachten:

- Stellen Sie sicher, dass Entwickler bei Ihren Self-Service-Optionen nur auf ihre eigenen Anwendungen zugreifen können.
- Sorgen Sie für Konsistenz, damit Sie nicht mehrere separate Systeme zur Benutzerverwaltung unterhalten müssen.

Transformation zur Unterstützung der Wiederherstellung

Kubernetes ist zweifellos eine schnelllebige Umgebung. Etwa alle drei Monate gibt es einen öffentlichen Release, sodass sich Objekte sehr schnell ändern können. Hinzu kommt, dass viele Unternehmen bei einem Upgrade gleich ein paar Versionen überspringen und in der Zwischenzeit nicht immer auf dem neuesten Stand sind. Zudem muss die mit containerbasierten Kubernetes-Umgebungen einhergehende Portierbarkeit berücksichtigt werden. Wie bringt man das alles unter einen Hut?

Mit Veränderungen Schritt halten

WARNUNG

Die Release-Zyklen werden immer kürzer. So kann es vorkommen, dass Sie ein Backup haben, das vor einem Monat oder einem Jahr erstellt wurde und deshalb veränderte Objekte nicht erkennt. Dabei geht es nicht nur um Kubernetes-APIs, sondern auch um kundenspezifische Ressourcen und andere Komponenten. In diesem Fall wird dann versucht, die Wiederherstellung mit einer API-Version durchzuführen, die das System nicht mehr unterstützt, und die Wiederherstellung schlägt fehl.

TIPP

Das System muss in der Lage sein, die Merkmale und Deskriptoren der Systemkomponenten älterer Versionen in aktualisierte Versionen zu transformieren. Es muss zum Beispiel die API des Objekts von der alten Version auf die neueste Version aktualisieren können. Oder sich mit einem kurzlebigen Zertifikat befassen. Oder mit Secrets, die aktualisiert werden müssen.

Diese Transformation ist auch wichtig, wenn Kubernetes-Cluster von einem Ort zum anderen verschoben werden (wie im folgenden Abschnitt über Portierbarkeit beschrieben). Sie möchten von einem lokalen System zu einer Public Cloud wechseln? Die Funktion „Transform" sorgt für einen einfachen Wiederherstellungsprozess und Portierbarkeit.

In einer derart schnelllebigen und dynamischen Umgebung ist es unvermeidlich, dass bestimmte Dinge veralten oder ablaufen, dass sich API-Versionen ändern, dass etwas aktualisiert werden muss oder ein Cluster sich die Portierbarkeit von Kubernetes zunutze macht. Bei der Wiederherstellung muss auch eine Transformation möglich sein.

Für die Transformation sollten Sie Folgendes beachten:

- Stellen Sie sicher, dass Ihre Lösung Transformationsfunktionen umfasst, um mit den immer schneller erscheinenden Releases Schritt halten zu können.
- Verwenden Sie „Transform" zur Gewährleistung der Portierbarkeit.

Nutzung von Portierbarkeit

Portierbarkeit ist eine der vielen großartigen Funktionalitäten, die Kubernetes zu bieten hat. Und eine Backup-Plattform kann mit diesem leistungsstarken Feature für viele neue Anwendungsfälle eingesetzt werden. Abbildung 3-3 zeigt einige der Möglichkeiten.

Im Folgenden sind einige potenzielle Anwendungsfälle für die Portierbarkeit aufgeführt:

- zwischen Namespaces im selben Cluster

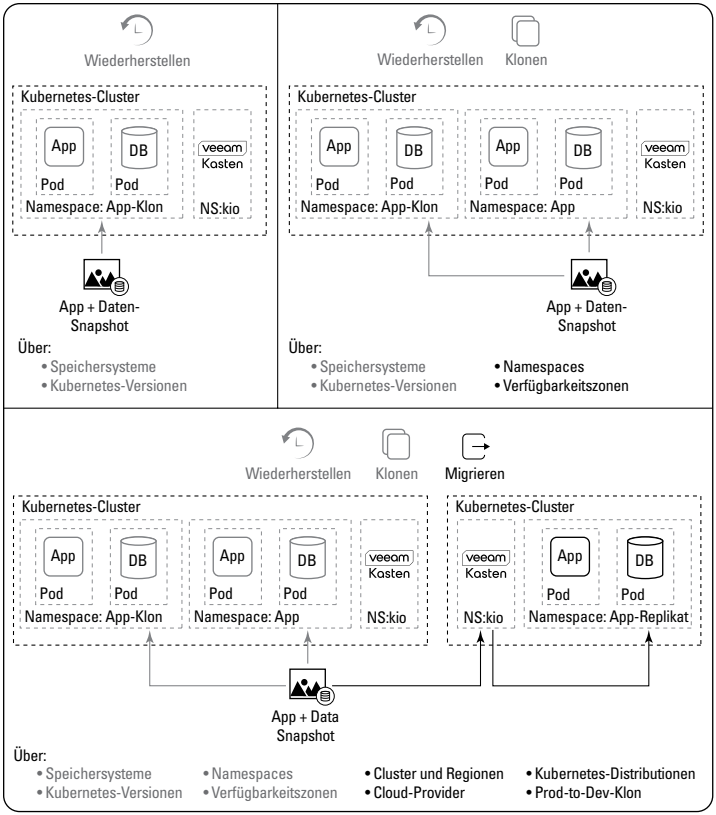

ABBILDUNG 3-3: Die Vorteile der Portierbarkeit bei Kubernetes-nativen Backups.

- zwischen Speichersystemen
- zwischen Kubernetes-Clustern, -Distributionen und -Versionen
- zwischen Verfügbarkeitszonen in derselben Region
- zwischen Regionen in derselben Cloud
- zwischen Cloud- oder Hybridumgebungen
- zwischen Test- und Entwicklungsumgebungen

NICHT VERGESSEN

Wie Sie wissen, ist das Ökosystem von Kubernetes sehr vielfältig und deckt lokale Umgebungen und Cloud-Umgebungen ab. Darüber hinaus nutzen immer mehr Unternehmen ein Hybrid-Cloud-Modell, um containerbasierte Anwendungen auszuführen. Ihre Backup- und Datenmanagement-Plattform muss daher in der Lage sein, Anwendungen über Quell- und Zielcluster hinweg zu migrieren, die auf unterschiedlichen Infrastrukturen ausgeführt werden können.

Abbildung 3-4 veranschaulicht, mit welchen Problemen Sie bei der Migration eines Workloads von Amazon Elastic Kubernetes Service zu Microsoft Azure Kubernetes Service rechnen müssen.

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
      name: gp2
provisioner: kubernetes.io/aws-ebs
parameters:
      type: gp2
      fsType: ext4
```

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
      name: managed-premium-retain
provisioner: kubernetes.io/azure-disk
reclaimPolicy: Retain
parameters:
      storageaccounttype:
Premium_LRS
```

ABBILDUNG 3-4: Migration von EKS zu AKS und wie die Terminologie angepasst werden muss.

Unterschiede zwischen Speicherklassen in Bezug auf unterschiedliche Distributionen sind erst der Anfang, selbst wenn diese Distributionen auf derselben zugrundeliegenden Kubernetes-Version basieren. Ihre Backup-Plattform muss Wiederherstellungen bei all diesen unterschiedlichen Distributionen und Infrastrukturoptionen durchführen können und Anwendungs-Backups automatisch an die Wiederherstellungsumgebung anpassen.

NICHT VERGESSEN

Das ist keine einfache Aufgabe, doch sie ist unerlässlich. Die Backup-Plattform muss in der Lage sein, alle Abhängigkeiten einer Anwendung zu verstehen und sie erfolgreich auf andere Umgebungen zu übertragen.

Für eine erfolgreiche Migration benötigen Sie einen Migrationsplan, der sicherstellt, dass Ihre Infrastruktur clusterweit abgedeckt und Anwendungsabhängigkeiten verfügbar sind oder in eine entsprechende Ressource transformiert werden. Denken Sie daran, dass Sie nicht nur Container und Speicher-Volumes migrieren müssen, sondern während der Übertragung auch Änderungen an FQDNs, Secrets und DNS-Adressen vornehmen müssen.

Zusammengefasst hängt die Portierbarkeit von Folgendem ab:

- Stellen Sie sicher, dass Ihre Backup- und Datenmanagementlösung Anwendungen über Cluster hinweg migrieren kann, die in unterschiedlichen Infrastrukturen betrieben werden.
- Erstellen Sie einen Migrationsplan, damit alle Abhängigkeiten verfügbar sind oder in eine gleichwertige Ressource transformiert werden können.

IN DIESEM KAPITEL

- Erkundung der Cloud-Umgebung
- Verbindung von Backup und Disaster Recovery
- Erfolgreich in einer Multi-Cluster-Umgebung

Kapitel 4
Cloudbasierte Anwendungsmobilität

Nach der Lektüre der vorangegangenen drei Kapitel denken Sie vielleicht, dass sich Backup und Disaster Recovery in der heutigen containerbasierten Welt kaum auf effiziente Weise durchführen lassen. Einfach ist es sicherlich nicht, aber mit den richtigen Tools ist diese Aufgabe gleich weniger überwältigend! Was Sie brauchen, ist ein zentraler Ort zur Verwaltung aller Arbeitsabläufe.

In diesem Kapitel geht es um etwas, das Sie sich schon immer gewünscht haben: Anwendungsmobilität mit zuverlässigem Backup und einer erfolgreichen, stets verfügbaren Disaster Recovery. Außerdem betrachten wir die verschwimmenden Grenzen zwischen Datensicherung und Wiederherstellung sowie die Herausforderungen, die Sie bewältigen müssen, wenn die Anzahl der Cluster und Ressourcen exponentiell zunimmt.

Ausführung in der Cloud

TIPP

Diejenigen, die noch nicht mit Kubernetes vertraut sind, sollten die Lösung zunächst in der Public Cloud testen. Kubernetes ist als Service von Providern wie Google, Microsoft und AWS verfügbar.

Public-Cloud-Provider halten den Service aufrecht und behalten Details wie Zertifikatsrotation, Version und Patching im Auge. Sie bieten auch unterschiedliche Speicheroptionen und verwalten jeden Aspekt im Zusammenhang mit dem Cluster.

Durch die Bereitstellung Ihres eigenen Kubernetes-Clusters erhalten Sie einen noch detaillierteren Einblick in den Aufbau und die Architektur. Bootstrapping ist sicherlich eine Option, aber bedenken Sie, dass Sie bei der Bereitstellung Ihres eigenen Clusters für alles verantwortlich sind, was der Serviceprovider in der Public Cloud für Sie übernehmen würde.

Verschwimmende Grenzen

Wahrscheinlich hätten Sie dieses Buch gar nicht erst aufgeschlagen, wenn Sie nicht bereits wüssten, wie wichtig Backup und Disaster Recovery für den Erfolg – vielleicht sogar für das Überleben – Ihres Unternehmens sind. Das ist mit Sicherheit nichts Neues.

Sie wissen wahrscheinlich auch schon, dass die Datensicherung und Wiederherstellung nicht unbedingt kinderleicht sind. Vor der Einführung von Kubernetes im Jahr 2015 gab es oft getrennte Plattformen für unterschiedliche Funktionen – und mehr als eine Kopie der Daten.

NICHT VERGESSEN

In der Kubernetes-Welt von heute wollen Benutzer ihre Daten nur einmal erfassen. Sie möchten, dass ein einziger Datensatz in mehreren Kontexten verwendet werden kann, z. B. für die Sicherung und Wiederherstellung, die Disaster Recovery, die Anwendungsmobilität und den Wechsel zwischen unterschiedlichen Clustern. Die Grenzen zwischen Backup und Disaster Recovery verschwimmen zunehmend, es gibt jedoch Tools, die Ihnen das Leben erleichtern können.

Diese Konzepte beziehen sich auf die Notwendigkeit, Entwicklern und Bedienern, die mehrere Aufgaben ausführen, Self-Service-Funktionen zur Verfügung zu stellen. Dies ist aufgrund der zahlreichen Anwendungsfälle wichtig und darf nicht erst im Nachhinein erfolgen. Portierbarkeit und Anwendungsmobilität sollten nicht allzu kompliziert sein. Deshalb ist die Unterstützung der Transformation (Transform) (wie in Kapitel 3 erläutert) so wichtig.

Beim Umzug in unterschiedliche Clouds oder Regionen sind Transformationen erforderlich. Doch wie erreicht man dies bei den für die Mobilität erforderlichen Features? Dies ist eine der grundlegenden Funktionen, die eine Datenmanagement-Plattform aufweisen muss.

Hinzufügen von Clustern

Viele Unternehmen betreiben eine große Anzahl von Clustern. 50 oder mehr Cluster in der Produktionsumgebung sind keine Seltenheit. Diese Unternehmen haben wahrscheinlich klein angefangen und begannen dann, in größeren Dimensionen zu denken und größere Cluster oder mehrere Cluster für unterschiedliche Zwecke einzurichten, z. B. für Tests, Entwicklung, Staging und Produktion.

Cluster können Workloads auf der Grundlage unterschiedlicher Attribute ausführen. Dazu zählen beispielsweise spezifische Anwendungen, Sicherheitsdomänen, der Bereitschaftsstatus, ein bestimmter Geschäftsbereich und eventuell auch eine geografische Untereinheit.

Es gibt viele gute Gründe für einen Multi-Cluster-Ansatz. Vorteile sind u. a. die Anpassung von Clustern an die Workload-Anforderungen (z. B. Knotengröße) und gegebenenfalls die Reduzierung des Blast-Radius. Multi-Cluster ist heutzutage Standard, und selbst viele kleine Cluster haben drei, fünf oder zehn Knoten.

NICHT VERGESSEN

Wenn man das Ganze multipliziert, werden der Aufgabenumfang und der Verwaltungsaufwand enorm. Zieht man die Anzahl der Cluster und Anwendungen sowie der Ressourcen und Volumes pro Anwendung in Betracht, wird schnell klar, dass all diese Komponenten nicht unabhängig voneinander verwaltet werden können.

TIPP

Ausführliche Fallstudien mit detaillierten Informationen zu Implementierungen zur Datensicherung und Wiederherstellung für KubernetesfindenSieunter`https://www.veeam.com/resources/customer-stories.html?product=product%3A68`.

Das liegt daran, dass Sie Backups und Berechtigungen verwalten, die mit den in den vorangegangenen Kapiteln erwähnten RBACs verknüpft sind. Das Ganze wird durch Aspekte wie Mandantenfähigkeit und Sicherheit noch zusätzlich erschwert.

Ihr Backup- und Disaster-Recovery-System sollte einen zentralen Ort vorsehen, an dem diese komplexen Aspekte verwaltet werden können und der einen umfassenden Überblick über alle Vorgänge bietet. Sie sollten in der Lage sein, detaillierte Einblicke in unterschiedliche Kontexte zu erhalten und die Mobilität zwischen diesen Clustern zu unterstützen.

Für Bediener muss es einfach sein, alles von einem Ort aus im Griff zu behalten. Auch die Arbeit der Entwickler muss vereinfacht werden. Ziel ist es, die Komplexität zu reduzieren und nicht einfach nur Cluster

hinzuzufügen, sondern die Mobilität von Anwendungen über mehrere Cluster hinweg zu ermöglichen.

Folgende Anforderungen an das Datenmanagement sind für einen reibungslosen Betrieb in Multi-Cluster-Umgebungen zu beachten:

- **Sicherheit:** Nicht nur für Daten, sondern auch für Vorgänge, damit die richtigen Benutzer und Systeme die nötigen Einblicke erhalten.
- **Einfache Einrichtung:** Sie müssen in der Lage sein, Multi-Cluster-Operationen ohne komplizierte Installationen einzurichten. Auch die Einrichtung und Verwaltung von Richtlinien und Ressourcen sollte einfach sein.
- **Automatische Erkennung:** Dazu gehört die Erkennung von Kubernetes-Anwendungen, Backup-Richtlinien und Änderungen in allen Clustern.
- **Zentrale Konsole:** Bestehen Sie auf einer einzigen Konsole, um den aktuellen Status in Echtzeit im Blick zu behalten.
- **Globale Richtlinien und Ressourcen:** Ihre Teams sollten in der Lage sein, globale Richtlinien zur Häufigkeit von Backups festzulegen und Ressourcen wie den Zielspeicherort zu bestimmen.
- **Flexible Clustergruppierung:** Benutzer können problemlos flexible und frei wählbare logische Clustergruppierungen zur Verteilung globaler Richtlinien und Ressourcen festlegen.
- **Drilldown in einzelne Cluster:** Im Rahmen dieser Übersichtsfunktion müssen Sie in der Lage sein, sich jeden einzelnen Cluster genau anzusehen.

NICHT VERGESSEN

Die Moral der Geschichte ist: Sie brauchen ein Kubernetes-natives Backup- und Datenmanagement-System, das diese Komplexität beseitigt.

IN DIESEM KAPITEL

- Informationen zum Ökosystem
- Anpassung an die Arbeitsweise der Bediener
- Auditing- und Sicherheitsrichtlinien
- Protokollierung und verbesserte Einblicke
- Aktualisierung des Backup-Systems

Kapitel 5
Eingewöhnung in ein cloudbasiertes Ökosystem

Bei einem effektiven Backup-System müssen viele Rädchen ineinandergreifen, um alle zufriedenzustellen und um Ihre Daten optimal zu schützen. Das System muss auf die Arbeitsabläufe, Präferenzen und Richtlinien des Unternehmens abgestimmt sein. Außerdem muss es mit einem Ökosystem Schritt halten, das sich in einem rasanten Tempo verändert.

Dieses Kapitel erläutert, warum das System mit den von Ihren Bedienern bevorzugten Tools und vorhandenen Sicherheitsrichtlinien und Auditing-Anforderungen integrierbar sein muss. Es wird erklärt, wie ein effektives System die Protokollierung erleichtert und Bedienern Einblicke/Observability in seine Funktionsweise gewährt. Außerdem wird erklärt, warum es so häufig neue Releases geben muss.

Das Ökosystem des Datenmanagements

Wie in Abbildung 5-1 dargestellt, besteht das Datenmanagement-Ökosystem von Kubernetes aus vier wichtigen Teilkomponenten, die bei einer Komplettlösung eine entscheidende Rolle spielen:

- **Anwendungen:** Bei einer Komplettlösung sollten Anwendungen über vorqualifizierte Integrationen mit führenden Datenservices verfügen, einschließlich relationaler und NoSQL-Systeme.
- **Kubernetes-Distributionen:** Distributionen sollten alle wichtigen cloudbasierten, verwalteten Kubernetes-Angebote sowie alle führenden lokalen Distributionen unterstützen.
- **Unterstützung für die Speicherinfrastruktur:** Um eine optimale Effizienz zu gewährleisten, sollte diese Unterstützung sowohl für Container-Speicherschnittstellen als auch für direkte Speicherintegrationen zur Verfügung stehen.
- **Sicherheitsservices:** Kubernetes-Sicherheit sollte die Sicherung der Clusterinfrastruktur, die Verwaltung der Zugriffskontrolle, den Schutz sensibler Daten und das Monitoring im Hinblick auf Schwachstellen und Ransomware-Angriffe umfassen.

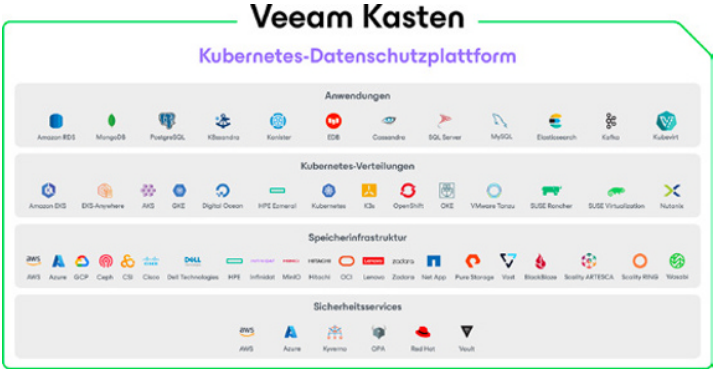

ABBILDUNG 5-1: Die vier wichtigen Bestandteile des Datenmanagement-Ökosystems von Kubernetes.

Umfassende Konsistenz, Datenbankintegrationen, automatische Anwendungserkennung, Multi-Cloud-Mobility und eine leistungsstarke webbasierte Benutzeroberfläche zeichnen eine starke Lösung aus.

Integration mit Prometheus und Grafana

Ältere Backup-Systeme haben viele nützliche Features mit sich gebracht, darunter Protokollierung, Berichterstellung, Benachrichtigung und Auditing, doch ihre Benutzerfreundlichkeit ließ oft zu wünschen übrig. Der Erfolg eines guten Backup-Systems sollte jedoch nicht nur an seiner eigentlichen Backup-Funktionalität gemessen werden, sondern auch daran, wie reibungslos es von Mitarbeitern im Operations-Team verwendet werden kann. Oder wie problemlos es sich in ihre Arbeitsabläufe und die vertraute Infrastruktur einfügen lässt.

Im Kubernetes-Ökosystem gilt Prometheus als Goldstandard für das Monitoring und die Speicherung der Metriken von Anwendungen, Software und Hardware, die zur Nachverfolgung des Systemzustands, zur Identifizierung von Problemen und zur Fehlerbehebung dienen. Grafana ist ein Open-Source-Visualisierungstool, das oft zusammen mit Prometheus verwendet wird. Ihre Kubernetes-native Backup-Plattform sollte sich in beide Lösungen integrieren lassen. Dadurch stehen Bedienern die richtigen Daten zur Verfügung, um Dashboards zu erstellen, Auslöser festzulegen, Warnmeldungen einzurichten und zusätzliche Einblicke in den Gesamtzustand des Systems zu erhalten.

Mehr Informationen durch Audits

Daten sind so wichtige Ressourcen, dass man immer genau wissen sollte, wann und wo jemand mit ihnen zu tun hatte. Ihr Backup-System muss Ihnen diese Einblicke für spätere Audits bieten.

Anstatt ein gesondertes Auditing-System bereitzustellen, sollte die Kubernetes-Backup-Lösung diese Aufgabe durch eine transparente Einbindung in das Auditing-System von Kubernetes durchführen. Dies funktioniert gut in einer mandantenfähigen Umgebung: Benutzerberechtigungen und -identitäten können bis in die Infrastruktur hinein nachverfolgt werden.

Das Backup-System muss auf den Benutzer abgestimmt sein, damit die richtigen Audit-Protokolle angezeigt werden. Es reicht nicht aus, zu wissen, dass das „Backup-System X getan hat“. Sie müssen auch wissen, dass das „Backup-System X im Auftrag von Benutzer Robert gemacht hat“. Für eine möglichst effektive und genaue Protokollierung muss diese Fähigkeit vorhanden sein.

Verknüpfung mit Netzwerkrichtlinien und Sicherheitsfunktionen

Der Sicherheitsstatus einer Anwendung darf niemals geschwächt werden. Ein Container, auf dem eine Datenbank auf einem SQL-System ausgeführt wird, darf zum Beispiel nicht der Außenwelt zugänglich gemacht werden, nur um ein Backup zu erstellen.


TIPP

Selbst wenn die Datenbank anwendungsintern ist, kann ein in die Kubernetes-Steuerungsebene eingebettetes Backup-System auf die Datenbank zugreifen, ohne sie der Außenwelt auszusetzen. Dadurch werden Bedrohungen durch Ransomware und andere bösartige Aktivitäten reduziert.

Erweiterte Protokollierung

Jeder Benutzer hat sein eigenes Protokollierungssystem. Das Backup-System muss darauf abgestimmt und an die Bedürfnisse des Unternehmens angepasst werden können.

Außerdem sollte es möglich sein, Protokolle zu extrahieren und zu verhindern, dass sie verloren gehen. Deshalb sollten Sie in Ihrer Umgebung eine ausgereifte Benutzeroberfläche mit dem Protokollierungssystem eines Drittanbieters verwenden.

Verbesserte Observability

Observability bedeutet, die Monitoring-Aspekte des Backup-Systems zu verstehen und die erforderlichen Daten zur Korrelation von Events vorliegen zu haben. Ihr Unternehmen braucht detaillierte Informationen über alle Anwendungen, die nicht einwandfrei funktionieren oder bei denen die Sicherung länger gedauert hat, weil sie stark ausgelastet waren.

Wichtig ist auch die Observability des Backend-Aspekts von Backup-Operationen, wenn Daten in einen Objektspeicher gepusht werden. Sie sollten in der Lage sein, diese Prozesse zu überwachen und den Datenfluss zu verfolgen.

Kubernetes-Release-Zyklen im Blick

Kubernetes wird alle drei Monate aktualisiert. Dies geht mit zahlreichen Systemänderungen einher, unter anderem bei der Datensicherung und Wiederherstellung. Das Backup-System muss mit diesen Änderungen Schritt halten, damit das Unternehmen jederzeit geschützt ist.

Die Tatsache, dass es so viele kommerzielle Distributionen mit ihren eigenen Release-Zyklen gibt, macht die Sache nur noch komplizierter. Oft führen Kunden auch Upgrades durch, weil sie wissen, dass ihre Version innerhalb von drei bis vier Release-Zyklen veraltet sein wird – manchmal sogar früher.

NICHT VERGESSEN

Das Backup muss daher immer auf dem neuesten Stand dieser Änderungen sein, um seine Schutzfunktion erfüllen zu können. Da sich in diesem Bereich alles in rasantem Tempo verändert, reichen ein oder zwei Updates pro Jahr einfach nicht aus.

Tatsächlich müssen Sie mit mehreren Releases pro Monat rechnen. Veeam veröffentlicht Updates zum Beispiel alle zwei Wochen, in manchen Fällen sogar noch häufiger. Dies ist die einzige Möglichkeit, um mit den Veränderungen im Rest des Ökosystems mithalten zu können.

Kapitel 6
Zehn Kernpunkte zum Thema Kubernetes-Backups

Über Kubernetes-Backups gibt es viel zu lernen. Die vorangegangenen 40+ Seiten enthalten bereits eine Menge nützlicher Informationen. Wenn Sie jedoch einen Überblick über die wichtigsten Punkte erhalten möchten, lesen Sie weiter, denn in diesem Kapitel werden einige der wichtigsten Erkenntnisse zusammengefasst.

Verständnis der Architektur

» **Schritthalten mit Kubernetes:** In der Kubernetes-Welt kann man schnell die Übersicht verlieren und verpasst am Ende viel von dem, was gerade passiert. Als Kubernetes 2015 auf den Markt kam, waren die auf der Kubernetes-Plattform ausgeführten Anwendungen zum Beispiel oft stateless konzipiert. Das ist heute nicht mehr der Fall. Diese Steuerungsebene ist ein ideales Medium für Stateless- und Stateful-Anwendungen. Das ist eine wichtige Entwicklung. Es bedeutet aber auch, dass Sie Ihre Daten besser im Auge behalten müssen.

- **Unterstützung von Entwicklern:** Bei Kubernetes geht es darum, dass Anwendungen reibungslos funktionieren. Entwickler verfolgen genau dasselbe Ziel. Sie haben zwar die Kontrolle, befinden sich aber auf ungewohntem Terrain, das die Navigation erschwert. Bei der Entwicklung von Anwendungskomponenten und der Festlegung von Infrastrukturanforderungen gehen sie heute vielleicht ganz anders vor als früher. Daraus ergeben sich einige zusätzliche Risiken. Deshalb benötigen Sie zuverlässige Backup- und Wiederherstellungsfunktionen, die speziell für dieses Ökosystem entwickelt wurden.

Fokus auf den Betrieb

- **Unterschiede bei der Bereitstellung:** Die Recheninfrastruktur von Kubernetes ist völlig anders als alles, was es bisher gab. Sie nutzen diese Infrastruktur zur Erstellung von containerbasierten Anwendungen, deren Komponenten Kubernetes auf mehrere Knoten verteilt, um die Leistung und Fehlertoleranz zu verbessern. Sie ordnen diese containerbasierten Anwendungen nicht bestimmten Servern oder virtuellen Maschinen zu. Ihre Backup-Lösung muss diese cloudbasierten Architekturmuster verstehen, um ihre Aufgabe effektiv erfüllen zu können.
- **Vergrößern und Verkleinern:** Containerbasierte Anwendungen in der Kubernetes-Umgebung können je nach Bedarf nach oben und unten skaliert werden. Die Backup-Lösung muss das ebenfalls können. Sie muss dasselbe cloudbasierte Architekturmuster aufweisen, damit sie entsprechend den Änderungen in Anwendungen und Clustern skaliert werden kann. Diese Lösung sorgt nicht nur für eine effektive Funktion, sondern ist auch noch besonders kosteneffizient.

Optimierung des Backup-Prozesses

- **Auswahl eines nativen Systems:** Sie wissen, wie wichtig Backups und Disaster Recovery sind. Gehen Sie jedoch nicht davon aus, dass Sie Ihre neuen Kubernetes-nativen Stateful-Anwendungen in eine traditionelle Backuplösung einbinden können, die für eine virtualisierungsbasierte Infrastruktur entwickelt wurde. Diese Art von Technologie unterscheidet sich grundlegend von dem, was in Kubernetes passiert. Die einzige sichere Lösung für Kubernetes-Backups ist ein Kubernetes-natives Backup-System, das genau auf Kubernetes zugeschnitten ist.

- **Verfügbarkeit ist nicht gleich Backup:** Ausfallzeiten können ausgesprochen kostspielig werden. Das ist einer der Gründe, warum containerbasierte Anwendungen so gerne auf Kubernetes ausgeführt werden. Allerdings ist die Replikation mitsamt ihren anderen hilfreichen Merkmalen und Funktionen nicht dasselbe wie ein Backup. Ihre Daten sind nach wie vor stark gefährdet, sofern Sie nicht ganz bewusst Maßnahmen zu ihrem Schutz ergreifen.
- **Gewährleistung der Wiederherstellbarkeit:** Bei der Kubernetes-nativen Datensicherung und Disaster Recovery liegt der Fokus auf den Beziehungen und Abhängigkeiten zwischen den vielen Komponenten in einem bestimmten Container. Kubernetes nimmt ständig Änderungen vor, um den Betrieb aufrechtzuerhalten und die Skalierbarkeit zu gewährleisten (man denke nur an die Burst-Kapazität, die eine Einzelhandelsanwendung am Black Friday bereitstellen muss). Das Backup muss wissen, wie alles zusammengesetzt wird. Sie brauchen einen detaillierten Wiederherstellungsplan, der Ansichten aller wiederherzustellenden Daten enthält. Außerdem müssen Sie Backup-Datenquellen identifizieren sowie den Zielspeicher festlegen und wissen, was bei der Wiederherstellung transformiert werden muss.

Gewährleistung der Sicherheit

- **Einbindung nativer Sicherheitsmaßnahmen:** Kubernetes bietet intelligente integrierte Sicherheitsmaßnahmen, die dafür sorgen, dass nur berechtigte Personen Zugriff auf Daten erhalten. Das gilt übrigens nicht nur für Menschen, sondern auch für Komponenten von Anwendungen. Wenn Ihre Backup- und Wiederherstellungslösung außerhalb Ihrer Kubernetes-Cluster ausgeführt wird, kann sie Anwendungen nicht erkennen und darauf zugreifen.
- **Pflege guter Nachbarschaft:** Haben Sie schon einmal in einem Wohnkomplex Tür an Tür mit Nachbarn zusammengelebt? Man lernt schnell, gegenseitiges Vertrauen mit einer gesunden Portion Selbstschutz zu verbinden. Genauso ist es bei Kubernetes-Clustern, bei denen Anwendungen aufgrund der Mandantenfähigkeit gemeinsam genutzte Bereiche haben. Entwickler brauchen Einblicke und Zugriff auf ihre eigenen Anwendungen, aber nicht auf die Anwendungen anderer. Das Backup-System muss denselben Regeln folgen und dieselben Zugriffskontrollen verwenden, damit alles und jeder an seinem rechtmäßigen Platz bleibt.

Schnellere Erzielung von Verbesserungen

- **Auf dem Laufenden bleiben:** Manche behalten ihr Auto, bis es 150.000 bis 200.000 Kilometer auf dem Tacho hat, während andere alle paar Jahre einen neuen Leasingvertrag abschließen. In der Kubernetes-Welt findet man keinen Computercode mit einem hohen „Kilometerstand". Wer sich in dieser Umgebung bewegt, muss mit den neuesten Trends und Technologien Schritt halten. Und Ihre Backup-Lösung muss das ebenfalls. Sie können regelmäßig mit neuen Releases rechnen. In einer Welt, in der Updates in einem schwindelerregenden Tempo herausgegeben werden, ist dies die einzige Möglichkeit, einen kontinuierlichen Schutz zu gewährleisten.

Kubernetes-native Datensicherung und Wiederherstellung, DR, Anwendungsmobilität und Ransomware-Schutz

Kubernetes ist die am schnellsten wachsende Infrastruktursoftware auf dem Markt und hat sich inzwischen zur führenden Plattform für Unternehmensanwendungen entwickelt. Die Hochverfügbarkeit und Skalierbarkeit von Anwendungsservices sind zwar wichtige Vorteile von Kubernetes, doch sie erstrecken sich nicht auf Ihre Daten. Daher muss der Datensicherheit bei Kubernetes-Workloads oberste Priorität eingeräumt werden. Dieses Buch gibt Ihnen die Informationen und Tools an die Hand, die Sie für einen effektiven Schutz Ihrer Kubernetes-Anwendungen benötigen.

Im Buch ...

- Kubernetes und cloudbasierte Anwendungen
- Kubernetes-native Datensicherheit
- Best Practices für die Datensicherung und Wiederherstellung für Kubernetes
- Cloudbasierte Anwendungsmobilität
- Ein cloudbasiertes Ökosystem

Steve Kaelble ist Autor zahlreicher Bücher der *„Für Dummies"*-Reihe. Seine Texte wurden außerdem in Zeitschriften, Zeitungen und Geschäftsberichten veröffentlicht. Wenn er sich nicht gerade in die *„Für Dummies"*-Welt vertieft oder Artikel schreibt, beschäftigt er sich mit Gesundheitskommunikation.

Besuchen Sie Dummies.com®
um sich Videos und schrittweise Bildanleitungen anzusehen oder Produkte zu kaufen!

ISBN: 978-1-394-35365-1
Nicht für den Wiederverkauf bestimmt

für dummies®
A Wiley Brand

WILEY END USER LICENSE AGREEMENT

Go to www.wiley.com/go/eula to access Wiley’s ebook EULA.