EASY Kube

PDF

HÄUFIGE KUBERNETES

SICHERHEITS-FEHLER UND MYTHEN

In diesem Artikel besprechen wir die 10 größten Sicherheitsmißverständnisse über Kubernetes. Wussten Sie, dass Namespaces keine Pods isolieren? Haben Sie Ihre Geheimnisse verschlüsselt? Kann jemand meinen Datenverkehr zwischen den Nodes abhören? All dies und mehr wird in diesem Artikel beantwortet.

KONTAKT

+1 (650) 382 0775

+49 160 1136770

easycloudhost@datafortress.cloud

easycloudhost.de/de/services/easykube/

KUBERNETES

SICHERHEITS-FEHLER UND MYTHEN

Kubernetes kann komplex sein, aber sobald man die Grundlagen kennt, ist es ein wunderbares Werkzeug, mit dem man schnell loslegen kann.
Sobald Sie Ihre ersten Dienste implementiert haben, gibt es häufige Fallstricke in Bezug auf die Sicherheit, die Sie übersehen könnten.

Dieser Artikel befasst sich mit den 7 größten Irrtümern.

Autor: Justin Güse, CEO
EasyCloudHost.de &
DataFortress.cloud

1. KUBERNETES NAMESPACES ISOLIEREN CONTAINER

Die Annahme, dass Namespaces Container isolieren, ist ein weit verbreiteter Irrglaube. Sie werden nur in diesem speziellen Namespace angezeigt, sind aber nicht voneinander isoliert. Nehmen wir an, Sie führen ein Postgresql-Deployment im "Standard"-Namensraum als Dienstname "psql-deployment" aus. Wenn Sie einen anderen Container starten, können Sie sich mit ihm verbinden, indem Sie sich mit "psql-deployment:5432" verbinden. Wenn Sie nun Ihr zweites Deployment in einen anderen Namespace verschieben, können Sie keine Verbindung mehr herstellen.

Nun könnte man meinen, dass Namespaces die Netzwerkressourcen isolieren, aber in Wahrheit hat sich nur das Routing geändert, und der Pod kann immer noch mit dem psql-Deployment kommunizieren. Wir können dies ausprobieren, indem wir das interne DNS-Routing namens "psql-deployment.default.svc.cluster.local" verwenden. Wie Sie sehen können, lautet die Struktur [SERVICE].[NAMESPACE].svc.cluster.local, und jeder Dienst kann damit erreicht werden.

Dies stellt ein großes Sicherheitsrisiko dar, kann aber durch die Verwendung spezieller Netzwerk-Plugins in Kubernetes vermieden werden. Gängige Plugins sind Calico, Flannel, Weave und Cilium. Sie können eingerichtet werden, um Namespaces und Container zu isolieren und verschiedene Sicherheitsgruppen und Zugriffskontrolllisten festzulegen, wie Sie sie vielleicht von anderen Cloud-Anbietern kennen.

KONTAKT

☎ +1 (650) 382 0775

☎ +49 160 1136770

✉ easycloudhost@datafortress.cloud

🌐 easycloudhost.de/services/easykube/

EASY Kube
Verwaltetes Kubernetes

KUBERNETES

SICHERHEITS-FEHLER UND MYTHEN

2. DIE KOMMUNIKATION ZWISCHEN DEN KNOTEN IN KUBERNETES IST VERSCHLÜSSELT

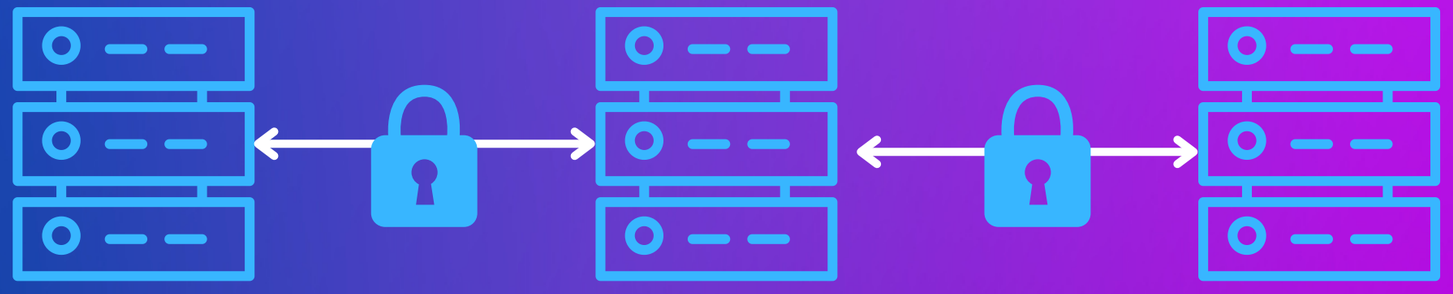
Ein ähnliches Problem tritt bei der Kommunikation zwischen Kubernetes-Knoten auf. Nehmen wir an, Sie verschlüsseln Ihren WordPress-Datenverkehr (oder jeden anderen) mit automatisiertem Cert-Manager und Letsencrypt. Alles gut, oder? Falsch - denn der Verkehr zwischen den Knoten ist standardmäßig unverschlüsselt.

Aber was bedeutet das?

Wenn sich ein Knoten in Frankfurt und ein anderer aus Verfügbarkeitsgründen in München befindet (gut für Sie!), findet die Kommunikation zwischen den Knoten trotzdem unverschlüsselt statt. Das bedeutet, dass jeder, der sich Zugang zur Rechenzentrumsverbindung verschafft, Ihren Datenverkehr mitlesen könnte!
Natürlich ist das bei einem zuverlässigen Hosting-Provider recht gut zu verhindern, aber gerade bei selbstgehosteten Setups sollte man hier vorsichtig sein.

Und was können Sie dagegen tun? Man könnte entweder manuell selbstsignierte Zertifikate für die Pods verwenden, oder - einfacher - Kubernetes-Plugins dafür nutzen. Die bereits erwähnten Networking-Plugins bieten eine Menge an Funktionalität.

Die Plugins implementieren einen so genannten "Service Mash", der zusammen mit Ihrem Pod ein Sidecar bereitstellt, das sich um die TLS-Terminierung bei aktiviertem mTLS zwischen Pods kümmert.
Calico implementiert dies nicht standardmäßig, aber Linkerd und Istio können die Verkehrsverschlüsselung in Kombination mit Calico verwenden.
Linkerd ist in dieser Hinsicht das einfachste Plugin, da es mTLS von Haus aus implementiert.



KONTAKT

+1 (650) 382 0775

+49 160 1136770

easycloudhost@datafortress.cloud

easycloudhost.de/services/easykube/

KUBERNETES

SICHERHEITS-FEHLER UND MYTHEN

3. VELERO BACKUPS SIND GESICHERT UND VERSCHLÜSSELT

Angenommen, Sie haben Backups implementiert und speichern diese zur sicheren Aufbewahrung in AWS S3. Im Hinblick auf die Notfallwiederherstellung ist das sicher ein guter Schritt, aber wussten Sie, dass Velero Ihre Backups standardmäßig nicht verschlüsselt?

Das kann schwerwiegende Folgen haben, wenn Sie einen isolierten, selbst gehosteten Kubernetes-Cluster betreiben, weil Sie aus der Finanzbranche kommen oder medizinische Daten speichern, das Backup aber in echter Form bei einem amerikanischen Cloud-Anbieter gespeichert ist.

Viele europäische Länder geben an, wegen der damit verbundenen Sicherheitsrisiken keine amerikanischen Cloud-Anbieter zu verwenden, und viele nutzen weiterhin S3 für Backups. Wenn jemand auf diese Objektspeicher in der öffentlichen Cloud zugreift, werden die gespeicherten Dateien nicht verschlüsselt.

Leider gibt es zum Zeitpunkt der Erstellung dieses Artikels keinen Backup-Dienst, der Verschlüsselung anbietet. Meine Lösung bestand darin, benutzerdefinierte Kubernetes-Jobs zu erstellen, die die Daten dumpen, sie im Job verschlüsseln und sie dann in Objektspeicher-Anbieterschnittstellen speichern.

4. KUBERNETES-GEHEIMNISSE WERDEN VERSCHLÜSSELT

Vielleicht haben Sie Geheimnisse in Kubernetes erstellt und einen Blick auf sie in Ihrem laufenden Cluster geworfen. "aWxvdmV0ZW5kaWVz", wow, das sieht ziemlich verschlüsselt aus! Aber auch hier ist die harte Wahrheit: Falsch!

Kubernetes speichert Secrets und eine Menge anderer Variablen als base64-kodierte Strings. Base64 ist nur eine Kodierung und keine Verschlüsselung, d.h. wir können die obige Zeichenkette ganz einfach, ohne Schlüssel, in die wahre Form "übersetzen". Probieren Sie es selbst mit Tools wie z.B. https://www.base64encode.org/ aus!

Das bedeutet, dass jemand, der sich Zugang zu Ihrem Kubernetes-Cluster verschafft, einfach alle Ihre Geheimnisse lesen kann.

Das ist derzeit nicht komplett zu verhindern, aber es gibt gute Lösungen.

Zum Beispiel "Sealed Secrets", die es Ihnen erlauben, Ihre Geheimnisse im Ruhezustand und bei der Erstellung zu verschlüsseln. Der Entschlüsselungsschlüssel wird im Cluster selbst gespeichert, wo auch die Geheimnisse entschlüsselt werden. Das bedeutet zwar immer noch, dass, wenn der Cluster selbst kompromittiert wird, auch der Verschlüsselungsschlüssel kompromittiert wird, aber zumindest während der Erstellung und beim Speichern der Geheimnisse auf Github bleiben sie sicher.

Ein etwas komplexerer Ansatz von Bitnami, die "Helm secrets", kann ein Schlüsselverwaltungssystem wie AWS KMS oder Google CKM zum Ver- und Entschlüsseln der Geheimnisse verwenden. Auf diese Weise müssen Sie nicht denselben Verschlüsselungsschlüssel mit allen Ihren Entwicklern teilen, sondern nur bestimmte Entwickler haben Zugriff auf die Verschlüsselungsschlüssel ihrer Anwendungen, auch im Cluster.

Dennoch bleiben die Geheimnisse im Cluster selbst "rein", was ein weiterer Grund ist, genau darauf zu achten, dass Sie Ihre Admin Kubeconfig nicht verlieren.

KUBERNETES

SICHERHEITS-FEHLER UND MYTHEN

5. VERWENDUNG DESSELBEN BENUTZERS FÜR ALLES

Dies bringt uns zu unserem nächsten Punkt. Bei der Erstellung neuer Cluster erhalten Sie von Ihrem Anbieter in der Regel eine Kubeconfig-Datei. Dies ist die Admin-Datei, mit der Sie alles machen können. Auch wenn es verlockend ist, sollten Sie sie nie verwenden, außer um andere Benutzer und Berechtigungen zu erstellen. Wenn ein Benutzer und eine Rolle für eine bestimmte Anwendung oder einen bestimmten Namespace kompromittiert wird, hat er keine Kontrolle über die anderen. Die Lösung heißt RBAC (rollenbasierte Zugriffskontrolle).

6. PERSISTENT VOLUMES WERDEN AUTOMATISCH IN ANDERE VERFÜGBARKEITSZONEN "SPRINGEN"

Auch wenn Kubernetes für die Handhabung der Persistenz hervorragend geeignet ist, muss sie zuerst konfiguriert werden. Wenn die Volumes nicht richtig konfiguriert sind, kann es passieren, dass die Anwendung monatelang auf einem Knoten läuft, aber wenn der Knoten offline geht und das Deployment automatisch zu einem anderen Knoten wechselt, fängt es mit den Daten "neu an". Dies geschieht, wenn die persistenten Volumes so konfiguriert sind, dass sie lokalen Speicher verwenden, wie es bei vielen K8s-Distributionen standardmäßig der Fall ist.

Um echte Beständigkeit zu erreichen, müssen Sie externe Volumes wie AWS EBS oder generell Blockspeicher verwenden. Sie können Ihre Lösung auch auf Bare-Metal-Servern mit Tools wie Longhorn aufbauen, aber es kann sich auszahlen, nur die bereitgestellten EBS-Volumes zu verwenden, da diese oft in mindestens drei und mehr Rechenzentren gesichert sind.
Glücklicherweise stellen viele Cloud-Anbieter die passenden Einstellungen zur Verfügung, die einfach mit kubectl apply, helm angewendet werden können, oder sie sind sogar standardmäßig enthalten.

Achten Sie jedoch darauf, dass Sie die richtigen Volumes verwenden und nicht nur lokalen Speicher, da Ihnen dies einige Wochen später ziemliche Kopfschmerzen bereiten kann.

KONTAKT

☎ +1 (650) 382 0775

☎ +49 160 1136770

✉ easycloudhost@datafortress.cloud

🌐 easycloudhost.de/services/easykube/

EASY Kube

Verwaltetes Kubernetes

KUBERNETES

SICHERHEITS-FEHLER UND MYTHEN

7. IGNORIEREN VON RESSOURCENGRENZEN

Ich weiß, es ist verlockend, Ihre Bereitstellungen einfach zu schreiben und zu veröffentlichen. Aber Sie sollten zumindest Namespace-Limits oder noch besser Deployment-Ressourcen-Limits festlegen, denn das erspart Ihnen später eine Menge Kopfschmerzen.
Kubernetes verwaltet die Ressourcen automatisch, was erstaunlich ist, aber wenn jemand einen bestimmten Ihrer Dienste mit einem DDOS-Angriff angreift (im Grunde eine Menge Datenverkehr), werden die Ressourcen für dieses Deployment fast alle Kubernetes-Ressourcen verbrauchen und alle anderen Anwendungen werden ausfallen.

Dies kann problematisch sein, wenn Ihr Unternehmen einen riesigen Cluster ohne diese Grenzen verwendet. Es muss nicht einmal ein Angriff sein. Es kann sich auch um eine Anwendung mit Fehlern handeln, die in eine Endlosschleife gerät oder einfach alle Ressourcen verbraucht. Die Lösung besteht darin, Grenzen für die Bereitstellung - oder zumindest für den Namensraum - festzulegen. Wenn ein Namespace diese harte Grenze erreicht, wird die Anwendung eingedämmt, und die anderen Anwendungen laufen weiter.

CHECKLISTE AUF DER NÄCHSTEN SEITE

Ich hoffe, diese Liste hat Ihnen einen guten Überblick darüber verschafft, was Sie tun können, um die Sicherheit Ihrer Kubernetes-Bereitstellung zu verbessern. Was war neu für Sie? Habe ich etwas übersehen? Lassen Sie es mich wissen und kontaktieren Sie mich, um Ihre Kubernetes-Projekte zu besprechen. Wussten Sie, dass ich ein sorgenfrei verwaltetes Kubernetes namens "EasyKube" anbiete? Probieren Sie es aus unter https://easycloudhost.de

KONTAKT

☎ +1 (650) 382 0775

☎ +49 160 1136770

✉ easycloudhost@datafortress.cloud

🌐 easycloudhost.de/de/services/easykube/

EASY Kube
Verwaltetes Kubernetes

KUBERNETES

SICHERHEITS-FEHLER UND MYTHEN

CHECKLISTE

- ☐ Installieren Sie ein CNI-Plugin, um Netzwerke zu isolieren
  Linkerd, Calico, Flannel, Weave, or Cilium
- ☐ Speichern Sie Backups an sicheren Orten oder verschlüsseln Sie sie
  Als deutsche Firma mit sensiblen Daten dürfen Sie kein AWS, Google und Azure verwenden!
- ☐ Verschlüsseln der Kubernetes secrets
  Sealed Secrets / Helm Secrets
- ☐ Erstellen Sie einen Benutzer für jeden Einsatz und Entwickler
  RBAC
- ☐ Einrichten von persistenten Volumes zur Verwendung von "echten" Volumes
  CSI-drivers
- ☐ Setzen Sie Deployment und Namespace Limits
  K8s Resource limits

EASY Kube

Wussten Sie, dass wir ein sorgenfreies verwaltetes Kubernetes anbieten, das all dies und mehr standardmäßig bietet? Sparen Sie Ihre wertvolle Zeit und konzentrieren Sie sich auf Ihr Kerngeschäft!

KONTAKT

☎ +1 (650) 382 0775

☎ +49 160 1136770

✉ easycloudhost@datafortress.cloud

🌐 easycloudhost.de/de/services/easykube/

EASY Services

EASY Kube

Verwaltetes Kubernetes