

9.2

Kapsayıcılarda IBM MQ

IBM

Not

Bu bilgileri ve desteklediđi ürünü kullanmadan önce, "[Özel notlar](#)" sayfa 157 bölümündeki bilgileri okuyun.

This edition applies to version 9 release 2 of IBM® MQ and to all subsequent releases and modifications until otherwise indicated in new editions.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 2007, 2024.**

İçindekiler

Taşıyıcılarda IBM MQ ve IBM Cloud Pak for Integration.....	5
Taşıyıcılarda IBM MQ için planlama.....	5
Kapsayıcılarda IBM MQ 'ı nasıl kullanmak istediğinizi seçme.....	5
IBM MQ Operator desteği.....	6
IBM MQ Operator bağımlılıkları.....	9
IBM MQ Operator için gerekli olan küme kapsamlı izinler.....	10
IBM MQ Operator ile ilgili depolama konuları.....	11
Kendi IBM MQ kuyruk yöneticisi taşıyıcı görüntülerinizi oluşturma desteği.....	12
Taşıyıcılarda IBM MQ için yüksek kullanılabilirlik.....	15
Taşıyıcılarda IBM MQ için olağanüstü durumdan kurtarma.....	17
Taşıyıcılarda IBM MQ için kullanıcı kimlik doğrulaması ve yetkilendirmesi.....	18
Kapsayıcılarda IBM MQ için ölçeklenebilirliğin ve performansın planlanması.....	18
IBM Cloud Pak for Integration ve Red Hat OpenShift içinde IBM MQ kullanılması.....	19
IBM MQ Operator için yayın geçmişi.....	19
IBM MQ 'un IBM Cloud Pak for Integration' e geçirilmesi.....	36
Red Hat OpenShift üzerinde IBM MQ Operator ürününün kurulması ve kaldırılması.....	58
IBM MQ Operator ve kuyruk yöneticilerinin yükseltilmesi.....	70
Deploying and configuring queue managers using the IBM MQ Operator.....	78
Operating IBM MQ using the IBM MQ Operator.....	113
IBM MQ Operator ile ilgili sorunların giderilmesi.....	123
IBM MQ Operator için API başvurusu.....	124
Kendi IBM MQ taşıyıcınız ve konuşlandırma kodunuzu oluşturma.....	145
Bir taşıyıcıyı kullanarak kendi IBM MQ kuyruk yöneticisi görüntüsünün planlanması.....	146
Örnek bir IBM MQ kuyruk yöneticisi taşıyıcı görüntüsü oluşturulması.....	146
Yerel bağ tanımlama uygulamalarının ayrı taşıyıcılarda çalıştırılması.....	149
Kendi taşıyıcılarınız yaratılıyorsa, Yerel HA grubunun yaratılması.....	151
Özel notlar.....	157
Programlama arabirimi bilgileri.....	158
Ticari Markalar.....	158

Multi Taşıyıcılarda IBM MQ ve IBM Cloud Pak for Integration

Kapsayıcılar, bir IBM MQ kuyruk yöneticisi ya da IBM MQ istemci uygulamasını, tüm bağımlılıkları ile yazılım geliştirme için standartlaştırılmış bir birime paketlemenizi sağlar.

You can run IBM MQ using the IBM MQ Operator on Red Hat® OpenShift®. Bu işlem IBM Cloud Pak for Integration, IBM MQ Advanced ya da IBM MQ Advanced for Developers kullanılarak yapılabilir.

IBM MQ ' i kendi oluşturduğunuz bir taşıyıcıda da çalıştırabilirsiniz.

  IBM MQ Operatöre ilgili daha fazla bilgi için aşağıdaki bağlantılara bakın.

Multi Taşıyıcılarda IBM MQ için planlama

When planning for IBM MQ in containers, consider the support that IBM MQ provides for various architectural options, such as how high availability is managed, and how to secure your queue managers.

Bu görev hakkında

Before you plan your IBM MQ in containers architecture, you should familiarize yourself with the basic IBM MQ concepts (see [IBM MQ Teknik genel bakış](#)) as well as basic Kubernetes/Red Hat OpenShift concepts (see [Red Hat OpenShift Container Platform mimarisi](#)).

Yordam

- [“Kapsayıcılarda IBM MQ ' ı nasıl kullanmak istediğinizi seçme” sayfa 5.](#)
- [“IBM MQ Operatör desteği” sayfa 6.](#)
- [“Kendi IBM MQ kuyruk yöneticisi taşıyıcı görüntülerinizi oluşturma desteği” sayfa 12.](#)
- [“IBM MQ Operatöre ilgili depolama konuları” sayfa 11.](#)
- [“Taşıyıcılarda IBM MQ için yüksek kullanılabilirlik” sayfa 15.](#)
- [“Taşıyıcılarda IBM MQ için olağanüstü durumdaki kurtarma” sayfa 17.](#)
- [“Taşıyıcılarda IBM MQ için kullanıcı kimlik doğrulaması ve yetkilendirmesi” sayfa 18.](#)

Kapsayıcılarda IBM MQ ' ı nasıl kullanmak istediğinizi seçme

There are multiple options for using IBM MQ in containers: you can choose to use the IBM MQ Operator, which uses pre-packaged container images, or you can build your own images and deployment code.

IBM MQ Operatör komutunu kullanma

Red Hat OpenShift Container Platform'ta konuşlandırmayı planlıyorsanız, büyük olasılıkla IBM MQ Operatör' u kullanmak isteyebilirsiniz.

IBM MQ Operator , Red Hat OpenShift Container Platform' a yeni bir QueueManager özel kaynağı ekler. İşletmen, yeni kuyruk yöneticisi tanımlamalarını izler ve bunları StatefulSet ve Service kaynakları gibi gerekli düşük düzeyli kaynaklara dönüştürür. Yerel HA ' nın durumunda, işletmen, kuyruk yöneticisi eşgörünümlerinin karmaşık yuvarlama güncellemelerini de gerçekleştirebilir. Bkz. [“Yerel HA kuyruk yöneticisine ilişkin kendi hareketli güncellenenizi gerçekleştirirken dikkat edilecek noktalar” sayfa 153](#)

Some IBM MQ features are not supported when using the IBM MQ Operator. Aşağıdakilerden herhangi birini yapmak istiyorsanız, kendi resimlerinizi ve grafiklerinizi oluşturmanız gerekir:

- Denetim ya da ileti alışverişi için REST API ' larını kullan

- Aşağıdaki MQ bileşenlerinden herhangi birini kullanın:
 - Managed File Transfer araçları ve kaynakları. Ancak, bir ya da daha çok Coordination, Command ya da Agent kuyruk yöneticisi sağlamak için IBM MQ Operator komutunu kullanabilirsiniz.
 - AMQP
 - IBM MQ Bridge to Salesforce
 - IBM MQ Bridge to blockchain (taşıyıcılarda desteklenmez)
 - IBM MQ Telemetry Transport (MQTT).
- Günlük dosyası sayfalarını yapılandırma gibi **crtmqm**, **strmqm** ve **endmqm** ile kullanılan özelleştirme seçeneklerini özelleştirin. Çoğu seçenek bir INI dosyası kullanılarak yapılandırılabilir.

IBM MQ Operator ve kapsayıcıların hızla geliştiğini ve bu nedenle Long Term Support yayınlarının altında desteklenmediğini unutmayın.

IBM MQ Operator , önceden oluşturulmuş taşıyıcı görüntülerinin yanı sıra, Red Hat OpenShift Container Platform'ünde çalıştırılmasına ilişkin konuşlandırma kodunu da içerir. IBM MQ Operator , sağlanan IBM MQ taşıyıcı görüntüsünü ya da bunun üzerine bir kap görüntüsü katmanlı olarak konuşlandırmak için kullanılabilir, ancak özel olarak oluşturulmuş MQ taşıyıcı görüntülerini konuşlandırmak için kullanılamaz.

Kendi görüntülerinizi ve konuşlandırma kodunuzu oluşturma

Multi

Bu, en esnek taşıyıcı çözümdür, ancak kapsayıcıların yapılandırılmasına ilişkin güçlü becerilere ve sonuçta elde edilen kapsayıcıya "Kendi" ' e gereksinim duymanızı gerektirir. Red Hat OpenShift Container Platform' u kullanmayı planlamıyorsanız, kendi görüntülerinizi ve konuşlandırma kodunuzu oluşturmanız gerekir.

Kendi resimlerinizi oluşturmak için örnekler kullanılabilir. Bkz. [“Kendi IBM MQ taşıyıcınızı ve konuşlandırma kodunuzu oluşturma” sayfa 145.](#)

İlgili kavramlar

[“IBM MQ Operator desteği” sayfa 6](#)

IBM MQ Operator yalnızca Red Hat OpenShift Container Platform'ünde konuşlandırıldığında desteklenir.

[“Kendi IBM MQ kuyruk yöneticisi taşıyıcı görüntülerinizi oluşturma desteği” sayfa 12](#)

IBM MQ , GitHub'ünde bir IBM MQ kuyruk yöneticisi kapsayıcısı oluşturmak için kod sağlar. Bu, IBM ' un kendi desteklenen kapsayıcısını oluşturmak için kullandığı sürece dayanır ve kendi kapsayıcı görüntülerinizin oluşturulmasını basitleştirmek ve hızlandırmak için bu GitHub havuzunu kullanabilirsiniz.

OpenShift CP4I CD EUS IBM MQ Operator desteği

IBM MQ Operator yalnızca Red Hat OpenShift Container Platform'ünde konuşlandırıldığında desteklenir.

The IBM MQ Operator uses images based on IBM MQ Continuous Delivery (CD) releases, though an Extended Update Support (EUS) release is available with IBM Cloud Pak for Integration. CD yayınları en çok bir yıl ya da iki CD yayın için desteklenir; hangisi daha uzunsa. IBM MQ ' in Long Term Support yayınları, IBM MQ Operator aracılığıyla kullanılamaz. IBM Cloud Pak for Integration 2020.4.1 is an Extended Update Support (EUS) release, which is supported for 18 months, if you use a version of IBM MQ marked as -eus. Ters durumda, IBM MQ 9.2 , IBM MQ Operator ile Continuous Delivery yayın düzeyi olarak kabul edilir.

IBM MQ Operator , IBM MQ tarafından kullanılan anahtar Linux® kitaplıkları ve yardımcı programları içeren bir Red Hat Universal Base Image (UBI) üzerinde IBM MQ kurulumu sağlayan kapsayıcı görüntülerini kullanır. UBI, Red Hat OpenShift' ta çalıştırıldığında Red Hat tarafından desteklenmektedir.

IBM MQ Operator , amd64 ve s390x (z/Linux) mimarilerinde desteklenir.

İlgili kavramlar

[“Kendi IBM MQ kuyruk yöneticisi taşıyıcı görüntülerinizi oluşturma desteği” sayfa 12](#)

IBM MQ , GitHub üzerinde bir IBM MQ kuyruk yöneticisi kapsayıcısı oluşturmak için kod sağlar. Bu, IBM ' un kendi desteklenen kapsayıcısını oluşturmak için kullandığı sürece dayanır ve kendi kapsayıcı görüntülerinizin oluşturulmasını basitleştirmek ve hızlandırmak için bu GitHub havuzunu kullanabilirsiniz.

OpenShift CP4I CD EUS IBM MQ Operator için sürüm desteği

A mapping between supported versions of IBM MQ, Red Hat OpenShift Container Platform and IBM Cloud Pak for Integration.

- [“Kullanılabilir IBM MQ sürümleri” sayfa 7](#)
- [“Uyumlu Red Hat OpenShift Container Platform sürümleri” sayfa 7](#)
- [“IBM Cloud Pak for Integration sürümleri” sayfa 8](#)
- [“Eski işleçlerdeki kullanılabilir IBM MQ sürümleri” sayfa 8](#)
- [“Eski işleçler için uyumlu Red Hat OpenShift Container Platform sürümleri” sayfa 9](#)

Kullanılabilir IBM MQ sürümleri

İşletme kanalı	İşleç sürümü	IBM MQ sürümler							
		9.1.5	9.2.0 CD 'si	9.2.0 EUS	9.2.1	9.2.2	9.2.3	9.2.4	9.2.5
v1.6	1.6	⚠	⚠	→	⚠	●	●		
v1.7	1.7	⚠	⚠	→	⚠	●	●	●	
v1.8	1.8	⚠	⚠	→	⚠	⚠	●	●	●

Anahtar:

- Continuous Delivery desteği var
- Extended Update Support Kullanılabilir
- Yalnızca Extended Update Support işleneninden bir Continuous Delivery işlenenine geçiş sırasında kullanılabilir.
- ⚠ Kullanımdan kaldırıldı. IBM MQ yayın düzeyleri desteğinden çıktıkça, bunlar hala işlecin içinde yapılandırılabilir, ancak artık destek için uygun değildir ve gelecekteki yayınlarda kaldırılabilir.

Her bir sürümdeki ayrıntılı özellikler, değişiklikler ve düzeltmeler de içinde olmak üzere, her bir sürümün tam ayrıntıları için [“IBM MQ Operator için yayın geçmişi” sayfa 19](#) ' e bakın.

Uyumlu Red Hat OpenShift Container Platform sürümleri

İşletme kanalı	İşleç sürümü	Red Hat OpenShift Container Platform sürümler ¹				
		4.6	4.7 ²	4.8	4.9	4.10
v1.6	1.6	●	●	●	●	●
v1.7	1.7	●	●	●	●	●

¹ Red Hat OpenShift Container Platform sürümleri kendi destek tarihlerine tabidir. Daha fazla bilgi için bkz. [Red Hat OpenShift Container Platform Yaşam Çevrimi İlkesi](#) .

² The IBM MQ Operator depends on IBM Cloud Pak foundational services. Red Hat OpenShift Container Platform 4.7 kullanmak istiyorsanız, öncelikle IBM Cloud Pak foundational services sürümünü yükseltmeniz gerekir.

İşletmen kanalı	İşleç sürümü	Red Hat OpenShift Container Platform sürümler ¹				
		4.6	4.7 ²	4.8	4.9	4.10
v1.8	1.8	●	●	●	●	●

Anahtar:

- Continuous Delivery desteği var
- Extended Update Support Kullanılabilir

IBM Cloud Pak for Integration sürümler

IBM MQ Operator 1.8.x , IBM Cloud Pak for Integration sürüm 2021.4.1' in bir parçası olarak ya da bağımsız olarak kullanılmak üzere desteklenir.

IBM MQ Operator 1.7.x , IBM Cloud Pak for Integration sürüm 2021.4.1' in bir parçası olarak ya da bağımsız olarak kullanılmak üzere desteklenir.

IBM MQ Operator 1.6.x , IBM Cloud Pak for Integration sürüm 2021.2.1, 2021.3.1ya da bağımsız olarak kullanım için desteklenir.

IBM MQ Operator 1.5.x artık desteklenmemektedir.

IBM MQ Operator 1.4.x artık desteklenmemektedir.

IBM MQ Operator 1.3.x artık desteklenmemektedir.

IBM MQ Operator 1.2.x artık desteklenmemektedir.

IBM MQ Operators 1.1.x ve 1.0.x artık desteklenmemektedir.

Eski işleçlerdeki kullanılabilir IBM MQ sürümleri

Aşağıdaki çizelge, artık "hayatın sonu" na ulaşmış olan IBM MQ Operator sürümleri için geçerlidir.

İşletmen kanalı	İşleç sürümü	IBM MQ sürümler							
		9.1.5	9.2.0 CD 'si	9.2.0 EUS	9.2.1	9.2.2	9.2.3	9.2.4	9.2.5
v1.0	1.0	⚠							
v1.1	1.1	⚠	⚠						
v1.2	1.2	⚠	⚠						
v1.3-eus	1.3	⚠	⚠	⚠					
v1.4	1.4	⚠	⚠	→	⚠				
v1.5	1.5	⚠	⚠	→	⚠	⚠			

Anahtar:

¹ Red Hat OpenShift Container Platform sürümleri kendi destek tarihlerine tabidir. Daha fazla bilgi için bkz. [Red Hat OpenShift Container Platform Yaşam Çevrimi İlkesi](#) .

² The IBM MQ Operator depends on IBM Cloud Pak foundational services. Red Hat OpenShift Container Platform 4.7kullanmak istiyorsanız, öncelikle IBM Cloud Pak foundational servissürümünü yükseltmeniz gerekir.



Yalnızca Extended Update Support işleneninden bir Continuous Delivery işlenenine geçiş sırasında kullanılabilir.



Kullanımdan kaldırıldı. IBM MQ yayın düzeyleri destekten çıktıkça, bunlar IBM MQ Operator' de yapılandırılabilir, ancak artık destek için uygun değildir.

Her bir sürümdeki ayrıntılı özellikler, değişiklikler ve düzeltmeler de içinde olmak üzere, her bir sürümün tam ayrıntıları için ["IBM MQ Operator için yayın geçmişi" sayfa 19](#) ' e bakın.

Eski işlemler için uyumlu Red Hat OpenShift Container Platform sürümleri

Aşağıdaki çizelge, artık "hayatın sonu" na ulaşmış olan IBM MQ Operator sürümleri için geçerlidir.

İşletmen kanalı	İşleç sürümü	Red Hat OpenShift Container Platform sürümler ³						
		4.4 ⁴	4.5 ⁵	4.6	4.7 ⁶	4.8	4.9	4.10
v1.0	1.0	⚠	⚠	⚠	⚠			
v1.1	1.1	⚠	⚠	⚠	⚠	⚠		
v1.2	1.2	⚠	⚠	⚠	⚠	⚠		
v1.3-eus	1.3			⚠	→	→	→	→
v1.4	1.4			⚠	⚠	⚠	⚠	
v1.5	1.5			⚠	⚠	⚠	⚠	⚠

Anahtar:



Yalnızca Extended Update Support işleneninden bir Continuous Delivery işlenenine geçiş sırasında kullanılabilir.



IBM MQ Operator sürümü "kullanım ömrünün sonuna" ulaştı, ancak daha önce bu Red Hat OpenShift Container Platform sürümünde kullanılmıştı

OpenShift CP4I IBM MQ Operator bağımlılıkları

IBM MQ Operator , IBM Operand Deployment Lifecycle Manager (ODLM) İşletmeni 'ni de kuran IBM Cloud Pak foundational services işlecinin bağımlıdır. IBM MQ Operator' u kurduğunuzda bu işlemler otomatik olarak kurulacaktır. Bu bağımlı işlemlerin küçük bir CPU ve bellek alanı vardır ve bazı durumlarda ek kaynakları devreye almak için kullanılır.

Bir QueueManager oluşturduğunuzda, IBM MQ Operator , gereksinim duyduğu ek hizmetler için bir OperandRequest oluşturur. OperandRequest , ODLM İşleci tarafından yerine getirilir ve gerekiyorsa, gerekli hizmetleri kurar ve somutlaştırır. Gerekli olan hizmetler, kuyruk yöneticisi konuşlandırılırken kabul edilen lisans sözleşmesine ve kuyruk yöneticisi bileşenlerinin istenmesine bağlı olarak belirlenir.

³ Red Hat OpenShift Container Platform sürümleri kendi destek tarihlerine tabidir. Daha fazla bilgi için bkz. [Red Hat OpenShift Container Platform Yaşam Çevrimi İlkesi](#) .

⁴ Red Hat OpenShift Container Platform 4.4 , "kullanım ömrünün sonuna" ulaştı. Daha fazla bilgi için bkz. [Red Hat OpenShift Container Platform Yaşam Çevrimi İlkesi](#) .

⁵ Red Hat OpenShift Container Platform 4.5 , "kullanım ömrünün sonuna" ulaştı. Daha fazla bilgi için bkz. [Red Hat OpenShift Container Platform Yaşam Çevrimi İlkesi](#) .

⁶ The IBM MQ Operator depends on IBM Cloud Pak foundational services. Red Hat OpenShift Container Platform 4.7 kullanmak istiyorsanız, öncelikle IBM Cloud Pak foundational services sürümünü yükseltmeniz gerekir.

- Bir IBM MQ Advanced ya da IBM MQ Advanced for Developers lisansı seçerseniz, ek hizmet isteğinde bulunmuyorsunuz. Örneğin, aşağıdaki durumda, IBM Cloud Pak foundational services kullanılmamış:

```
spec:
  license:
    accept: true
    license: L-APIG-BZDDDY
    use: "Production"
```

- Bir IBM Cloud Pak for Integration lisansı seçerseniz ve web sunucusunu etkinleştirmeyi seçerseniz, tek oturum açma özelliğini etkinleştirmek için IBM MQ Operator , IBM Identity and Access Management (IAM) işlecini de örnek olarak başlatacaktır. IAM İşleci, IBM Cloud Pak for Integration işlecini kurduysa zaten kullanılabilir olacaktır. Örneğin:

```
spec:
  license:
    accept: true
    license: L-RJON-BUVMQX
    use: "Production"
```

Ancak, web sunucusunu devre dışı bırakırsanız, IBM Cloud Pak foundational services isteğinde bulunmuyorsunuz. Örneğin:

```
spec:
  license:
    accept: true
    license: L-RJON-BUVMQX
    use: "Production"
  web:
    enabled: false
```

Older versions of the IBM MQ Operator always requested the installation of the IBM Licensing Operator (and its dependencies), to track license use. IBM MQ Operator 1.5 ' den başlayarak, lisanslama hizmeti istenmez ve ayrıca bunu ayrı olarak istemeniz gerekir.

IBM MQ Operator için 1 CPU çekirdeği ve 1 GB bellek gerekir. Bağımlı işleçler için donanım ve yazılım gereksinimlerinin ayrıntılı bir dökümü için [Donanım gereksinimleri ve dökümhane hizmetleri için öneriler](#) başlıklı konuya bakın.

Kuyruk yöneticilerinizin kullandığı CPU ve bellek miktarını seçebilirsiniz. Ek bilgi için [“.spec.queueManager.resources” sayfa 134](#) başlıklı konuya bakın.

İlgili başvurular

[“mq.ibm.com/v1beta1 için lisans başvurusu” sayfa 124](#)

OpenShift CP4I IBM MQ Operator için gerekli olan küme kapsamlı izinler

IBM MQ Operator , kabul web kancalarını ve örneklerini yönetmek ve depolama sınıfı ve küme sürümü bilgilerini okumak için küme kapsamlı izinlerin kullanılmasını gerektirir.

IBM MQ Operator , aşağıdaki küme kapsamı izinlerinin kullanılmasını gerektirir:

- Web kancalarını kabul etme izni. Bu, İşletmen tarafından sağlanan kapsayıcıları oluşturma ve yönetme işleminde kullanılan belirli web kancalarının oluşturulmasına, alınmasına ve güncellenmesine olanak sağlar.
 - API Grupları: **admissionregistration.k8s.io**
 - Kaynaklar: **validatingwebhookconfigurations**
 - Fiiller: **create, get, update**
- Özel kaynaklar yaratırken örnek ve parçacıklar sağlamak için Red Hat OpenShift konsolunda kullanılan kaynakları yaratma ve yönetme izninin kullanılması.
 - API Grupları: **console.openshift.io**
 - Kaynaklar: **consoleyamlsamples**
 - Fiiller: **create, get, update, delete**

- Küme sürümünü okuma izni. Bu, İşletmen 'in küme ortamıyla ilgili tüm sorunları beslemesini sağlar.
 - API Grupları: **config.openshift.io**
 - Kaynaklar: **clusterversions**
 - Fiiller: **get, list, watch**
- Kümedeki depolama sınıflarını okuma izni. Bu, İşletmen 'in seçilen depolama sınıflarıyla olan sorunları kaplarda beslemesini sağlar.
 - API Grupları: **storage.k8s.io**
 - Kaynaklar: **storageclasses**
 - Fiiller: **get, list**

OpenShift CP4I Kubernetes IBM MQ Operatorile ilgili depolama konuları

IBM MQ Operator iki depolama kipinde çalışır:

- Taşıyıcı yeniden başlatıldığında, taşıyıcıya ilişkin tüm durum bilgileri atılabilirken **Ephemeral deposu** kullanılır. Bu, gösterim için ortamlar yaratıldığında ya da bağımsız kuyruk yöneticileriyle geliştirildiğinde yaygın olarak kullanılır.
- **Kalıcı depolama**, IBM MQ için ortak yapılandırmadır ve taşıyıcı yeniden başlatılırsa, yeniden başlatılan kapsayıcıda var olan yapılandırmanın, günlüklerin ve kalıcı iletilerin kullanılabilir olduğunu doğrular.

IBM MQ Operator, ortama bağlı olarak farklılık gösterebilen depolama özelliklerini ve istenen depolama kipini özelleştirme yeteneği sağlar.

Ephemeral deposu

IBM MQ, durumlu bir uygulamadır ve yeniden başlatma durumunda kurtarma için bu durumu depolamaya devam eder. Ephemeral deposu kullanılıyorsa, kuyruk yöneticisine ilişkin tüm durum bilgileri yeniden başlatıldığında kaybedilir. Bu bilgiler şunları içerir:

- Tüm İletiler
- Kuyruk yöneticisinin tüm kuyruk yöneticisi iletişim durumu (kanal ileti sıra numaraları)
- Kuyruk yöneticisinin MQ Küme kimliği
- Tüm hareket durumu
- Tüm kuyruk yöneticisi yapılanışı
- Tüm yerel tanımlama verileri

Bu nedenle, ephemeral depolamanın bir üretim, test veya geliştirme senaryosu için uygun bir yaklaşım olup olmadığını göz önünde bulundurmanız gerekir. Örneğin, tüm iletilerin kalıcı olmadığı bilindiği ve kuyruk yöneticisi bir MQ Cluster üyesi değil. Tüm ileti sistemi durumunu yeniden başlatmanın yanı sıra, kuyruk yöneticisinin yapılandırması da atılır. To enable a completely ephemeral container the IBM MQ configuration must be added to the container image itself (for more information, see “Red Hat OpenShift CLI kullanılarak özel MQSC ve INI dosyaları içeren bir resim oluşturma” sayfa 111). Bu işlem tamamlanmazsa, taşıyıcının her yeniden başlatıldığı her defasında IBM MQ ' in yapılandırılması gerekir.

OpenShift CP4I For example, to configure IBM MQ with ephemeral storage the storage type of the QueueManager should include the following:

```
queueManager:
  storage:
    queueManager:
      type: ephemeral
```

Kalıcı depolama

OpenShift CP4I

IBM MQ olağan durumda, kuyruk yöneticisinin yeniden başlatma işleminden sonra kalıcı iletilerini ve yapılandırmasını korumasını sağlamak için kalıcılık depolamasıyla çalışır. Bu nedenle, varsayılan davranış budur. Çeşitli depolama sağlayıcıları ve her bir destek için farklı yetenekler nedeniyle, bu genellikle yapılandırmanın özelleştirilmesinin gerekli olduğu anlamına gelir. Aşağıdaki örnekte, v1beta1 API 'sında MQ depolama yapılandırmasını özelleştiren ortak alanlar gösterilmektedir:

- `spec.queueManager.availability` , kullanılabilirlik modunu denetler. `SingleInstance` kullanıyorsanız, yalnızca `ReadWriteOnce` depolama alanı gereklidir, `MultiInstance` ise doğru dosya kilitleme özelliklerine sahip `ReadWriteMany` ' i destekleyen bir depolama sınıfı gerektirir. IBM MQ , bir destek bildirim ve bir test bildirim sağlar. Kullanılabilirlik kipi, kalıcı birim düzenini de etkiler. Daha fazla bilgi için bkz. “[Taşıyıcılarda IBM MQ için yüksek kullanılabilirlik](#)” sayfa 15
- `spec.queueManager.storage` , tek tek depolama ayarlarını denetler. Bir kuyruk yöneticisi, bir ile dört kalıcı birim arasında kullanılmak üzere yapılandırılabilir

Aşağıdaki örnekte, tek yönetim ortamı kuyruk yöneticisi kullanan basit bir yapılandırmanın bir parçası gösterilmiştir:

```
spec:
  queueManager:
    storage:
      queueManager:
        enabled: true
```

Aşağıdaki örnek, varsayılan olmayan bir depolama sınıfından ve ek gruplar gerektiren dosya depolamasıyla, çok eşgörsümlü bir kuyruk yöneticisi yapılandırmasının parçacıklarını göstermektedir:

```
spec:
  queueManager:
    availability:
      type: MultiInstance
    storage:
      queueManager:
        class: ibmc-file-gold-gid
      persistedData:
        enabled: true
        class: ibmc-file-gold-gid
      recoveryLogs:
        enabled: true
        class: ibmc-file-gold-gid
    securityContext:
      supplementalGroups: [99]
```

Not: Ek grupları tek eşgörsüm kuyruk yöneticileriyle de yapılandırabilirsiniz.

Not: Yerel HA ' yı kullanıyorsanız, paylaşılan dosya sistemlerine gerek duymuyorsunuz (bkz. “[Taşıyıcılarda IBM MQ için yüksek kullanılabilirlik](#)” sayfa 15). Özellikle NFSv3' i kullanmamanız gerekir.

Linux Kendi IBM MQ kuyruk yöneticisi taşıyıcı görüntülerinizi oluşturma desteği

IBM MQ , GitHub üzerinde bir IBM MQ kuyruk yöneticisi kapsayıcısı oluşturmak için kod sağlar. Bu, IBM ' un kendi desteklenen kapsayıcısını oluşturmak için kullandığı sürece dayanır ve kendi kapsayıcı görüntülerinizin oluşturulmasını basitleştirmek ve hızlandırmak için bu GitHub havuzunu kullanabilirsiniz.

Kod, mq-container GitHub havuzunda şu adrese sağlanır: <https://github.com/ibm-messaging/mq-container>. Bu, topluluk tarafından sağlanan destekle birlikte bir Apache 2.0 lisansı altında sağlanır.

Havuz, standart Linux rpm paketlerini kullanmaz; kapsayıcı devreye alımları için sıkıştırılmış paketi kullanır. Bu yaklaşımın yararı, üst seviyeye iletilmiş izinlere gerek kalmadan daha güvenli kapsayıcı ortamlarında çalışabilmenizdir. Ancak bu, IBM MQ geleneksel olarak işletim sistemi tabanlı kimlik doğrulaması için yükseltilmiş izinleri kullandığından, kullanılabilir güvenlik seçeneklerini etkiler. Kapsayıcı devreye alımı için, işletim sistemi tabanlı kimlik doğrulamasını kullanmak normalde iyi bir uygulama değildir; bunun yerine karşılıklı TLS ya da LDAP kimlik doğrulamasını kullanabilirsiniz. IBM MQ Advanced for Developers ile dosya tabanlı kimlik doğrulamasını kullanarak kullanıcılarınızın hızlı bir şekilde çalışmaya başlamasını sağlayabilirsiniz.

Eşlenen veri kuyruğu yöneticisi (RDQM) bir kapsayıcı ortamında desteklenmiyor. [“Yerel HA” sayfa 91](#) komutunu kullanarak RDQM ' ye benzer yetenekler elde edebilirsiniz.

İlgili kavramlar

[“IBM MQ Operator desteği” sayfa 6](#)

IBM MQ Operator yalnızca Red Hat OpenShift Container Platform üzerinde konuşlandırıldığında desteklenir.

[IBM MQ kuruluş dışı görüntüler](#)

Linux Kendi IBM MQ taşıyıcı görüntüsünü oluştururken lisans ek açıklamaları

Lisans ek açıklamaları, temel makine yerine, taşıyıcıda tanımlanan sınırlara dayalı olarak kullanımı izlemenize olanak sağlar. You configure your clients to deploy the container with specific annotations that the IBM License Service then uses to track usage.

Kendi kendine oluşturulmuş bir IBM MQ kapsayıcısı görüntüsünü devreye alırken, lisanslamaya ilişkin iki ortak yaklaşım vardır:

- Taşıyıcıyı çalıştıran tüm makineyi lisanslayın.
- Taşıyıcıyı ilişkili sınırlara dayalı olarak lisanslayın.

Her iki seçenek de istemciler tarafından kullanılabilir ve daha fazla ayrıntı, [Passport Advantage üzerindeki IBM Kapsayıcı Lisansları sayfasında](#) bulunabilir.

If the IBM MQ container is to be licensed based on the container limits, then the IBM License Service needs to be installed to track usage. Desteklenen ortamlara ve kuruluş yönergelerine ilişkin ek bilgi için GitHub' da [ibm-licensing-operator](#) (ibm-lisanslama-işlecini) sayfasında bulabilirsiniz.

IBM License Service , IBM MQ taşıyıcısının konuşlandırıldığı Kubernetes kümesine kurulur ve kullanımı izlemek için pod ek açıklamaları kullanılır. Bu nedenle, istemcilerin, kapsülü IBM License Service ' in kullandığı belirli ek açıklamalarla konuşlandırmaları gerekir. Taşıyıcı içinde devreye alınan yetki ve yeteneklerinize dayalı olarak, aşağıdaki ek açıklamalardan birini ya da birkaçını kullanın:

- [“IBM MQ Gelişmiş taşıyıcı” sayfa 13](#)
- [“IBM MQ Advanced high Availability Replicator taşıyıcısı” sayfa 14](#)
- [“IBM MQ Temel taşıyıcısı” sayfa 14](#)
- [“IBM MQ Base High Availability Replica taşıyıcısı” sayfa 14](#)
- [“IBM MQ Advanced for Developers taşıyıcısı” sayfa 14](#)
- [“CP4I yetkisi \(Üretim\) olan IBM MQ Advanced kapsayıcısı” sayfa 14](#)
- [“IBM MQ Advanced High Availability Replica container with a CP4I entitlement \(Production\)” sayfa 14](#)
- [“CP4I yetkisi \(Üretim Dışı\) olan IBM MQ Advanced container \(Gelişmiş taşıyıcı\)” sayfa 14](#)
- [“CP4I \(Non-Production\) yetkisine sahip IBM MQ Advanced high Availability Replica \(Gelişmiş\)” sayfa 15](#)
- [“CP4I yetkisine sahip IBM MQ Base \(Üretim\)” sayfa 15](#)
- [“IBM MQ Temel Yüksek Kullanılabilirlik Eşlemesi CP4I yetkisi \(Üretim\)” sayfa 15](#)
- [“IBM MQ Base with a CP4I entitlement \(Non-Production\)” sayfa 15](#)
- [“IBM MQ Temel Yüksek Kullanılabilirlik Eşlemesi CP4I yetkisi \(Üretim Dışı\)” sayfa 15](#)

IBM MQ Gelişmiş taşıyıcı

```
productName: "IBM MQ Advanced"
productID: "208423bb063c43288328b1d788745b0c"
productChargedContainers: "All" | "NAME_OF_CONTAINER"
productMetric: "PROCESSOR_VALUE_UNIT" | "VIRTUAL_PROCESSOR_CORE"
```

IBM MQ Advanced high Availability Replicator taşıyıcısı

```
productName: "IBM MQ Advanced High Availability Replica"  
productID: "546cb719714942c18748137ddd8d5659"  
productChargedContainers: "All" | "NAME_OF_CONTAINER"  
productMetric: "PROCESSOR_VALUE_UNIT" | "VIRTUAL_PROCESSOR_CORE"
```

IBM MQ Temel taşıyıcısı

```
productName: "IBM MQ"  
productID: "c661609261d5471fb4ff8970a36bccea"  
productChargedContainers: "All" | "NAME_OF_CONTAINER"  
productMetric: "PROCESSOR_VALUE_UNIT" | "VIRTUAL_PROCESSOR_CORE"
```

IBM MQ Base High Availability Replica taşıyıcısı

```
productName: "IBM MQ High Availability Replica"  
productID: "2a2a8e0511c849969d2f286670ea125e"  
productChargedContainers: "All" | "NAME_OF_CONTAINER"  
productMetric: "PROCESSOR_VALUE_UNIT" | "VIRTUAL_PROCESSOR_CORE"
```

IBM MQ Advanced for Developers taşıyıcısı

```
productName: "IBM MQ Advanced for Developers"  
productID: "2f886a3eefbe4ccb89b2adb97c78b9cb"  
productChargedContainers: "All" | "NAME_OF_CONTAINER"  
productMetric: "FREE"
```

CP4I yetkisi (Üretim) olan IBM MQ Advanced kapsayıcısı

```
productName: "IBM MQ Advanced with CP4I License"  
productID: "208423bb063c43288328b1d788745b0c"  
productChargedContainers: "All" | "NAME_OF_CONTAINER"  
productMetric: "VIRTUAL_PROCESSOR_CORE"  
productCloudpakRatio: "2:1"  
cloudpakName: "IBM Cloud Pak for Integration"  
cloudpakId: "c8b82d189e7545f0892db9ef2731b90d"
```

IBM MQ Advanced High Availability Replica container with a CP4I entitlement (Production)

```
productName: "IBM MQ Advanced High Availability Replica with CP4I License"  
productID: "546cb719714942c18748137ddd8d5659"  
productChargedContainers: "All" | "NAME_OF_CONTAINER"  
productMetric: "VIRTUAL_PROCESSOR_CORE"  
productCloudpakRatio: "10:1"  
cloudpakName: "IBM Cloud Pak for Integration"  
cloudpakId: "c8b82d189e7545f0892db9ef2731b90d"
```

CP4I yetkisi (Üretim Dışı) olan IBM MQ Advanced container (Gelişmiş taşıyıcı)

```
productName: "IBM MQ Advanced for Non-Production with CP4I License"  
productID: "21dfe9a0f00f444f888756d835334909"  
productChargedContainers: "All" | "NAME_OF_CONTAINER"  
productMetric: "VIRTUAL_PROCESSOR_CORE"  
productCloudpakRatio: "4:1"  
cloudpakName: "IBM Cloud Pak for Integration"  
cloudpakId: "c8b82d189e7545f0892db9ef2731b90d"
```

CP4I (Non-Production) yetkisine sahip IBM MQ Advanced high Availability Replica (Gelişmiş)

```
productName: "IBM MQ Advanced High Availability Replica for Non-Production with CP4I License"
productID: "b3f8f984007d47fb981221589cc50081"
productChargedContainers: "All" | "NAME_OF_CONTAINER"
productMetric: "VIRTUAL_PROCESSOR_CORE"
productCloudpakRatio: "20:1"
cloudpakName: "IBM Cloud Pak for Integration"
cloudpakId: "c8b82d189e7545f0892db9ef2731b90d"
```

CP4I yetkisine sahip IBM MQ Base (Üretim)

```
productName: "IBM MQ with CP4I License"
productID: "c661609261d5471fb4ff8970a36bceca"
productChargedContainers: "All" | "NAME_OF_CONTAINER"
productMetric: "VIRTUAL_PROCESSOR_CORE"
productCloudpakRatio: "4:1"
cloudpakName: "IBM Cloud Pak for Integration"
cloudpakId: "c8b82d189e7545f0892db9ef2731b90d"
```

IBM MQ Temel Yüksek Kullanılabilirlik Eşlemesi CP4I yetkisi (Üretim)

```
productName: "IBM MQ High Availability Replica with CP4I License"
productID: "2a2a8e0511c849969d2f286670ea125e"
productChargedContainers: "All" | "NAME_OF_CONTAINER"
productMetric: "VIRTUAL_PROCESSOR_CORE"
productCloudpakRatio: "20:1"
cloudpakName: "IBM Cloud Pak for Integration"
cloudpakId: "c8b82d189e7545f0892db9ef2731b90d"
```

IBM MQ Base with a CP4I entitlement (Non-Production)

```
productName: "IBM MQ with CP4I License Non-Production"
productID: "151bec68564a4a47a14e6fa99266deff"
productChargedContainers: "All" | "NAME_OF_CONTAINER"
productMetric: "VIRTUAL_PROCESSOR_CORE"
productCloudpakRatio: "8:1"
cloudpakName: "IBM Cloud Pak for Integration"
cloudpakId: "c8b82d189e7545f0892db9ef2731b90d"
```

IBM MQ Temel Yüksek Kullanılabilirlik Eşlemesi CP4I yetkisi (Üretim Dışı)

```
productName: "IBM MQ High Availability Replica with CP4I License Non-Production"
productID: "f5d0e21c013c4d4b8b9b2ce701f31928"
productChargedContainers: "All" | "NAME_OF_CONTAINER"
productMetric: "VIRTUAL_PROCESSOR_CORE"
productCloudpakRatio: "40:1"
cloudpakName: "IBM Cloud Pak for Integration"
cloudpakId: "c8b82d189e7545f0892db9ef2731b90d"
```

OpenShift

CP4I

Kubernetes

Taşıyıcılarda IBM MQ için yüksek kullanılabilirlik

IBM MQ Operator ile yüksek kullanılabilirlik için üç seçenek vardır: **Yerel HA kuyruk yöneticisi** (etkin bir eşlemeyi ve iki yedek eşlemeyi içerir), **Çok eşgörünümlü kuyruk yöneticisi** (paylaşılan, ağ üzerinden çalışan bir dosya sistemi kullanan etkin bir yedekte bekleme çiftidir) ya da **Tek bir esnek kuyruk yöneticisi** (ağ üzerinde çalışan depolamayı kullanan HA için basit bir yaklaşım sunar). İkinci olarak, kurtarılabılır verilerin kullanılabilirliğini sağlamak için dosya sistemine güvenir, ancak Yerel HA 'nın kullanılmaması gerekir. Bu nedenle, Yerel HA 'yı kullanmadığınızda, dosya sisteminin kullanılabilirliği

kuyruk yöneticisi kullanılabilirliği açısından kritik önem içerir. Veri kurtarmanın önemli olduğu durumlarda, dosya sisteminin eşleme yoluyla yedeklilik sağlanması gerekir.

Ayrıca, ayrı olarak **ileti** ve **hizmet** kullanılabilirliğini göz önünde bulundurmanız gerekir. IBM MQ for Multiplatforms ile, bir ileti tam olarak bir kuyruk yöneticisiyle saklanır. Bu nedenle, kuyruk yöneticisi kullanılamaz duruma gelirse, geçici olarak tuttuğu iletilere erişimi geçici olarak kaybedersiniz. Yüksek ileti kullanılabilirliğine ulaşmak için, bir kuyruk yöneticisini mümkün olan en kısa sürede geri yükleyebilmeniz gerekir. You can achieve hizmet availability by having multiple instances of queues for client applications to use, for example by using an IBM MQ uniform cluster.

Kuyruk yöneticisi iki kısımda düşünülebilmektedir: diskte saklanan veriler ve verilere erişilmesine izin veren çalışan işlemler. Herhangi bir kuyruk yöneticisi farklı bir Kubernetes düğümüne taşınabilir (Kubernetes Kalıcı Birimler tarafından sağlanan) aynı verileri alıkoymasını ve istemci uygulamaları tarafından ağ üzerinde hala adreslenebilir olması koşuluyla. Kubernetes' ta, tutarlı bir ağ kimliği sağlamak için bir Hizmet kullanılır.

IBM MQ , kalıcı birimlerdeki verilerin kullanılabilirliğine dayanır. Bu nedenle, IBM MQ , kullanmakta olduğu depolama alanından daha fazla kullanılabilir olamayacağı için, kalıcı birimleri sağlayan depolamanın kullanılabilirliğini kuyruk yöneticisi kullanılabilirliği açısından kritik öneme sahip olur. Bir kullanılabilirlik bölgesinin kesintisine göz yummak istiyorsanız, disk yazmasını başka bir bölgeye kopyalayan bir birim sağlayıcısı kullanmanız gerekir.

Yerel HA kuyruk yöneticisi

CP4I V 9.2.3

Yerel HA kuyruk yöneticileri IBM Cloud Pak for Integration 2021.2.1' den, IBM MQ Operator 1.6 ya da sonraki bir sürümünü kullanarak IBM MQ 9.2.3 ya da sonraki bir sürümünü kullanır.

Yerel HA kuyruk yöneticileri, her biri kendi Kubernetes Kalıcı Birim kümesiyle tam olarak üç eşleme içeren bir Kubernetes StatefulSet ' in bir parçası olarak çalışan bir **etkin** ve iki **eşleme** Kubernetes Pod içerir. Paylaşılan dosya sistemlerine ilişkin IBM MQ gereksinimleri, yerel bir HA kuyruk yöneticisi (kira temelli kilitleme dışında) kullanılırken de geçerli olur, ancak paylaşılan bir dosya sistemi kullanmanız gerekmez. Üst üste uygun bir dosya sistemi ile blok depolamayı kullanabilirsiniz. Örneğin, *xfs* ya da *ext4*. Yerel bir HA kuyruk yöneticisine ilişkin kurtarma süreleri aşağıdaki etmenlerle denetlenir:

1. Eşleme eşgörünümlerinin etkin yönetim ortamının başarısız olduğunu saptamak için ne kadar süre geçeceğini belirler. Bu yapılandırılabilir.
2. Hazır taşıyıcının, ağ trafiğini değiştirdiğini ve yeniden yönlendirdiğini algılaması için Kubernetes Pod Readiness araştırması ne kadar sürer? Bu yapılandırılabilir.
3. IBM MQ istemcilerinin yeniden bağlanmasını ne kadar süre alır.

Daha fazla bilgi için bkz. [“Yerel HA” sayfa 91](#)

Çok eşgörünümlü kuyruk yöneticisi

Multi

Çok eşgörünümlü kuyruk yöneticileri, tam olarak iki eşleme ve bir Kubernetes Kalıcı Birim kümesi ile Kubernetes Stateful Set 'in bir parçası olarak çalışan bir **etkin** ve bir **beklemedeki** Kubernetes Pod 'unu içerir. Kuyruk yöneticisi hareket günlükleri ve verileri, paylaşılan bir dosya sistemi kullanılarak iki kalıcı birimde tutulur.

Çok eşgörünümlü kuyruk yöneticileri hem **active** , hem de **standby** Pod 'larının kalıcı birime eşzamanlı erişime sahip olmasını gerektirir. To configure this, you use Kubernetes Persistent Volumes with **access mode** set to ReadWritePek çok. IBM MQ , dosya kilitlerinin bir kuyruk yöneticisini hata durumunda yedek sisteme geçişe teşvik etmek için otomatik olarak serbest bırakılmasına dayandığından, birimlerin IBM MQ paylaşılan kütük sistemlerine ilişkin gereksinimleri karşılaması gerekir. IBM MQ , bir test edilen dosya sistemlerinin listesi oluşturur.

Çok eşgörünümlü bir kuyruk yöneticisine ilişkin kurtarma süreleri aşağıdaki etkenler tarafından denetlenir:

1. Paylaşılan kütük sisteminin, etkin yönetim ortamı tarafından ilk olarak alınan kilitleme serbest bırakması için bir hata oluşuktan sonra ne kadar zaman alır.
2. Yedek yönetim ortamının kilitleme elde etmesi ve daha sonra başlatılması ne kadar sürer?
3. Hazır taşıyıcının, ağ trafiğini değiştirdiğini ve yeniden yönlendirdiğini algılaması için Kubernetes Pod Readiness araştırması ne kadar sürer? Bu yapılandırılabilir.
4. IBM MQ istemcilerinin yeniden bağlanması ne kadar sürer?

Tek bir esnek kuyruk yöneticisi

Multi

Tek bir esnek kuyruk yöneticisi, tek bir Kubernetes Pod 'unda çalışan bir kuyruk yöneticisinin tek bir eşgörünümüdür; burada Kubernetes kuyruk yöneticisini izler ve gerektiği şekilde Pod 'un yerini alır.

IBM MQ [Paylaşılan dosya sistemlerine ilişkin gereksinimler](#) , tek bir esnek kuyruk yöneticisi (kira temelli kilitleme dışında) kullanırken de uygulanır, ancak paylaşılan bir dosya sistemi kullanmanız gerekmez. Üst üste uygun bir dosya sistemi ile blok depolamayı kullanabilirsiniz. Örneğin, *xfs* ya da *ext4*.

Tek bir esnek kuyruk yöneticisine ilişkin kurtarma süreleri aşağıdaki etkenler tarafından denetlenir:

1. Hızlılık sondasının çalışması ne kadar zaman alır ve kaç tane başarısızlığa tahammül eder. Bu yapılandırılabilir.
2. Kubernetes Zamanlayıcısı, başarısız olan Pod 'ı yeniden yeni bir düğüme yeniden zamanlamak için ne kadar zaman alır.
3. Taşıyıcı görüntüsünü yeni düğüme karşıdan yüklemek ne kadar sürerse. `IfNotPresent imagePullPolicy` değerini kullanıyorsanız, görüntü o düğümde önceden kullanılabilir olabilir.
4. Yeni kuyruk yöneticisi yönetim ortamının başlatılması ne kadar sürer?
5. How long it takes for the Kubernetes Pod readiness probe to detect that the container is ready. Bu yapılandırılabilir.
6. IBM MQ istemcilerinin yeniden bağlanması ne kadar sürer?

Önemli:

Tek bir esnek kuyruk yöneticisi örüntünün bazı yararlar sunmasına rağmen, düğüm hataları etrafındaki sınırlamalarla, uygunluk hedeflerinize erişip erişemediğinizi anlamamız gerekir.

Kubernetes' ta, arızalı bir Pod genellikle hızlı bir şekilde kurtarılır; ancak, tüm düğümün başarısız olması farklı bir şekilde işlenir. Kubernetes StatefulSet ile IBM MQ gibi durumlu bir iş yükü kullanırken, bir Kubernetes Ana Düğümü bir işçi düğümüyle iletişim kuruyorsa, düğümün başarısız olup olmadığını ya da ağ bağlantılılığını kaybetmişse bu düğümün başarısız olup olmadığını saptayamaz. Bu nedenle, Kubernetes , aşağıdaki olaylardan biri gerçekleşinceye kadar bu durumda **işlem yok** seçeneğini alır:

1. Düğüm, Kubernetes Ana Düğümü ile iletişim kurabileceği bir duruma kurtarır.
2. An administrative action is taken to explicitly delete the Pod on the Kubernetes Master Node. Bu, Pod 'un çalışmasını durdurması gerekmez, ancak yalnızca Kubernetes mağazasından silinir. Bu nedenle, bu yönetimle ilgili işlem çok dikkatli bir şekilde alınmalıdır.

İlgili görevler

[“IBM MQ Operator kullanarak kuyruk yöneticileri için yüksek kullanılabilirliğin yapılandırılması” sayfa 91](#)

İlgili başvurular

[Yüksek kullanılabilirlik yapılandırmaları](#)

OpenShift CP4I Kubernetes Taşıyıcılarda IBM MQ için olağanüstü durumdan kurtarma

Ne tür bir felakete hazırlandığını göz önünde bulunmanız gerekir. Bulut ortamlarında, kullanılabilirlik bölgelerinin kullanımı, olağanüstü durumlar için belirli bir tolerans düzeyi sağlar ve kullanımı çok daha kolaydır. Çok sayıda veri merkeziniz (çekirdek için) ve düşük gecikme süreli ağ bağlantısı varsa, her biri ayrı

bir fiziksel konumda bulunan birden çok kullanılabilir bölgeye sahip tek bir Red Hat OpenShift Container Platform ya da Kubernetes kümesi çalıştırabilirsiniz. Bu konuda, bu kriterlerin karşılanmadığı olağanüstü durumdan kurtarma için dikkat edilmesi gereken noktalar ele alınmıştır: bu, veri merkezlerinin sayısı ya da yüksek gecikme süresi ağ bağlantısı.

Olağanüstü durumdan kurtarma için aşağıdakileri göz önünde bulundurmanız gerekir:

- Replication of IBM MQ data (held in one or more PersistentVolume resources) to the disaster recovery location
- Eşlenmiş verileri kullanarak kuyruk yöneticisinin yeniden yaratılması
- IBM MQ istemci uygulamaları ve diğer kuyruk yöneticileri tarafından görülebilen kuyruk yöneticisi ağ tanıtıcısı. Bu tanıtıcı, örneğin bir DNS girdisi olabilir.

Olağanüstü durumdan kurtarma yerine zamanuyumlu ya da zamanuyumsuz olarak kalıcı verilerin eşlenmesi gerekir. Bu, genellikle depolama sağlayıcıya özgüdür, ancak bir VolumeSnapshot kullanılarak da yapılabilir. Birim anlık görüntülerine ilişkin ek bilgi için [CSI birim anlık görüntüleri](#) başlıklı konuya bakın.

Bir olağanüstü durumdan kurtarma işlemi sırasında, eşlenen verileri kullanarak yeni Kubernetes kümesinde kuyruk yöneticisi yönetim ortamını yeniden yaratmanız gerekir. If you are using the IBM MQ Operator, you will need the QueueManager YAML, as well as the YAML for other supporting resources like a ConfigMap or Secret.

İlgili bilgiler


[ha_for_ctr.dita](#)

Taşıyıcılarda IBM MQ için kullanıcı kimlik doğrulaması ve yetkilendirmesi

IBM MQ , LDAP kullanıcılarını ve gruplarını kullanacak şekilde yapılandırılabilir. Diğer bir seçenek olarak, yerel işletim sistemi kullanıcılarını ve gruplarını kapsayıcı görüntüsü içinde de kullanabilirsiniz. IBM MQ Operator , güvenlik endişeleri nedeniyle işletim sistemi kullanıcıları ve grupları kullanıcılarına izin vermez.

Çok kiracılı bir kapsayıcı ortamda, olası güvenlik sorunlarını önlemek için genellikle güvenlik kısıtlamaları koyulur; örneğin:

- **Bir taşıyıcının "root" (root) kullanıcılarını kullanmayı önleme**
- **Rasgele UID kullanımını zorlama.** For example, in Red Hat OpenShift Container Platform the default SecurityContextConstraints (called restricted) uses a randomized user ID for each container.
- **Ayrıcalık yükseltme kullanımının önlenmesi.** Linux üzerinde IBM MQ , kullanıcıların parolalarını denetlemek için ayrıcalık yükseltme özelliğini kullanıyor; bunu yapmak için "root" kullanıcısı olmak üzere bir "setuid" programı kullanıyor.

 Bu güvenlik önlemlerine uygunluğu sağlamak için, IBM MQ Operator , bir taşıyıcının içinde bulunan işletim sistemi kitaplıklarında tanımlı olan kimliklerin kullanılmasına izin vermez. Taşıyıcıda tanımlı bir mqm kullanıcı kimliği ya da grubu yok. IBM Cloud Pak for Integration ve Red Hat OpenShift'te IBM MQ kullanırken, kuyruk yöneticinizi kullanıcı kimlik doğrulaması ve yetkilendirmesi için LDAP kullanacak şekilde yapılandırmanız gerekir. Bunu yapmak üzere IBM MQ ' in yapılandırılmasına ilişkin bilgi için bkz. [Bağlantı kimlik doğrulaması: Kullanıcı havuzları](#) ve [LDAP yetkilendirmesi](#)

Multi Kapsayıcılarda IBM MQ için ölçeklenebilirliğin ve performansın planlanması

Çoğu durumda, kapsayıcılarda IBM MQ ' in ölçeklenmesi ve performansı, IBM MQ for Multiplatforms ile aynıdır. Ancak, konteyner platformu tarafından uygulanabilecek birkaç ek sınır vardır.

Bu görev hakkında

Kapsayıcılarda IBM MQ için ölçeklenebilirlik ve performans planlarken aşağıdaki seçenekleri göz önünde bulundurun:

Yordam

- **İş parçacığı ve işlem sayısını sınırla**

IBM MQ , koştuzamanlılığı yönetmek için iş parçacıklarını kullanır. Linux' de iş parçacıkları işlem olarak gerçekleştirilir; bu nedenle, kapsayıcı platformu ya da işletim sistemi tarafından işlem sayısı üst sınırına uygulanan sınırlarla karşılaşabilirsiniz. Red Hat OpenShift Container Platform'ında, kapsayıcı başına varsayılan 4096 işlem sınırı vardır (OpenShift 4.11 tarihine kadar 1024 işlem). Bu, senaryoların büyük çoğunluğu için yeterli olsa da, bunun bir kuyruk yöneticisine ilişkin istemci bağlantısı sayısını etkileyeceği durumlar olabilir.

Kubernetes içindeki işlem sınırı, **podPidsLimit** kubelet yapılandırma ayarı kullanılarak bir küme yöneticisi tarafından yapılandırılabilir. Kubernetes belgesinde [Süreç Tanıtıcısı sınırları ve ayırmaları](#) konusuna bakın. Red Hat OpenShift Container Platform'ında [CRI-O deęiřtirgelerini düzenlemek için ContainerRuntimeConfig](#) özel kaynak yaratda yapabilirsiniz.

IBM MQ yapılanışınızda, bir kuyruk yöneticisi için istemci bağlantısı sayısı üst sınırını da ayarlayabilirsiniz. Tek bir sunucu bağlantısı kanalına sınırlar uygulamak için [Sunucu bağlantısı kanal sınırları](#) ve tüm kuyruk yöneticisine sınırlar uygulamak için [MAXCHANNEL INI öznitelięi](#) konusuna bakın.

- **Birim sayısını sınırla.**

Bulut ve kapsayıcı sistemlerinde, aęa baęlı depolama birimleri yaygın olarak kullanılır. Linux Döęümlerine baęlanabilecek birim sayısı sınırlamaları vardır. Örneęin, AWS EC2 , VM başına en çok 30 birimsınırlar. Red Hat OpenShift Container Platform [benzer bir sınıra sahip](#), Microsoft Azure ve Google Cloud Platform gibi.

Yerel HA kuyruk yöneticisi, üç yönetim ortamının her biri için bir birim gerektirir ve yönetim ortamlarının Döęümler arasında daęıtılmasını zorunlu kılar. Ancak, kuyruk yöneticisini eşgörünüm başına üç birim (kuyruk yöneticisi verileri, kurtarma günlükleri ve kalıcı veriler) kullanacak şekilde yapılandırabilirsiniz.

- **IBM MQ ölçekleme tekniklerini kullanın.**

Az sayıda büyük kuyruk yöneticisi yerine, aynı yapılandırmaya sahip birden çok kuyruk yöneticilerini çalıştırmak için IBM MQ gibi tek tip kümeler gibi IBM MQ ölçekleme tekniklerini kullanmak yararlı olabilir. Bu, tek bir kapsayıcının yeniden başlatılmasının (örneğin, kapsayıcı platform bakımının bir parçası olarak) etkisinin azaltılmasının sağladığı ek bir avantaja sahiptir.

OpenShift CP4I CD EUS IBM Cloud Pak for Integration ve Red Hat OpenShift'te IBM MQ kullanılması

IBM MQ Operator , IBM Cloud Pak for Integration'ın bir parçası olarak ya da Red Hat OpenShift Container Platform' da baęımsız olarak IBM MQ ' u devreye alır ve yönetir.

Yordam

- [“IBM MQ Operator için yayın geęmiři” sayfa 19.](#)
- [“IBM MQ 'un IBM Cloud Pak for Integration' e geęirilmesi” sayfa 36.](#)
- [“Red Hat OpenShift üzerinde IBM MQ Operator ürününün kurulması ve kaldırılması” sayfa 58.](#)
- [“IBM MQ Operator ve kuyruk yöneticilerinin yükseltilmesi” sayfa 70.](#)
- [“Deploying and configuring queue managers using the IBM MQ Operator” sayfa 78.](#)
- [“Operating IBM MQ using the IBM MQ Operator” sayfa 113.](#)
- [“IBM MQ Operator için API başvurusu” sayfa 124.](#)

OpenShift CP4I CD EUS IBM MQ Operator için yayın geęmiři

IBM MQ Operator

IBM MQ Operator 1.8.2

CD

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2021.4.1

İşletmen kanalı

v1.8

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.0-r3, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.2-r2-eus, 9.2.0.5-r1-eus, 9.2.0.5-r2-eus, [9.2.0.5-r3-eus](#), 9.2.1.0-r1, 9.2.1.0-r2, 9.2.2.0-r1, 9.2.3.0-r1, 9.2.4.0-r1, 9.2.5.0-r1, 9.2.5.0-r2, [9.2.5.0-r3](#)

Red Hat OpenShift Container Platform sürümler

Red Hat OpenShift Container Platform 4.6 ve üstü

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.8 ve üstü (v3 kanalı)

Yenilikler

- Yalnızca [IBM MQ İşleci 1.8.0](#) üzerinde oluşturulan güvenlik güncelleştirmesi
- Ele alınan güvenlik açıkları bu [Security Bulletin](#)inde ayrıntılı olarak açıklanmıştır.

IBM MQ Operator 1.8.1

CD

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2021.4.1

İşletmen kanalı

v1.8

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.0-r3, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.2-r2-eus, 9.2.1.0-r1, 9.2.1.0-r2, 9.2.2.0-r1, 9.2.3.0-r1, 9.2.4.0-r1, 9.2.5.0-r1, [9.2.5.0-r2](#)

Red Hat OpenShift Container Platform sürümler

Red Hat OpenShift Container Platform 4.6 ve üstü

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.8 ve üstü (v3 kanalı)

Yenilikler

- Yalnızca [IBM MQ İşleci 1.8.0](#) üzerinde oluşturulan güvenlik güncelleştirmesi
- Ele alınan güvenlik açıkları bu [Security Bulletin](#)inde ayrıntılı olarak açıklanmıştır.

IBM MQ Operator 1.8.0

CD

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2021.4.1

İşletmen kanalı

v1.8

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.0-r3, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.2-r2-eus, 9.2.1.0-r1, 9.2.1.0-r2, 9.2.2.0-r1, 9.2.3.0-r1, 9.2.4.0-r1, [9.2.5.0-r1](#)

Red Hat OpenShift Container Platform sürümler

Red Hat OpenShift Container Platform 4.6 ve üstü

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.8 ve üstü (v3 kanalı)

Yenilikler

- Kullanımdan kaldırılan IBM MQ sürümleri için durum koşulları ekler.

Değişiklikler

- Görüntüler, Docker Hub 'dan IBM Container Registry' e taşındı.
 - Güvenlik duvarı kuralları olan müşterilerin, IBM Container Registry üzerindeki görüntülere erişebilmeleri için bunları ayarlamaya gereksinim duyabilir.
 - Hava boşluğu müşterileri, IBM MQ Operator 1.8.0 sürümüne yükselirken bir düğüm yeniden başlatma deneyimi yaşatır.
- Kullanımdan kaldırılan sürümler: IBM MQ 9.1.5, 9.2.0 CD, 9.2.1, 9.2.2. These versions might not be reconciled by future versions of the IBM MQ Operator.
- Changes to license logic: Customers upgrading to IBM MQ 9.2.5 may use only the licenses specified to work with IBM MQ 9.2.5. Bkz. [mq.ibm.com/v1beta1 için lisans başvurusu](https://mq.ibm.com/v1beta1/ibm-licensing-requirements) sayfa 124.
- Ele alınan güvenlik açıkları bu [Security Bulletin](#) içinde ayrıntılı olarak açıklanmıştır.

IBM MQ Operator 1.7.0



IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2021.4.1

İşletmen kanalı

v1.7

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.0-r3, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.2-r2-eus, 9.2.1.0-r1, 9.2.1.0-r2, 9.2.2.0-r1, 9.2.3.0-r1, 9.2.4.0-r1

Red Hat OpenShift Container Platform sürümler

Red Hat OpenShift Container Platform 4.6 ve üstü

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.8 ve üstü (v3 kanalı)

Yenilikler

- Adds IBM MQ 9.2.4 as a continuous delivery release

IBM MQ Operator 1.6.0



IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2021.2.1

İşletmen kanalı

v1.6

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.0-r3, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.2-r2-eus, 9.2.1.0-r1, 9.2.1.0-r2, 9.2.2.0-r1, 9.2.3.0-r1

Red Hat OpenShift Container Platform sürümler

Red Hat OpenShift Container Platform 4.6 ve üstü

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.7 ve üstü (v3 kanalı)

Yenilikler

- Adds IBM MQ 9.2.3 as a continuous delivery release (amd64 only for IBM Cloud Pak for Integration 2021.2.1; amd64 or s390x when using an IBM MQ license)
- Kuyruk yöneticileri için yeni kullanılabilirlik tipi: [Yerel HA](#). Available for production use, as part of IBM Cloud Pak for Integration 2021.2.1.

Değişiklikler

- IBM MQ Operator 1.6 ve daha yüksek bir sürümü, Docker Hub yerine IBM Container Registry ' yı kullanır. Bu, `icr.io` ' dan bir `catalogSource` kullanmanız gerektiği anlamına gelir. Bkz. "[Red Hat OpenShift üzerinde IBM MQ Operator ürününün kurulması ve kaldırılması](#)" sayfa 58.
- Yerel HA kayan güncelleme, bir eşlemenin bir sonraki eşlemeye geçmeden önce eşitlenmesi için beklemesini beklemez.
- OCP 4.7 ve üstü için Yerel HA benzeşimiyle ilgili düzeltmeler sorunu.
- Yerel HA ' ya sahip CA imzalı sertifikalar kullanılırken düzeltmeler sorunu.

IBM MQ Operator 1.5.0



IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2021.1.1

İşletmen kanalı

v1.5

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, [9.2.0.0-r3](#), 9.2.0.1-r1-eus, 9.2.1.0-r1, [9.2.1.0-r2](#), [9.2.2.0-r1](#)

Red Hat OpenShift Container Platform sürümler

Red Hat OpenShift Container Platform 4.6 ve üstü

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.7 ve üstü (v3 kanalı)

Yenilikler

- Adds IBM MQ 9.2.2 as a continuous delivery release (amd64 only for IBM Cloud Pak for Integration 2021.1.1; amd64 or s390x when using an IBM MQ license)
- Kuyruk yöneticileri için yeni kullanılabilirlik tipi: [Yerel HA](#). Yalnızca, IBM Cloud Pak for Integration 2021.1.1' in bir parçası olarak yalnızca değerlendirme amacıyla kullanılabilir.
- Bir `ServiceMonitor` kaynağı sağlayarak, Prometheus ölçümlerine ilişkin Red Hat OpenShift Container Platform Cluster Monitoring ile bütünleştirme

Değişiklikler

- The IBM Licensing Operator is no longer created by default when you create a queue manager
- Çok eşgörünümlü kuyruk yöneticilerine yapılan güncellemeler, şu anda bir hareketli siparişte ele alınır. Bu değişikliğin bir parçası olarak, hızlılık sondası yapılandırılırken kullanılan değerleri etkileyen bir Kubernetes startup probe olanağı kullanıma sunulmuştur. Başlatma birimi hemen başlatılır, sonra kuyruk yöneticisinin başarıyla başlatılması için bekler. Başlatma sondası, bu bekleme süresi içinde herhangi bir zamanda geçerse, hızlılık ve hazır olma olasılıkları başlar. Daha önce, başlamak için yavaş olan bir kuyruk yöneticiniz varsa, hızlılık sondası üzerindeki `initialDelaySeconds` ayarını artırmış olabilirsiniz. Bunu yaptıysanız, `initialDelaySeconds` ' u daha önceki bir ayara geri döndürmelisiniz.
- `CustomResourceDefinition`, `apiextensions.k8s.io/v1beta1` kaynağından `apiextensions.k8s.io/v1` sürümüne yükseltiliyor.

Bilinen sorunlar ve sınırlamalar

- Identity and Access Management (IAM) bileşeninde uyumsuz bir değişiklik içeren IBM Cloud Pak foundational services 3.7' yi gerektirir. IBM Cloud Pak for Integration lisansı kullanan kuyruk

yöneticileriniz varsa, bu büyütmeden sonra web konsoluna erişmek için kuyruk yöneticisi yeniden başlatılmalıdır; web konsolunda diğer hatalar oturum açma işlemini de görürsünüz. İşleç büyütme tamamlandıktan sonra, seçtiğiniz IBM MQ sürümünüz için en son `.spec.version` değerine yükselterek bu hataları düzeltebilirsiniz.

- MQ sürümünü yükseltiyorsanız, toplama güncelleştirmesi otomatik olarak başlamaz. Kapsülleri el ile silmeniz gerekir.

IBM MQ Operator 1.4.0

CD

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.4.1 (IBM MQ Operator 1.4.0 , bir CD yayın düzeyidir ve Extended Update Support için uygun değildir)

İşletmen kanalı

v1.4

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.1-r1-eus, [9.2.1.0-r1](#)

Red Hat OpenShift Container Platform sürümler

Red Hat OpenShift Container Platform 4.6 ve üstü

Yenilikler

- Adds IBM MQ 9.2.1 as a continuous delivery release
- Şimdi `.spec.queueManager.route.enabled` ayarını `false` olarak ayarlayarak varsayılan kuyruk yöneticisi Rotası yaratılmasını engelleyebilirsiniz.

Bilinen sorunlar ve sınırlamalar

- When updating a QueueManager with an availability type of MultiInstance, both Pods will be deleted immediately. Her ikisi de Red Hat OpenShift Container Platform tarafından hızlı bir şekilde yeniden başlatılmalıdır.

IBM MQ Operator 1.3.8 (EUS)

EUS

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.4.1

İşletmen kanalı

v1.3-eus

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.4-r1-eus, 9.2.0.5-r1-eus, 9.2.0.5-r2-eus, 9.2.0.5-r3-eus, 9.2.0.6-r1-eus, 9.2.0.6-r2-eus, [9.2.0.6-r3-eus](#)

Red Hat OpenShift Container Platform sürümler

Yalnızca Red Hat OpenShift Container Platform 4.6

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.6 (stable-v1 kanalı)

Yenilikler

- Yeni işlenen sürümü [9.2.0.6-r3-eusekler](#).
- Ele alınan güvenlik açıkları bu [Security Bulletin](#)inde ayrıntılı olarak açıklanmıştır.

IBM MQ Operator 1.3.7 (EUS)

EUS

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.4.1

İşletmen kanalı

v1.3-eus

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.4-r1-eus, 9.2.0.5-r1-eus, 9.2.0.5-r2-eus, 9.2.0.5-r3-eus, 9.2.0.6-r1-eus, [9.2.0.6-r2-eus](#)

Red Hat OpenShift Container Platform sürümler

Yalnızca Red Hat OpenShift Container Platform 4.6

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.6 (stable-v1 kanalı)

Yenilikler

- Yeni işlenen sürümü [9.2.0.6-r2-eusekler](#).
- Ele alınan güvenlik açıkları bu [Security Bulletin](#)inde ayrıntılı olarak açıklanmıştır.

IBM MQ Operator 1.3.6 (EUS)



IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.4.1

İşletmen kanalı

v1.3-eus

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.4-r1-eus, 9.2.0.5-r1-eus, 9.2.0.5-r2-eus, 9.2.0.5-r3-eus, [9.2.0.6-r1-eus](#)

Red Hat OpenShift Container Platform sürümler

Yalnızca Red Hat OpenShift Container Platform 4.6

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.6 (stable-v1 kanalı)

Yenilikler

- Yeni işlenen sürümü [9.2.0.6-r1-eusekler](#).
- Ele alınan güvenlik açıkları bu [Security Bulletin](#)inde ayrıntılı olarak açıklanmıştır.

IBM MQ Operator 1.3.5 (EUS)



IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.4.1

İşletmen kanalı

v1.3-eus

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.4-r1-eus, 9.2.0.5-r1-eus, 9.2.0.5-r2-eus, [9.2.0.5-r3-eus](#)

Red Hat OpenShift Container Platform sürümler

Yalnızca Red Hat OpenShift Container Platform 4.6

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.6 (stable-v1 kanalı)

Yenilikler

- Yeni işlenen sürümü [9.2.0.5-r3-eusekler](#).
- Ele alınan güvenlik açıkları bu [Security Bulletin](#)inde ayrıntılı olarak açıklanmıştır.

IBM MQ Operator 1.3.4 (EUS)

EUS

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.4.1

İşletmen kanalı

v1.3-eus

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.4-r1-eus, 9.2.0.5-r1-eus, 9.2.0.5-r2-eus

Red Hat OpenShift Container Platform sürümler

Yalnızca Red Hat OpenShift Container Platform 4.6

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.6 (stable-v1 kanalı)

Yenilikler

- Yeni işlenen sürümü ekler 9.2.0.5-r2-eus
- Ele alınan güvenlik açıkları bu [Security Bulletin](#)inde ayrıntılı olarak açıklanmıştır.

IBM MQ Operator 1.3.3 (EUS)

EUS

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.4.1

İşletmen kanalı

v1.3-eus

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.4-r1-eus, 9.2.0.5-r1-eus

Red Hat OpenShift Container Platform sürümler

Yalnızca Red Hat OpenShift Container Platform 4.6

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.6 (stable-v1 kanalı)

Yenilikler

- Yeni işlenen sürümü ekler 9.2.0.5-r1-eus
- Ele alınan güvenlik açıkları bu [Security Bulletin](#)inde ayrıntılı olarak açıklanmıştır.

IBM MQ Operator 1.3.2 (EUS)

EUS

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.4.1

İşletmen kanalı

v1.3-eus

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.1-r1-eus, 9.2.0.2-r1-eus, 9.2.0.4-r1-eus

Red Hat OpenShift Container Platform sürümler

Yalnızca Red Hat OpenShift Container Platform 4.6

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.6 (stable-v1 kanalı)

Yenilikler

- Yeni işlenen sürümü ekler 9.2.0.4-r1-eus

IBM MQ Operator 1.3.1 (EUS)

EUS

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.4.1

İşletmen kanalı

v1.3-eus

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.1-r1-eus, [9.2.0.2-r1-eus](#)

Red Hat OpenShift Container Platform sürümler

Yalnızca Red Hat OpenShift Container Platform 4.6

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.6 (stable-v1 kanalı)

Yenilikler

- Yeni işlenen sürümü ekler [9.2.0.2-r1-eus](#)

IBM MQ Operator 1.3.0 (EUS)

EUS

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.4.1

İşletmen kanalı

v1.3-eus

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, [9.2.0.1-r1-eus](#)

Red Hat OpenShift Container Platform sürümler

Yalnızca Red Hat OpenShift Container Platform 4.6

IBM Cloud Pak foundational services sürümler

IBM Cloud Pak foundational services 3.6 (stable-v1 kanalı)

Yenilikler

- Extended Update Support (EUS) is offered for .spec.version fields ending with -eus, when using a IBM Cloud Pak for Integration license
- Adds a new way of setting labels and annotations on the QueueManager resource using .spec.labels and .spec.annotations

Değişiklikler

- Tek eşgörünümde çok eşgörünümde değişiklik yapmaya çalışırken hata işlemeyi iyileştirir
- QueueManager özelliklerinin IBM Cloud Pak for Integration Platform Navigator'de görsel olarak nasıl gerçekleştirileceği ve Red Hat OpenShift Container Platform web konsolunun "Form Görünümü" ne ilişkin geliştirmeler
- IBM Cloud Pak for Integration lisansı kullanırken varsayılan lisans metriği VirtualProcessorCore' u düzeltmeye ilişkin düzeltmeleri
- Fixes the **Kaynaklar** tab for QueueManager in the Red Hat OpenShift Container Platform web console, which now correctly shows the resources managed by the IBM MQ Operator for that queue manager

Bilinen sorunlar ve sınırlamalar

- When updating a QueueManager with an availability type of MultiInstance, both Pods will be deleted immediately. Her ikisi de Red Hat OpenShift Container Platform tarafından hızlı bir şekilde yeniden başlatılmalıdır.

IBM MQ Operator 1.2.0

CD

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.3.1

İşletmen kanalı

v1.2

.spec.version için izin verilen değerler

9.1.5.0-r2, 9.2.0.0-r1, [9.2.0.0-r2](#)

Red Hat OpenShift Container Platform sürümler

Red Hat OpenShift Container Platform 4.4 ve üstü

Yenilikler

- Z/Linux için destek ekler
- QueueManager kaynağına daha ayrıntılı durum koşulları ekler. Daha fazla bilgi için bkz. [“Status conditions for QueueManager \(mq.ibm.com/v1beta1\)” sayfa 143](#)
- Geçersiz depolama sınıflarının kullanımını önlemek için ek yürütme ortamı denetimleri ekler. Daha fazla bilgi için bkz. [“Runtime webhook denetimlerinin geçersiz kılınması” sayfa 113](#)
- Çok eşgörünümlü kuyruk yöneticileri için deneyimi basitleştirir: bu artık QueueManager kaynağındaki tek bir özellik (.spec.queueManager.availability.type) ile seçilebilir.
- QueueManager özelinde .spec.queueManager.storage.defaultClass özelliğini tanıtarak, varsayılan olmayan bir Depolama Sınıfı seçmeyi basitleştirir

Değişiklikler

- QueueManager özelliklerinin IBM Cloud Pak for Integration Platform Navigator'de görsel olarak nasıl gerçekleştirileceği ve Red Hat OpenShift Container Platform web konsolunun "Form Görünümü" 'ne ilişkin geliştirmeler
- Yükseltilmiş bir kuyruk yöneticisi sürümü varsa, bu sürüm IBM Cloud Pak for Integration Platform Navigator' de işaretlenir.

IBM MQ Operator 1.1.0

CD

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.2.1

İşletmen kanalı

v1.1

.spec.version için izin verilen değerler

9.1.5.0-r2, [9.2.0.0-r1](#)

Red Hat OpenShift Container Platform sürümler

Red Hat OpenShift Container Platform 4.4 ve üstü

Yenilikler

- Adds IBM MQ Advanced 9.2.0 as a continuous delivery release
- Bir ConfigMap ya da Secret içinde INI ve MQSC bilgilerini belirtmek için özellik ekler.
- Enables the schema navigator when using the Red Hat OpenShift Container Platform web console

Değişiklikler

- Ağ ilkesiyle ilgili düzeltme sorunu, IBM Cloud üzerinde Red Hat OpenShift etki
- QueueManager kaynaklarındaki ayarların geçersiz birleşimlerini önlemek için, web kancası doğrulanırken yapılan iyileştirmeler

IBM MQ Operator 1.0.0

CD

IBM Cloud Pak for Integration sürüm

IBM Cloud Pak for Integration 2020.2.1

İşletmen kanalı

v1.0

.spec.version için izin verilen değerler

[9.1.5.0-r2](#)

Red Hat OpenShift Container Platform sürümler

Red Hat OpenShift Container Platform 4.4 ve üstü

Yenilikler

- İşletmenin ilk sürümü, mq.ibm.com/v1beta1 API 'si tanıtılıyor

IBM MQ Operator ile kullanılmak üzere kuyruk yöneticisi taşıyıcı görüntüleri

9.2.5.0-r3

CD

Gerekli operatör sürümü

[1.8.2](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.5.0-r3
- cp.icr.io/cp/ibm-mqadvanced-server:9.2.5.0-r3
- icr.io/ibm-messaging/mq:9.2.5.0-r3

Yenilikler

- [IBM MQ 9.2.5' daki yenilikler](#)

Değişiklikler

- [IBM MQ 9.2.5 içinde değişen özellikler](#)
- [Red Hat Universal Base Image 8.6-751](#) temel alınarak

9.2.5.0-r2

CD

Gerekli operatör sürümü

[1.8.1](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.5.0-r2
- cp.icr.io/cp/ibm-mqadvanced-server:9.2.5.0-r2
- icr.io/ibm-messaging/mq:9.2.5.0-r2

Yenilikler

- [IBM MQ 9.2.5' daki yenilikler](#)

Değişiklikler

- [IBM MQ 9.2.5](#) içinde değişen özellikler
- [Red Hat Universal Base Image 8.5-240.1648458092](#) temel alınarak

9.2.5.0-r1



Gerekli operatör sürümü

[1.8.0](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- [cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.5.0-r1](#)
- [cp.icr.io/cp/ibm-mqadvanced-server:9.2.5.0-r1](#)
- [icr.io/ibm-messaging/mq:9.2.5.0-r1](#)

Yenilikler

- [IBM MQ 9.2.5](#)' daki yenilikler

Değişiklikler

- [IBM MQ 9.2.5](#) içinde değişen özellikler
- Geçersiz Uzak kuyruk yöneticileri seçeneği artık IBM MQ Console' den kaldırıldı
- [Red Hat Universal Base Image 8.5-240](#) tabanlı

9.2.4.0-r1



Gerekli operatör sürümü

[1.7.0](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- [cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.4.0-r1](#)
- [cp.icr.io/cp/ibm-mqadvanced-server:9.2.4.0-r1](#)
- [docker.io/ibmcom/mq:9.2.4.0-r1](#)

Yenilikler

- [IBM MQ 9.2.4](#)' daki yenilikler

Değişiklikler

- [IBM MQ 9.2.4](#) içinde değişen özellikler
- [Red Hat Universal Base Image 8.5-204](#) tabanlı

9.2.3.0-r1



Gerekli operatör sürümü

[1.6.0](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.3.0-r1 (yalnızcaamd64)
- cp.icr.io/cp/ibm-mqadvanced-server:9.2.3.0-r1
- docker.io/ibmcom/mq:9.2.3.0-r1

Yenilikler

- [IBM MQ 9.2.3' daki yenilikler](#)
- Support for MQ [Yerel HA](#) for production use, when used with a IBM Cloud Pak for Integration license. Note that queue managers using Native HA under an evaluation license with IBM MQ 9.2.2 cannot be upgraded to 9.2.3. Değerlendirme dönemi sona erdi.

Değişiklikler

- [IBM MQ 9.2.3'ünde değişen özellikler](#)
- [Red Hat Universal Base Image 8.4-205](#)tabanlı

9.2.2.0-r1



Gerekli operatör sürümü

[1.5.0](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.2.0-r1 (yalnızcaamd64)
- cp.icr.io/cp/ibm-mqadvanced-server:9.2.2.0-r1
- docker.io/ibmcom/mq:9.2.2.0-r1

Yenilikler

- [IBM MQ 9.2.2' daki yenilikler](#)
- Bir IBM Cloud Pak for Integration lisansı ile birlikte kullanıldığında, değerlendirme amacıyla MQ [Native HA](#) desteği

Değişiklikler

- [IBM MQ 9.2.2'ünde değişen özellikler](#)
- Bir IBM MQ Advanced for Developers kuyruk yöneticisini sona erdirirken FDC ' ye neden olan sabit sorun
- [Red Hat Universal Base Image 8.3-291](#) temel alınarak

9.2.1.0-r2



Gerekli operatör sürümü

[1.5.0](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.1.0-r2
- cp.icr.io/cp/ibm-mqadvanced-server:9.2.1.0-r2
- docker.io/ibmcom/mq:9.2.1.0-r2

Değişiklikler

- IBM Cloud Pak foundational services 3.7 ve üstü ile tekli oturum açma ile ilgili düzeltme sorunu.
- [Red Hat Universal Base Image 8.3-291](#) temel alınarak

9.2.1.0-r1



Gerekli operatör sürümü

[1.4.0](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- [cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.1.0-r1](#)
- [cp.icr.io/cp/ibm-mqadvanced-server:9.2.1.0-r1](#)
- [docker.io/ibmcom/mq:9.2.1.0-r1](#)

Yenilikler

- [IBM MQ 9.2.1' daki yenilikler](#)
- Varsayılan Rota ile ilgili bağlantı bilgileri MQ web konsolunda kullanılabilir

Değişiklikler

- [IBM MQ 9.2.1 içinde değişen özellikler](#)
- [Red Hat Universal Base Image 8.3-230](#) temel alınarak

9.2.0.6-r3-eus



Gerekli operatör sürümü

[1.3.8](#) ve sonraki düzeltme paketleri

Desteklenen mimariler

amd64, s390x

Resimler

- [cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.6-r3-eus](#)

Değişiklikler

- Includes IBM MQ 9.2.0 Fix Pack 6. Daha fazla bilgi için bkz. [Fix list for IBM MQ Version 9.2 LTS](#).
- Based on [Red Hat Universal Base Image 8.6-941](#).

9.2.0.6-r2-eus



Gerekli operatör sürümü

[1.3.7](#) ve sonraki düzeltme paketleri

Desteklenen mimariler

amd64, s390x

Resimler

- [cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.6-r2-eus](#)

Değişiklikler

- Includes IBM MQ 9.2.0 Fix Pack 6. Daha fazla bilgi için bkz. [Fix list for IBM MQ Version 9.2 LTS](#).

- [Red Hat Universal Base Image 8.6-902](#) temel alınarak bağlantı kuruludur.

9.2.0.6-r1-eus



Gerekli operatör sürümü

[1.3.6](#) ve sonraki düzeltme paketleri

Desteklenen mimariler

amd64, s390x

Resimler

- `cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.6-r1-eus`

Değişiklikler

- Includes IBM MQ 9.2.0 Fix Pack 6. Daha fazla bilgi için bkz. [Fix list for IBM MQ Version 9.2 LTS](#).
- [Red Hat Universal Base Image 8.6-854](#) temel görüntüsüne dayalıdır.

9.2.0.5-r3-eus



Gerekli operatör sürümü

[1.3.5](#) ve sonraki düzeltme paketleri

Desteklenen mimariler

amd64, s390x

Resimler

- `cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.5-r3-eus`

Değişiklikler

- Includes IBM MQ 9.2.0 Fix Pack 5. Daha fazla bilgi için bkz. [IBM MQ 9.2.0 Fix Pack 5 içinde değişen ve IBM MQ Sürüm 9.2 LTS 'ye ilişkin düzeltme listesi](#).
- [Red Hat Universal Base Image 8.6-751.1655117800](#) temel görüntüsüne dayalıdır.

9.2.0.5-r2-eus



Gerekli operatör sürümü

[1.3.4](#) ve sonraki düzeltme paketleri

Desteklenen mimariler

amd64, s390x

Resimler

- `cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.5-r2-eus`

Değişiklikler

- Includes IBM MQ 9.2.0 Fix Pack 5. Daha fazla bilgi için bkz. [IBM MQ 9.2.0 Fix Pack 5 içinde değişen ve IBM MQ Sürüm 9.2 LTS 'ye ilişkin düzeltme listesi](#)
- [Red Hat Universal Base Image 8.6-751](#) temel alınarak

9.2.0.5-r1-eus



Gerekli operatör sürümü

[1.3.3](#) ve sonraki düzeltme paketleri

Desteklenen mimariler

amd64, s390x

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.5-r1-eus

Değişiklikler

- Includes IBM MQ 9.2.0 Fix Pack 5. Daha fazla bilgi için bkz. [IBM MQ 9.2.0 Fix Pack 5 içinde değişen ve IBM MQ Sürüm 9.2 LTS 'ye ilişkin düzeltme listesi](#)
- [Red Hat Universal Base Image 8.5-240.1648458092](#) temel alınarak

9.2.0.4-r1-eus



Gerekli operatör sürümü

[1.3.2](#) ve gelecekteki düzeltme paketleri

Desteklenen mimariler

amd64, s390x

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.4-r1-eus

Değişiklikler

- Includes IBM MQ 9.2.0 Fix Pack 4. Daha fazla bilgi için bkz. [IBM MQ 9.2.0 Fix Pack 4 içinde değişen ve IBM MQ Sürüm 9.2 LTS 'ye ilişkin düzeltme listesi](#)
- [Red Hat Universal Base Image 8.5-204](#) temel alınarak

9.2.0.2-r2-eus



Gerekli operatör sürümü

[1.6.0](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.2-r2-eus

Değişiklikler

- IBM Cloud Pak foundational services 3.7 ve sonraki sürümleriyle tek oturum açma sorunu olan düzeltmeler, yalnızca bir EUS yayınından CD yayınına geçiş sırasında gereklidir.
- [Red Hat Universal Base Image 8.4-200.1622548483](#) temel alınarak

9.2.0.2-r1-eus



Gerekli operatör sürümü

[1.3.1](#) andve gelecekteki düzeltme paketleri ; 1.6.0 ya da sonraki sürümü

Desteklenen mimariler

amd64, s390x

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.2-r1-eus

Değişiklikler

- Operations Dashboard bütünleşirmesi, izleme aracı ve toplayıcı sürüm 1.0.8kullanır
- Includes IBM MQ 9.2.0 Fix Pack 2. Daha fazla bilgi için bkz. [IBM MQ 9.2.0 Fix Pack 2 içinde deęişen ve IBM MQ Sürüm 9.2 LTS ' ye ilişkin düzeltme listesi](#)
- [Red Hat Universal Base Image 8.4-200.1622548483](#)temel alınarak

9.2.0.1-r1-eus

EUS

Gerekli operatör sürümü

[1.3.0](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- [cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.1-r1-eus](#)

Yenilikler

- Yalnızca IBM Cloud Pak for Integration lisansı kullanılırken kullanılabilir
- Extended Update Support (EUS) is available when using IBM MQ Operator 1.3.x and IBM Common Services 3.6, on Red Hat OpenShift Container Platform 4.6

Değişiklikler

- Includes IBM MQ 9.2.0 Fix Pack 1. Daha fazla bilgi için bkz. [IBM MQ 9.2.0 Fix Pack 1 içinde deęişen ve IBM MQ Sürüm 9.2 LTS ' ye ilişkin düzeltme listesi](#)
- [Red Hat Universal Base Image 8.3-201](#)temel alınarak
- Fixes issue with liveness probe (chkmqhealthy) and readiness probe (chkmqready) when running under SecurityContextConstraints which allow privilege escalation.

9.2.0.0-r3

CD

Gerekli operatör sürümü

[1.5.0](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- [cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.0-r3](#)
- [cp.icr.io/cp/ibm-mqadvanced-server:9.2.0.0-r3](#)
- [docker.io/ibmcom/mq:9.2.0.0-r3](#)

Değişiklikler

- [Red Hat Universal Base Image 8.3-291](#)temel alınarak

9.2.0.0-r2

CD

Gerekli operatör sürümü

[1.2.0](#) ya da üstü

Desteklenen mimariler

amd64, s390x

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.0-r2
- cp.icr.io/cp/ibm-mqadvanced-server:9.2.0.0-r2
- docker.io/ibmcom/mq:9.2.0.0-r2

Yenilikler

- Şimdi z/Linux

Değişiklikler

- [Red Hat Universal Base Image 8.2-349](#)temel alınarak

9.2.0.0-r1



Gerekli operatör sürümü

[1.1.0](#) ya da üstü

Desteklenen mimariler

amd64

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.0-r1-amd64
- cp.icr.io/cp/ibm-mqadvanced-server:9.2.0.0-r1-amd64
- docker.io/ibmcom/mq:9.2.0.0-r1

Yenilikler

- [IBM MQ 9.2.0'](#) daki yenilikler

Değişiklikler

- [IBM MQ 9.2.0](#) içinde değişen özellikler
- MQSC dosyalarını otomatik olarak uygulamak için `-ic` bağımsız değişkenini `crtmqm` olarak kullanır. `runmqsc` komutlarının önceki kullanımını değiştirir
- [Red Hat Universal Base Image 8.2-301.1593113563](#) temel alınarak

9.1.5.0-r2



Gerekli operatör sürümü

[1.0.0](#) ya da üstü

Desteklenen mimariler

amd64

Resimler

- cp.icr.io/cp/ibm-mqadvanced-server-integration:9.1.5.0-r2-amd64
- cp.icr.io/cp/ibm-mqadvanced-server:9.1.5.0-r2-amd64
- docker.io/ibmcom/mq:9.1.5.0-r2

Değişiklikler

- [Red Hat Universal Base Image 8.2-267](#) temel alınarak

Integration' e geçirilmesi

This set of topics describes the key steps to migrate an existing IBM MQ queue manager into a container environment using the IBM MQ Operator in IBM Cloud Pak for Integration.

Bu görev hakkında

Red Hat OpenShift üzerinde IBM MQ ' i yerleştiren istemciler aşağıdaki senaryolara ayrılabilir:

1. Yeni uygulamalar için Red Hat OpenShift içinde yeni bir IBM MQ konuşlandırma yaratılması.
2. Extending an IBM MQ network into Red Hat OpenShift for new applications in Red Hat OpenShift.
3. Var olan uygulamaları desteklemeye devam etmek için bir IBM MQ uygulamasını Red Hat OpenShift içine taşıma.

Yalnızca 3 numaralı senaryo için IBM MQ yapılandırmanızı geçirmeniz gerekir. Diğer senaryolar yeni konuşlandırmalar olarak kabul edilir.

Bu konular kümesi 3. senaryoya odaklanır ve var olan bir IBM MQ kuyruk yöneticisini IBM MQ Operatorkullanarak kapsayıcı bir ortama geçirmenin temel adımlarını açıklar. Because of the flexibility and extensive use of IBM MQ, there are several optional steps. Bunların her biri bir "Bunu yapmam gerekiyor" bölümünü içerir. Gereksininiz doğrulanıyor, geçiş sırasında size zaman kazanmanız gerekir.

Hangi verilerin geçirileceğini de göz önünde bulundurmanız gerekir:

1. IBM MQ olanağını, kuyruğa alınmış bir ileti olmadan aynı yapılandırmalarla geçirin.
2. IBM MQ ' u aynı yapılandırma ve var olan iletilerle geçirin.

Sürüm geçişinin tipik bir sürümü her iki yaklaşımı da kullanabilir. Geçiş noktasındaki tipik bir IBM MQ kuyruk yöneticisinde, kuyruklarda saklanan iletiler, birçok vaka için seçenek 1 'i uygun hale getiren birkaç ileti vardır. Bir konteyner platformuna geçiş durumunda, geçişin karmaşıklığını azaltmak ve mavi yeşil bir dağıtıma izin vermek için seçenek 1 'i kullanmak daha da yaygındır. Bu nedenle, yönergelerde bu senaryoya odaklanılıyor.

Bu senaryonun amacı, taşıyıcı ortamında var olan kuyruk yöneticisinin tanımlamasıyla eşleşen bir kuyruk yöneticisi yaratmasıdır. Bu, varolan ağ bağlantılı uygulamaların, başka bir yapılandırma ya da uygulama mantığı değiştirmeden yeni kuyruk yöneticisini işaret edecek şekilde yeniden yapılandırılmasını sağlar.

Bu geçiş boyunca, yeni kuyruk yöneticisine uygulanacak birden çok yapılandırma dosyası oluşturursun. Bu dosyaların yönetimini basitleştirmek için, bir dizin oluşturmalı ve bu dizini bu dizine oluşturmalısınız.

Yordam

1. [“Gerekli işlevlerin kullanılabilir olup olmadığını denetleme” sayfa 37](#)
2. [“Kuyruk yöneticisi yapılandırmasının çekilmesi” sayfa 37](#)
3. İsteğe bağlı: [“İsteğe bağlı: Kuyruk yöneticisi anahtarları ve sertifikalarının alınması ve alınması” sayfa 38](#)
4. İsteğe bağlı: [“İsteğe bağlı: LDAP yapılandırması” sayfa 40](#)
5. İsteğe bağlı: [“İsteğe bağlı: IBM MQ yapılandırmasındaki IP adreslerinin ve anasistem adlarının değiştirilmesi” sayfa 47](#)
6. [“Taşıyıcı ortamı için kuyruk yöneticisi yapılandırmasının güncellenmesi” sayfa 49](#)
7. [“Kapsayıcılarda çalışan IBM MQ için hedef HA mimarisinin seçilmesi” sayfa 51](#)
8. [“Kuyruk yöneticisi için kaynaklar yaratılması” sayfa 52](#)
9. [“Red Hat OpenShift üzerinde yeni kuyruk yöneticisi yaratılıyor” sayfa 53](#)
10. [“Yeni taşıyıcı konuşlandırmasının doğrulanması” sayfa 57](#)

olmadığını denetleme

IBM MQ Operator , IBM MQ Advanced içinde var olan tüm özellikleri içermez ve bu özelliklerin gerekli olmadığını doğrulamanız gerekir. Diğer özellikler kısmen desteklenir ve taşıyıcıdaki kullanılabilir olanları eşleştirmek üzere yeniden yapılandırılabilir.

Başlamadan önce

Bu, [“IBM MQ 'un IBM Cloud Pak for Integration' e geçirilmesi” sayfa 36](#) içindeki ilk adımdır.

Yordam

1. Hedef taşıyıcı görüntüsünün gerekli tüm işlevleri içerdiğini doğrulayın.

En son bilgiler için bkz. [“Kapsayıcılarda IBM MQ ' ı nasıl kullanmak istediğinizi seçme” sayfa 5.](#)

2. IBM MQ Operator , dinleyici olarak bilinen tek bir IBM MQ trafik kapısına sahiptir. Birden çok dinleyiciniz varsa, taşıyıcıda tek bir dinleyici kullanmak için bunu basitleştirin. Bu, yaygın bir senaryo olmadığı için, bu değişiklik ayrıntılı olarak belgelenmez.
3. IBM MQ çıkışlar kullanılıyorsa, bunları IBM MQ çıkış çiftlerine katlayarak taşıyıcıya geçirin. Bu, gelişmiş bir geçiş senaryodur ve bu nedenle buraya eklenmez. Adımların anahattı için bkz. [“Red Hat OpenShift CLI kullanılarak özel MQSC ve INI dosyaları içeren bir resim oluşturma” sayfa 111.](#)
4. IBM MQ sisteminiz High Availability (Yüksek Kullanılabilirlik) içeriyorsa, kullanılabilir seçenekleri gözden geçirin.

Bkz. [“Taşıyıcılarda IBM MQ için yüksek kullanılabilirlik” sayfa 15.](#)

Sonraki adım

Artık [kuyruk yöneticisi yapılandırmasını alma](#) ' ya hazırsınız.

çekilmesi

Yapılandırma çoğunluğu, kuyruk yöneticileri arasında taşınır. Örneğin, uygulamaların, kuyruklar, konular ve kanalların tanımları gibi etkileşimde bulunduğu şeyler. Yapılandırmayı, var olan IBM MQ kuyruk yöneticisinden çıkarmak için bu görevi kullanın.

Başlamadan önce

Bu görev, [imled that requirend functions are avasolduğunu varsayar.](#)

Yordam

1. Var olan IBM MQ kuruluşuyla makinede oturum açın.
2. Yapılandırmayı yedekle.

Aşağıdaki komutu çalıştırın:

```
dmpmqcfg -m QMGR_NAME > /tmp/backup.mqsc
```

Bu komut için kullanım notları:

- Bu komut, yedeklemeyi tmp dizininde saklar. Yedeklemeyi başka bir yerde saklayabilirsiniz, ancak bu senaryo sonraki komutlara ilişkin tmp dizinini varsayar.

- Replace `QMGR_NAME` with the queue manager name from your environment. Değerden emin değilseniz, makineden kullanılabilir kuyruk yöneticilerini görüntülemek için **dspm** komutunu çalıştırın. Aşağıda qm1adlı bir kuyruk yöneticisi için örnek **dspm** komut çıkışı vardır:

```
QMNAME(qm1)                STATUS(Running)
```

dspm komutu, IBM MQ kuyruk yöneticisinin başlatılmasını gerektirir; tersi durumda, aşağıdaki hatayı alırsınız:

```
AMQ8146E: IBM MQ queue manager not available.
```

Gerekliyse, aşağıdaki komutu çalıştırarak kuyruk yöneticisini başlatın:

```
startmq QMGR_NAME
```

Sonraki adım

Artık [Kuyruk yöneticisi anahtarlarını ve sertifikalarını alma ve alma](#) a hazırsınız.

OpenShift V 9.2.1 CD EUS İsteğe bağlı: Kuyruk yöneticisi anahtarları ve sertifikalarının alınması ve alınması

IBM MQ , trafiği kuyruk yöneticisine şifrelemek için TLS kullanılarak yapılandırılabilir. Kuyruk yöneticinizin, anahtarları ve sertifikaları almak ve geçirilen kuyruk yöneticisi üzerinde TLS ' yi yapılandırmak için kuyruk yöneticinizin TLS kullanıp kullanmadığından emin olmak için bu görevi kullanın.

Başlamadan önce

Bu görev, [kuyruk yöneticisi yapılandırmasını çıkardığınız](#) varsayar.

Bu görev hakkında

Bunu yapmam gerekiyor mu?

IBM MQ , trafiği kuyruk yöneticisine şifrelemek için yapılandırılabilir. Bu şifreleme, kuyruk yöneticisinde yapılandırılmış bir anahtar havuzu kullanılarak tamamlanır. IBM MQ channels then enable the TLS communication. Ortamınızda yapılandırılıp yapılandırıldığını bilmiyorsanız, doğrulamak için aşağıdaki komutu çalıştırın:

```
grep 'SECCOMM(ALL\|SECCOMM(ANON\|SSLIPH)' backup.mqsc
```

Herhangi bir sonuç bulunamazsa, TLS kullanılmamaktadır. Ancak bu, TLS ' nin geçirilen kuyruk yöneticisinde yapılandırılmaması gerektiği anlamına gelmez. Bu davranışı değiştirmek istemeniz için birkaç neden vardır:

- Red Hat OpenShift ortamındaki güvenlik yaklaşımı, önceki ortama göre geliştirilmelidir.
- Yeni düzeye geçirilen kuyruk yöneticisine Red Hat OpenShift ortamının dışından erişmeniz gerekiyorsa, TLS ' nin Red Hat OpenShift Route 'dan geçmesi gerekir.

Yordam

1. Güvenilir sertifikaları var olan mağazadan çıkarın.

TLS şu anda kuyruk yöneticisinde kullanımdaysa, kuyruk yöneticisinin saklanmış bir dizi güvenilir sertifikası olabilir. Bunların ayıklanması ve yeni kuyruk yöneticisine kopyalanması gerekir. Aşağıdaki isteğe bağlı adımlardan birini tamamlayın:

- Sertifikaların ayıklamayı kolaylaştırmak için yerel sistemde şu komut dosyasını çalıştırın:

```
#!/bin/bash

keyr=$(grep SSLKEYR $1)
if [ -n "${keyr}" ]; then
    keyrlocation=$(sed -n "s/^.*\(.*\)'.*$/\1/ p" <<< ${keyr})
    mapfile -t runmqckmResult < <(runmqckm -cert -list -db ${keyrlocation}.kdb -stashed)
    cert=1
    for i in "${runmqckmResult[@]:1}"
    do
        certlabel=$(echo ${i} | xargs)
        echo Extracting certificate $certlabel to $cert.cert
        runmqckm -cert -extract -db ${keyrlocation}.kdb -label "$certlabel" -target $
    {cert}.cert -stashed
        cert=$((cert+1))
    done
fi
```

Komut dosyasını çalıştırırken bağımsız değişken olarak IBM MQ yedeğinin yerini belirtin ve sertifikalar çıkarılır. Örneğin, komut dosyası `extractCert.sh` olarak adlandırıldıysa ve IBM MQ yedeklemesi `/tmp/backup.mqsc` konumunda bulunuyorsa, aşağıdaki komutu çalıştırın:

```
extractCert.sh /tmp/backup.mqsc
```

- Diğer bir seçenek olarak, gösterilen sırayla aşağıdaki komutları çalıştırın:

a. TLS deposunun yerini belirtin:

```
grep SSLKEYR /tmp/backup.mqsc
```

Örnek çıkış:

```
SSLKEYR('/run/runmqserver/tls/key') +
```

Anahtar deposunun bulunduğu yer: `/run/runmqserver/tls/key.kdb`

- b. Bu konum bilgilerine dayalı olarak, saklanan sertifikaları belirlemek için anahtar deposunu sorgulayın:

```
runmqckm -cert -list -db /run/runmqserver/tls/key.kdb -stashed
```

Örnek çıkış:

```
Certificates in database /run/runmqserver/tls/key.kdb:
    default
    CN=cs-ca-certificate,0=cert-manager
```

- c. Listelenen sertifikaların her birini çıkarın. Bunu yapmak için aşağıdaki komutu kullanın:

```
runmqckm -cert -extract -db KEYSTORE_LOCATION -label "LABEL_NAME" -target OUTPUT_FILE -stashed
```

Bu denklemleri önceden gösterilen örneklerde aşağıdakine denk gelir:

```
runmqckm -cert -extract -db /run/runmqserver/tls/key.kdb -label "CN=cs-ca-certificate,0=cert-manager" -target /tmp/cert-manager.crt -stashed
runmqckm -cert -extract -db /run/runmqserver/tls/key.kdb -label "default" -target /tmp/default.crt -stashed
```

2. Kuyruk yöneticisi için yeni bir anahtar ve sertifika edinir

Yeni düzeye geçirilen kuyruk yöneticisi üzerinde TLS 'yi yapılandırmak için yeni bir anahtar ve sertifika üretmiş olun. Bu, daha sonra devreye alma sırasında kullanılır. Birçok kurulumda bu, anahtar

ve sertifika istemek için güvenlik ekibinizle iletişim kurulması anlamına gelir. Bazı kuruluşlarda bu seçenek kullanılamaz ve kendinden onaylı sertifikalar kullanılır.

Aşağıdaki örnek, süre bitiminin 10 yıl olarak ayarlandığı kendinden onaylı bir sertifika oluşturur:

```
openssl req \  
-newkey rsa:2048 -nodes -keyout qmgr.key \  
-subj "/CN=mq queuemanager/OU=ibm mq" \  
-x509 -days 3650 -out qmgr.crt
```

İki yeni dosya yaratılır:

- qmgr.key , kuyruk yöneticisinin özel anahtırı
- qmgr.crt , genel sertifikadır

Sonraki adım

Şimdi [LDAP 'ı yapılandır'](#) a hazır olun.

OpenShift V 9.2.1 CD EUS İsteğe bağlı: LDAP yapılandırması

IBM MQ Operator , farklı güvenlik yaklaşımlarını kullanmak üzere yapılandırılabilir. Genellikle LDAP, bir kurumsal devreye alma için en etkilidir ve LDAP bu geçiş senaryosu için kullanılır.

Başlamadan önce

Bu görev, [Kuyruk yöneticisi anahtarları ve sertifikaları alındı ve alınıdsahip olduğunuz varsayılmıştır](#).

Bu görev hakkında

Bunu yapmam gerekiyor mu?

Kimlik doğrulama ve yetkilendirme için zaten LDAP kullanıyorsanız, herhangi bir değişiklik gerekmez.

LDAP ' ın kullanılıp kullanılmadığından emin değilseniz, aşağıdaki komutu çalıştırın:

```
connauthname="$ (grep CONNAUTH backup.mqsc | cut -d "(" -f2 | cut -d ")" -f1)"; grep -A 20  
AUTHINFO\($connauthname\) backup.mqsc
```

Örnek çıkış:

```
DEFINE AUTHINFO('USE.LDAP') +  
  AUTHTYPE(IDPWLDAP) +  
  ADOPTCTX(YES) +  
  CONNAME('ldap-service.ldap(389)') +  
  CHCKCLNT(REQUIRED) +  
  CLASSGRP('groupOfUniqueNames') +  
  FINDGRP('uniqueMember') +  
  BASEDNG('ou=groups,dc=ibm,dc=com') +  
  BASEDNU('ou=people,dc=ibm,dc=com') +  
  LDAPUSER('cn=admin,dc=ibm,dc=com') +  
  * LDAPPWD('*****') +  
  SHORTUSR('uid') +  
  GRPFIELD('cn') +  
  USRFIELD('uid') +  
  AUTHORMD(SEARCHGRP) +  
  * ALTDAT(2020-11-26) +  
  * ALTTIME(15.44.38) +  
  REPLACE
```

Çıkışta belirli bir ilgi alanı olan iki öznitelik vardır:

YAZAR TIPI

IDPWLDAPdeğeri varsa, kimlik doğrulama için LDAP ' ı kullanıyorsunuz demektir.

Değer boşsa ya da başka bir değerse, LDAP yapılandırılmaz. In this case, check the AUTHORMD attribute to see if LDAP users are being used for authorization.

YAZDIRMA

OSdeğeri varsa, yetkilendirme için LDAP kullanmayasınız demektir.

LDAP 'ı kullanmak üzere yetkilendirmeyi ve kimlik doğrulamasını değiştirmek için aşağıdaki görevleri tamamlayın:

Yordam

1. LDAP sunucusu için IBM MQ yedeğini güncelleyin.
2. LDAP yetkilendirme bilgileri için IBM MQ yedeğini güncelleyin.

OpenShift V 9.2.1 CD EUS **LDAP 1. kısım: LDAP sunucusu için IBM**

MQ yedeğinin güncellenmesi

LDAP 'ı ayarlamaya ilişkin kapsamlı bir açıklama, bu senaryonun kapsamı dışında olur. Bu konu, daha fazla bilgi için sürecin özetini, bir örneği ve başvurularını verir.

Başlamadan önce

Bu görev, [Kuyruk yöneticisi anahtarları ve sertifikaları alındı ve alındısahip olduğunuz varsayılmıştır](#).

Bu görev hakkında

Bunu yapmam gerekiyor mu?

Kimlik doğrulama ve yetkilendirme için zaten LDAP kullanıyorsanız, herhangi bir değişiklik gerekmez. LDAP 'ın kullanılıp kullanılmadığından emin değilseniz, bkz. [“İsteğe bağlı: LDAP yapılandırması” sayfa 40](#).

LDAP sunucusunu ayarlamaya ilişkin iki kısım vardır:

1. [Bir LDAP yapılandırması tanımlayın](#).
2. [LDAP yapılandırmasını kuyruk yöneticisi tanımlamasıyla ilişkilendirin](#).

Bu yapılanışınız için size yardımcı olacak ek bilgiler:

- [Kullanıcı Havuzuna Genel Bakış](#)
- [AUTHINFO komutuna ilişkin başvuru kılavuzu](#)

Yordam

1. Bir LDAP yapılandırması tanımlayın.

LDAP sistemi için yeni bir **AUTHINFO** nesnesi tanımlamak üzere `backup.mqsc` dosyasını düzenleyin. Örneğin:

```
DEFINE AUTHINFO(USE.LDAP) +
  AUTHTYPE(IDPWLDAP) +
  CONNAME('ldap-service.ldap(389)') +
  LDAPUSER('cn=admin,dc=ibm,dc=com') +
  LDAPPWD('admin') +
  SECCOMM(NO) +
  USRFIELD('uid') +
  SHORTUSR('uid') +
  BASEDNU('ou=people,dc=ibm,dc=com') +
  AUTHORMD(SEARCHGRP) +
  BASEDNG('ou=groups,dc=ibm,dc=com') +
  GRPFIELD('cn') +
  CLASSGRP('groupOfUniqueNames') +
  FINDGRP('uniqueMember')
REPLACE
```

burada:

- **CONNAME** , LDAP sunucusunun ana makine adı ve kapısıdır. Esneklik için birden çok adres varsa, bunlar virgülle ayrılmış bir liste kullanılarak yapılandırılabilir.

- **LDAPUSER** , kullanıcı kayıtlarını sorgulamak için LDAP 'ye bağlanırken IBM MQ ' in kullandığı kullanıcı ile ilgili ayırt edici addır.
- **LDAPPWD** , **LDAPUSER** kullanıcısına karşılık gelen paroladır.
- **SECCOM** , LDAP sunucusu ile iletişimin TLS ' yi kullanıp kullanmayacağını belirtir. Olası değerler:
 - YES: TLS kullanılır ve bir sertifika IBM MQ sunucusu tarafından sunulur.
 - ANON: TLS, IBM MQ sunucusu tarafından sunulmakta olan bir sertifika olmadan kullanılır.
 - NO: TLS, bağlantı sırasında kullanılmaz.
- **USRFIELD** , LDAP kaydındaki, sunulan kullanıcı adının karşılaştırılacağı alanı belirtir.
- **SHORTUSR** , LDAP kaydındaki uzunluğu 12 karakteri aşmayan bir alandır. Kimlik doğrulaması başarılı olursa, bu alandaki değer, değerlendirilen kimliğe sahip olur.
- **BASEDNU** , LDAP aramak için kullanılması gereken temel DN 'dir.
- **BASEDNG** , LDAP içindeki gruplar için temel DN 'dir.
- **AUTHORMD** , kullanıcıya ilişkin grup üyeliğini çözümlmek için kullanılan mekanizmayı tanımlar. Dört seçenek vardır:
 - OS: Kısa adla ilişkili grupların işletim sistemini sorgulayın.
 - SEARCHGRP: Kimliği doğrulanan kullanıcı için LDAP ' taki grup girdilerinde arama yapın.
 - SEARCHUSR: Grup üyeliği bilgileri için kimliği doğrulanmış kullanıcı kaydında arama yapın.
 - SRCHGRPSN: LDAP ' taki grup girişlerinde, kimliği doğrulanan kullanıcılar için kısa kullanıcı adı (SHORTUSR alanı tarafından tanımlanır) için arama yapın.
- **GRPFIELD** , LDAP grubu kaydındaki bir yalın ada karşılık gelen öznitedir. Belirtilirse, yetki kayıtlarını tanımlamak için bu kullanılabilir.
- **CLASSUSR** , bir kullanıcıya karşılık gelen LDAP nesne sınıfıdır.
- **CLASSGRP** , bir gruba karşılık gelen LDAP nesne sınıfıdır.
- **FINDGRP** , LDAP kaydındaki grup üyeliğine karşılık gelen öznitedir.

Yeni giriş, dosya içinde herhangi bir yere yerleştirilebilir; ancak, dosyanın başında yeni bir giriş bulunmanın yararlı olduğunu da bulabilirsiniz:

```
Open [icon]
backup.mqsc
*****
* Script generated on 2020-10-21 at 11.48.32
* Script generated by user ' CallumJackso' on host 'LAPTOP-VLQ
* Queue manager name: qm1
* Queue manager platform: Windows
* Queue manager command level: (920/920)
* Command issued: dmpmqcfg -m qm1
*****
DEFINE AUTHINFO(USE.LDAP) +
  AUTHTYPE(IDPWLDAP) +
  CONNAME('ldap-service.ldap(389)') +
  LDAPUSER('cn=admin,dc=ibm,dc=com') +
  LDAPPWD('admin') +
  SECCOMM(NO) +
  USRFIELD('uid') +
  SHORTUSR('uid') +
  BASEDNU('ou=people,dc=ibm,dc=com') +
  AUTHORMD(SEARCHGRP) +
  BASEDNG('ou=groups,dc=ibm,dc=com') +
  GRPFIELD('cn') +
  CLASSGRP('groupOfUniqueNames') +
  FINDGRP('uniqueMember') +
  REPLACE
ALTER QMGR +
* ALTDATE(2020-10-26) +
* ALTTIME(11.43.11) +
```

2. LDAP yapılandırmasını kuyruk yöneticisi tanımlamasıyla ilişkilendirin.

LDAP yapılandırmasını kuyruk yöneticisi tanımlamasıyla ilişkilendirmeniz gerekir. DEFINE AUTHINFO girdisinin hemen altında bir ALTER QMGR girdisi vardır. CONNAUTH girişini, yeni yaratılan AUTHINFO adına karşılık gelecek şekilde değiştirin. For example in the previous example AUTHINFO(USE.LDAP) was defined, meaning the name is USE.LDAP. Bu nedenle, CONNAUTH('SYSTEM.DEFAULT.AUTHINFO.IDPWOS') seçeneğini CONNAUTH('USE.LDAP') olarak değiştirin:

```
Open [icon]
backup.mqsc
*****
* Script generated on 2020-10-21 at 11.48.32
* Script generated by user ' CallumJackso' on host 'L
* Queue manager name: qm1
* Queue manager platform: Windows
* Queue manager command level: (920/920)
* Command issued: dmpmqcfg -m qm1
*****
DEFINE AUTHINFO(USE.LDAP) +
  AUTHTYPE(IDPWLDAP) +
  CONNAME('ldap-service.ldap(389)') +
  LDAPUSER('cn=admin,dc=ibm,dc=com') +
  LDAPPWD('admin') +
  SECCOMM(NO) +
  USRFIELD('uid') +
  SHORTUSR('uid') +
  BASEDNU('ou=people,dc=ibm,dc=com') +
  AUTHORMD(SEARCHGRP) +
  BASEDNG('ou=groups,dc=ibm,dc=com') +
  GRPFIELD('cn') +
  CLASSGRP('groupOfUniqueNames') +
  FINDGRP('uniqueMember') +
  REPLACE
ALTER QMGR +
* ALTDATE(2020-10-26) +
* ALTTIME(11.43.11) +
  CCSID(850) +
  CERTLABL('default') +
  CLWLUSEQ(LOCAL) +
* COMMANDO(SYSTEM_ADMIN_COMMAND.QUEUE) +
  CONNAUTH('USE.LDAP') +
```

LDAP ' in hemen gerçekleşmesi için, ALTER QMGR komutundan hemen sonra bir satır ekleyerek bir REFRESH SECURITY komutunu çağırın:

```
*backup.mqsc
*****
* Script generated on 2020-10-21 at 11.48.32
* Script generated by user ' CallumJackso' on host 'LAPTOP-VLQKJ5UH'
* Queue manager name: qm1
* Queue manager platform: Windows
* Queue manager command level: (920/920)
* Command issued: dmpmqcfc -m qm1
*****
DEFINE AUTHINFO(USE.LDAP) +
  AUTHTYPE(IDPWLDAP) +
  CONNAME('ldap-service.ldap(389)') +
  LDAPUSER('cn=admin,dc=ibm,dc=com') +
  LDAPPWD('admin') +
  SECCOMM(NO) +
  USRFIELD('uid') +
  SHORTUSR('uid') +
  BASEDNU('ou=people,dc=ibm,dc=com') +
  AUTHORMD(SEARCHGRP) +
  BASEDNG('ou=groups,dc=ibm,dc=com') +
  GRPFIELD('cn') +
  CLASSGRP('groupOfUniqueNames') +
  FINDGRP('uniqueMember') +
  REPLACE
ALTER QMGR +
* ALTDATA(2020-10-26) +
* ALTTIME(11.43.11) +
  CCSID(850) +
  CERTLABL('default') +
  CLWLUSEQ(LOCAL) +
* COMMANDQ(SYSTEM.ADMIN.COMMAND.QUEUE) +
  CONNAUTH('USE.LDAP') +
* CRDATE(2020-10-26) +
* CRTIME(11.43.11) +
* QMID(qm1_2020-10-26_11.43.11) +
  SSLCRYP(' ') +
  SSLKEYR('/run/runmqserver/tls/key') +
  SUITEB(NONE) +
* VERSION(09020000) +
  FORCE
REFRESH SECURITY
```

Sonraki adım

Şimdi, LDAP yetkilendirme bilgileri için IBM MQ yedeklemesini güncellemeye hazırsınız.

OpenShift V 9.2.1 CD EUS **LDAP parça 2: LDAP yetkilendirme**

bilgileri için IBM MQ yedeğinin güncellenmesi

IBM MQ , IBM MQ nesnelere erişimi denetleyen, ayrıntılı yetkilendirme kuralları sağlar. LDAP ' de kimlik doğrulamasını ve yetkilendirmeyi değiştirdiyseniz, yetkilendirme kuralları geçersiz olabilir ve güncelleme gerektirebilir.

Başlamadan önce

Bu görev, LDAP sunucusu için yedeklemeyi güncellemiş olduğunu varsayar.

Bu görev hakkında

Bunu yapmam gerekiyor mu?

Kimlik doğrulama ve yetkilendirme için zaten LDAP kullanıyorsanız, herhangi bir değişiklik gerekmez. LDAP 'ın kullanılıp kullanılmadığından emin değilseniz, bkz. [“İsteğe bağlı: LDAP yapılandırması” sayfa 40.](#)

LDAP yetkilendirme bilgilerini güncellemeye ilişkin iki kısım vardır:

1. Dosyadan var olan tüm yetkilendirmeyi kaldırın.
2. LDAP için yeni yetki bilgilerini tanımlayın.

Yordam

1. Var olan tüm yetkilendirmeyi dosyadan kaldırın.

Dosyanın sonuna yakın yedekleme dosyasında, SET AUTHREC ile başlayan bazı girdileri görmeniz gerekir:

```
Open [icon] *backup.mqsc /tmp
OBJTYPE(PROCESS) +
AUTHADD(CRT)
SET AUTHREC +
PROFILE('@CLASS') +
PRINCIPAL('CallumJackson@AzureAD') +
OBJTYPE(QMGR) +
AUTHADD(CRT)
SET AUTHREC +
PROFILE('@CLASS') +
GROUP('mqm@LAPTOP-VLQKJ5UH') +
OBJTYPE(QMGR) +
AUTHADD(CRT)
SET AUTHREC +
PROFILE('SYSTEM.ADMIN.CHANNEL.EVENT') +
PRINCIPAL('CallumJackson@AzureAD') +
OBJTYPE(Queue) +
AUTHADD(BROWSE,CHG,CLR,DLT,DSP,GET,INQ,PUT,PASSALL,PASSID,SET,SETALL,SETID)
SET AUTHREC +
PROFILE('SYSTEM.ADMIN.CHANNEL.EVENT') +
GROUP('mqm@LAPTOP-VLQKJ5UH') +
OBJTYPE(Queue) +
AUTHADD(BROWSE,CHG,CLR,DLT,DSP,GET,INQ,PUT,PASSALL,PASSID,SET,SETALL,SETID)

* Script ended on 2020-10-26 at 11.48.32
* Number of Inquiry commands issued: 14
* Number of Inquiry commands completed: 14
* Number of Inquiry responses processed: 295
* QueueManager count: 1
* Queue count: 57
* NameList count: 3
* Process count: 1
* Channel count: 11
* AuthInfo count: 4
* Listener count: 4
* Service count: 2
* CommInfo count: 1
* Topic count: 6
* Subscription count: 1
* ChlAuthRec count: 3
* AuthRec count: 199
* Number of objects/records: 293
*****
```

Var olan girdileri bulun ve silin. En basit yaklaşım, var olan tüm SET AUTHREC kurallarını kaldırdıktan sonra, LDAP girdilerine dayalı olarak yeni girdiler oluşturmasıdır.

2. LDAP için yeni yetki bilgilerini tanımlayın

Kuyruk yöneticisi yapılandırılmasına ve kaynak ve grup sayısına bağlı olarak, bu bir zaman alıcı ya da basit bir etkinlik olabilir. Aşağıdaki örnekte, kuyruk yöneticinizin Q1adlı tek bir kuyruk olduğu ve apps LDAP grubunun erişime sahip olması için izin vermek istediğiniz varsayılmıştır.

```
SET AUTHREC GROUP('apps') OBJTYPE(QMGR) AUTHADD(ALL)
SET AUTHREC PROFILE('Q1') GROUP('apps') OBJTYPE(Queue) AUTHADD(ALL)
```

İlk AUTHREC komutu kuyruk yöneticisine erişim izni ekler; ikincisi ise kuyruğa erişim sağlar. İkinci bir kuyruğa erişim gerekiyorsa, daha sosyal erişim sağlamak için joker karakter kullanmaya karar vermezseniz, üçüncü bir AUTHREC komutu gereklidir.

İşte başka bir örnek. Bir yönetici grubunun (adminsadlı) kuyruk yöneticisine tam erişime ihtiyacı varsa, aşağıdaki komutları ekleyin:

```
SET AUTHREC PROFILE('*') OBJTYPE(Queue) GROUP('admins') AUTHADD(ALL)
SET AUTHREC PROFILE('*') OBJTYPE(TOPIC) GROUP('admins') AUTHADD(ALL)
SET AUTHREC PROFILE('*') OBJTYPE(CHANNEL) GROUP('admins') AUTHADD(ALL)
SET AUTHREC PROFILE('*') OBJTYPE(CLNTCONN) GROUP('admins') AUTHADD(ALL)
SET AUTHREC PROFILE('*') OBJTYPE(AUTHINFO) GROUP('admins') AUTHADD(ALL)
SET AUTHREC PROFILE('*') OBJTYPE(LISTENER) GROUP('admins') AUTHADD(ALL)
SET AUTHREC PROFILE('*') OBJTYPE(NAMELIST) GROUP('admins') AUTHADD(ALL)
SET AUTHREC PROFILE('*') OBJTYPE(PROCESS) GROUP('admins') AUTHADD(ALL)
SET AUTHREC PROFILE('*') OBJTYPE(SERVICE) GROUP('admins') AUTHADD(ALL)
SET AUTHREC PROFILE('*') OBJTYPE(QMGR) GROUP('admins') AUTHADD(ALL)
```

Sonraki adım

Artık IBM MQ yapılandırmasındaki IP adreslerini ve anasistem adlarını [değiştirme](#)' a hazırsınız.

OpenShift V 9.2.1 CD EUS İsteğe bağlı: IBM MQ yapılandırmasındaki IP adreslerinin ve anasistem adlarının değiştirilmesi

IBM MQ yapılandırmasında IP adresleri ve anasistem adları belirtilmiş olabilir. Bazı durumlarda bu durum kalabilir, ancak diğer durumlarda da güncellenmesi gerekir.

Başlamadan önce

Bu görev, [yapılandırılan LDAP](#) sahip olduğunuz varsayılmıştır.

Bu görev hakkında

Bunu yapmam gerekiyor mu?

Önce, bir önceki bölümde tanımlanan LDAP yapılandırması dışında, herhangi bir IP adresi ya da anasistem adınız olup olmadığını belirleyin. Bunu yapmak için aşağıdaki komutu çalıştırın:

```
grep 'CONNAME\|LOCLADDR\|IPADDRV' -B 3 backup.mqsc
```

Örnek çıkış:

```
*****
DEFINE AUTHINFO(USE.LDAP) +
  AUTHTYPE(IDPWLDAP) +
  CONNAME('ldap-service.ldap(389)') +
--
DEFINE AUTHINFO('SYSTEM.DEFAULT.AUTHINFO.IDPWLDAP') +
  AUTHTYPE(IDPWLDAP) +
  ADOPTCTX(YES) +
  CONNAME(' ') +
--
REPLACE
DEFINE AUTHINFO('SYSTEM.DEFAULT.AUTHINFO.CRLLDAP') +
```

```
AUThTYpE(CRLLDAP) +  
CONNAME(' ') +
```

Bu örnekte arama üç sonuç döndürür. Bir sonuç daha önce tanımlanan LDAP yapılandırmasına karşılık gelir. LDAP sunucusunun anasistem adı aynı kaldığından, bu değer yoksayılabılır. Diğer iki sonuç da boş bağlantı girişleridir; bu nedenle bunlar yoksayılabılır. Başka bir girişiniz yoksa, bu konunun geri kalanını atlayabilirsiniz.

Yordam

1. Döndürülen girdileri anlasın.

IBM MQ , yapılandırmanın birçok yönünün IP adreslerini, anasistem adlarını ve kapılarını içerebilir. Bunları iki kategori olarak sınıflandırabiliriz:

- Bu kuyruk yöneticisinin yeri:** Bu kuyruk yöneticisinin kullandığı ya da yayınladığı konum bilgileri, bir IBM MQ ağı içindeki diğer kuyruk yöneticilerinin ya da uygulamaların bağlantı için kullanabilecekleri bir yer.
- Kuyruk yöneticisi bağımlılıklarının yeri:** Bu kuyruk yöneticisinin farkında olması gereken diğer kuyruk yöneticilerinin ya da sistemlerin yerleri.

Bu senaryo yalnızca, bu kuyruk yöneticisi yapılandırmasındaki değişikliklere odaklandığından, yalnızca kategoriye ilişkin yapılandırma güncellemelerini işliyoruz (a). Ancak, bu kuyruk yöneticisi konumuna diğer kuyruk yöneticileri ya da uygulamalar tarafından başvurulursa, bunların yapılandırmalarının bu kuyruk yöneticisinin yeni yerine eşleşmesi için güncelleştirmeye gerek duyabilir.

Güncellenmesi gereken bilgileri içerebilecek iki anahtar nesnesi vardır:

- Dinleyiciler:** Bu, IBM MQ ' in dinlediği ağ adresini temsil eder.
- KüME NESNESİ KANALI:** Kuyruk yöneticisi bir IBM MQ kümesinin bir parçasıysa, bu nesne var demektir. Bu, diğer kuyruk yöneticilerinin bağlanabileceği ağ adresini belirtir.

2. `grep 'CONNAME\|LOCLADDR\|IPADDRV' -B 3 backup.mqsc` komutunun özgün çıktısında, herhangi bir CLUSTER RECEIVER kanallarının tanımlı olup olmadığını saptayın. Varsa, IP adreslerini güncelleyin.

Herhangi bir CLUSTER RECEIVER kanallarının tanımlı olup olmadığını saptamak için, özgün çıktıda CHLTYPE (CLUSRCVR) ile herhangi bir giriş bulun:

```
DEFINE CHANNEL(ANY_NAME) +  
CHLTYPE(CLUSRCVR) +
```

Girişler varsa, CONNAME ' u IBM MQ Red Hat OpenShift Rotanıyla güncelleyin. Bu değer Red Hat OpenShift ortamını temel alır ve öngörülebilir bir sözdizimi kullanır:

```
queue_manager_resource_name-ibm-mq-qm-openshift_project_name.openshift_app_route_hostname
```

Örneğin, kuyruk yöneticisi konuşlandırması cp4i ad alanı içinde qm1 olarak adlandırıldıysa ve `openshift_app_route_hostname` apps.callumj.icp4i.com ise, yol URL adresi şöyledir:

```
qm1-ibm-mq-qm-cp4i.apps.callumj.icp4i.com
```

Rotaya ilişkin kapı numarası genellikle 443 'tür. Red Hat OpenShift Denetimciniz size farklı bir şekilde bildirmediği sürece, bu durum olağan olarak doğru değerdir. Bu bilgileri kullanarak CONNAME alanlarını güncelleyin. Örneğin:

```
CONNAME('qm1-ibm-mq-qm-cp4i.apps.callumj.icp4i.com(443)')
```


grep 'CONNAME\|LOCLADDR\|IPADDRV' -B 3 backup.mqsc komutundan özgün çıktıda, LOCLADDR ya da IPADDRV için herhangi bir giriş olup olmadığını doğrulayın. Varsa, silin. Bunlar kapsayıcı bir ortamda ilgili değildir.

Sonraki adım

Artık Kapsayıcı bir ortam için kuyruk yöneticisi yapılandırmasını güncelleme' a hazırsınız.

Taşıyıcı ortamı için kuyruk yöneticisi yapılandırmasının güncellenmesi

Bir taşıyıcıda çalışırken, bazı yapılandırma yönleri taşıyıcı tarafından tanımlanır ve dışa aktarılan yapılandırmayla çakışabilir.

Başlamadan önce

Bu görev, IP adreslerinin ve anasistem adlarının IBM MQ yapılandırmasını değiştirdi sahip olduğunuz varsayılmıştır.

Bu görev hakkında

Kapsayıcı tarafından aşağıdaki yapılandırma öğeleri tanımlanır:

- Dinleyici tanımları (gösterilen kapılara karşılık gelir).
- Herhangi bir potansiyel TLS deposunun konumu.

Bu nedenle, dışa aktarılan yapılandırma güncellemeniz gerekir:

1. Dinleyici tanımlarını kaldırın.
2. TLS anahtar havuzunun konumunu tanımlayın.

Yordam

1. Dinleyici tanımlamalarını kaldırın.

Yedekleme yapılandırmasında, DEFINE LISTENER için arama yapın. Bu, AUTHINFO ile SERVICE tanımlamaları arasında olmalıdır. Alanı vurgulayın ve silin.

```
** ALTDATA(2020-11-26) +
* ALTTIME(11.43.28) +
  REPLACE
DEFINE AUTHINFO('SYSTEM.DEFAULT.AUTHINFO.CRLLDAP') +
  AUTHTYPE(CRLLDAP) +
  CONNAME(' ') +
* ALTDATA(2020-10-26) +
* ALTTIME(11.43.28) +
  REPLACE
DEFINE LISTENER('SYSTEM.DEFAULT.LISTENER.LU62') +
  TRPTYPE(LU62) +
  CONTROL(MANUAL) +
* ALTDATA(2020-10-26) +
* ALTTIME(11.43.28) +
  REPLACE
DEFINE LISTENER('SYSTEM.DEFAULT.LISTENER.NETBIOS') +
  TRPTYPE(NETBIOS) +
  CONTROL(MANUAL) +
  LOCLNAME(' ') +
* ALTDATA(2020-10-26) +
* ALTTIME(11.43.28) +
  REPLACE
DEFINE LISTENER('SYSTEM.DEFAULT.LISTENER.SPX') +
  TRPTYPE(SPX) +
  CONTROL(MANUAL) +
* ALTDATA(2020-10-26) +
* ALTTIME(11.43.28) +
  REPLACE
DEFINE LISTENER('SYSTEM.DEFAULT.LISTENER.TCP') +
  TRPTYPE(TCP) +
  CONTROL(MANUAL) +
* ALTDATA(2020-10-26) +
* ALTTIME(11.43.28) +
  REPLACE
DEFINE SERVICE('SYSTEM.AMQP.SERVICE') +
  CONTROL(QMGR) +
  SERVTYPE(SERVER) +
  STARTCMD('+MQ_INSTALL_PATH+\bin\amqp.bat') +
  STARTARG('start -m +QMNAME+ -d "+MQ_Q_MGR_DATA_PATH+\.'
  STOPCMD('+MQ_INSTALL_PATH+\bin64\endmqsd.exe') +
```

2. TLS anahtar havuzunun konumunu tanımlayın.

Kuyruk yöneticisi yedeklemesi, özgün ortama ilişkin TLS yapılandırmasını içerir. Bu, kapsayıcı ortamdan farklıdır; dolayısıyla, birkaç güncelleme gereklidir:

- **CERTLABL** girdisini default olarak değiştirin
- TLS anahtar havuzunun konumunu (**SSLKEYR**) şu şekilde değiştirin: /run/runmqserver/tls/key

Dosyadaki **SSLKEYR** özniteliğinin konumunu bulmak için **SSLKEYR** dosyasını arayın. Genellikle yalnızca bir giriş bulunur. Birden çok girdi bulunursa, aşağıdaki şekilde gösterildiği gibi, **QMGR** nesnesini düzenliyorsunuz gibi bir kontrol edin.

```

*backup.mqsc
*****
* Script generated on 2020-10-21 at 11.48.32
* Script generated by user ' CallumJackso' on host 'LAPTOP-VLQKJ5UH'
* Queue manager name: qm1
* Queue manager platform: Windows
* Queue manager command level: (920/920)
* Command issued: dmpmqcfg -m qm1
*****
DEFINE AUTHINFO(USE.LDAP) +
  AUTHTYPE(IDPWLDAP) +
  CONNAME('ldap-service.ldap(389)') +
  LDAPUSER('cn=admin,dc=ibm,dc=com') +
  LDAPPWD('admin') +
  SECCOMM(NO) +
  USRFIELD('uid') +
  SHORTUSR('uid') +
  BASEDNU('ou=people,dc=ibm,dc=com') +
  AUTHORMD(SEARCHGRP) +
  BASEDNG('ou=groups,dc=ibm,dc=com') +
  GRPFIELD('cn') +
  CLASSGRP('groupOfUniqueNames') +
  FINDGRP('uniqueMember') +
  REPLACE
ALTER QMGR +
* ALTDATE(2020-10-26) +
* ALTTIME(11.43.11) +
  CCSTD(850) +
  CERTLABL('default') +
  CLWLUSEQ(LOCAL) +
* COMMANDQ(SYSTEM.ADMIN.COMMAND.QUEUE) +
  CONNAUTH('USE.LDAP') +
* CRDATE(2020-10-26) +
* CRTIME(11.43.11) +
* QMID(qm1_2020-10-26_11.43.11) +
  SSLCRYP(' ') +
  SSLKEYR('/run/runmqserver/tls/key') +
  SUITEB(NONE) +
* VERSION(09020000) +
  FORCE
REFRESH SECURITY

```

Sonraki adım

Artık Kapsayıcılarda çalışan IBM MQ için hedef mimariyi seçme' a hazırsınız.

OpenShift V 9.2.1 CD EUS Kapsayıcılarda çalışan IBM MQ için hedef HA mimarisinin seçilmesi

Yüksek kullanılabilirlik gereksinimlerinizi karşılamak için tek bir eşgörünüm (tek bir Kubernetes Pod 'si) ve çok eşgörünümlü (iki Pd) arasında seçim yapın.

Başlamadan önce

Bu görev, [Kapsayıcı bir ortam için kuyruk yöneticisi yapılandırmasını güncelleştir](#)disahip olduğunuz varsayılmıştır.

Bu görev hakkında

IBM MQ Operator , iki yüksek kullanılabilirlik seçeneği sağlar:

- **Tek eşgörünüm:** Tek bir taşıyıcı (Pod) başlatılır ve bir hata durumunda Red Hat OpenShift ' ın yeniden başlatılması sorumluluğu olur. Kubernetesiçinde durumlu bir ayarın özellikleri nedeniyle, bu hata durumunda yedek sisteme geçişin uzun sürebileceği ya da bir yönetim işleminin tamamlanmasını gerektiren birkaç durum vardır.
- **Çoklu eşgörünüm:** İki kap (her biri ayrı bir Pod 'da) başlatıldı, biri etkin kipte, diğeri de yedekte bekleme kipinde. Bu topoloji, hata durumunda daha hızlı geçiş sağlar. Bu, IBM MQ gereksinimlerini karşılayan bir Okuma Yazma Dosya sistemi gerektirir.

Bu görevde, yalnızca hedef HA mimarisini seçiniz. Seçtiğiniz mimarinin yapılandırılmasına ilişkin adımlar, bu senaryoda ("[Red Hat OpenShiftüzerinde yeni kuyruk yöneticisi yaratılıyor](#)" sayfa 53) sonraki bir görevde açıklanmaktadır.

Yordam

1. İki seçeneği gözden geçirin.

Bu iki seçeneğe ilişkin kapsamlı açıklamalar için bkz. "[Taşıyıcılarda IBM MQ için yüksek kullanılabilirlik](#)" sayfa 15.

2. Hedef HA mimarisini seçin.

Hangi seçeneğin seçileceğinden emin değilseniz, **Tek eşgörünüm** seçeneğini kullanarak başlayın ve bu seçeneğin yüksek kullanılabilirlik gereksinimlerinizi karşılayıp karşılamadığını doğrulayın.

Sonraki adım

Artık [kuyruk yöneticisi kaynaklarını yaratma](#)için hazırsınız.

Kuyruk yöneticisi için kaynaklar yaratılması

IBM MQ yapılandırmasını ve TLS sertifikalarını ve anahtarlarını, Red Hat OpenShift ortamına aktarın.

Başlamadan önce

Bu görev, [Kapsayıcılarda çalışan IBM MQ için hedef mimari seçil](#)disahip olduğunuz varsayılmıştır.

Bu görev hakkında

Önceki bölümlerde, iki kaynak ayıklanmış, güncellenmiş ve tanımlandınız:

- IBM MQ yapılandırması
- TLS sertifikaları ve anahtarları

Kuyruk yöneticisi konuşlandırılmadan önce bu kaynakları Red Hat OpenShift ortamına aktarmanız gerekir.

Yordam

1. IBM MQ yapılandırmasını Red Hat OpenShiftiçine aktarın.

Aşağıdaki yönergelerde IBM MQ yapılanışınız, backup .mqscadlı bir dosyada bulunancurrentyapılanışınız olduğunu varsayar. Ters durumda, dosya adını ortamınıza dayalı olarak uyarlamak gerekir.

- a) Log into your cluster using `oc login`.
- b) IBM MQ yapılandırmasını bir `configmap` içine yükleyin.

Aşağıdaki komutu çalıştırın:

```
oc create configmap my-mqsc-migrated --from-file=backup.mqsc
```

- c) Dosyanın başarıyla yüklendiğini doğrulayın.

Aşağıdaki komutu çalıştırın:

```
oc describe configmap my-mqsc-migrated
```

2. IBM MQ TLS kaynaklarını içe aktarın

“İsteğe bağlı: Kuyruk yöneticisi anahtarları ve sertifikalarının alınması ve alınması” sayfa 38' ta ele alındığı gibi, kuyruk yöneticisi konuşlandırması için TLS gerekli olabilir. If so, you should already have a number of files ending with `.crt` and `.key`. Konuşlandırma sırasında gönderme yapmak için, bunları kuyruk yöneticisi için Kubernetes sızlarına eklemeniz gerekir.

Örneğin, kuyruk yöneticisi için bir anahtar ve sertifika aldıysanız şu çağrılabilir:

- `qmgr.crt`
- `qmgr.key`

Bu dosyaları içe aktarmak için şu komutu çalıştırın:

```
oc create secret tls my-tls-migration --cert=qmgr.crt --key=qmgr.key
```

Kubernetes , eşleşen bir genel ve özel anahtar içe aktarırken bu yararlı yardımcı programı sağlar. Ekleme için ek sertifikalar varsa, örneğin kuyruk yöneticisi güven deposuna girmek için aşağıdaki komutu çalıştırın:

```
oc create secret generic my-extra-tls-migration --from-file=comma_separated_list_of_files
```

Örneğin, içe aktarılacak dosyalar `trust1.crt`, `trust2.crt` ve `trust3.crt` ise, komut şöyle olur:

```
oc create secret generic my-extra-tls-migration --from-file=trust1.crt,trust2.crt,trust3.crt
```

Sonraki adım

Artık Red Hat OpenShift üzerinde yeni kuyruk yöneticisini yaratma konusunda hazırsınız.

OpenShift V 9.2.1 CD EUS Red Hat OpenShift üzerinde yeni kuyruk yöneticisi yaratılıyor

Red Hat OpenShift üzerinde tek bir yönetim ortamı ya da çok eşgözümlü kuyruk yöneticisi konuşlandırın.

Başlamadan önce

Bu görev, kuyruk yöneticisi kaynakları yaratıldı ve IBM MQ Operator ürününü Red Hat OpenShift için kurdu nız olduğunu varsayar.

Bu görev hakkında

“Kapsayıcılarda çalışan IBM MQ için hedef HA mimarisinin seçilmesi” sayfa 51 içinde açıklandığı gibi, iki olası devreye alma topolojisi vardır. Bu nedenle bu konu iki farklı şablon sağlar:

- Tek bir eşgörümlü kuyruk yöneticisi konuşlandırın.

- Çoklu eşgörünüm kuyruk yöneticisi konuşlandırır.

Önemli: Tercih edilen topolojisine dayalı olarak, iki şablondan yalnızca birini tamamlayın.

Yordam

- Tek bir eşgörünüm kuyruk yöneticisi konuşlandırın.

Geçirilen kuyruk yöneticisi, bir YAML dosyası kullanarak Red Hat OpenShift ' a konuşlandırılır. Aşağıda, önceki başlarda kullanılan adlara dayalı bir örnek vardır:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: qm1
spec:
  version: 9.2.5.0-r3
  license:
    accept: true
    license: L-RJON-C7QG3S
    use: "Production"
  pki:
    keys:
      - name: default
        secret:
          secretName: my-tls-migration
          items:
            - tls.key
            - tls.crt
  web:
    enabled: true
  queueManager:
    name: QM1
  mqsc:
    - configMap:
        name: my-mqsc-migrated
        items:
          - backup.mqsc
```

Gerçekleştirdiğiniz adımlara bağlı olarak, önceki YAML ' nin uyarlanabileceği bir tarih olabilir. Bu konuda yardımcı olmak için, YAML ' in bir açıklaması var:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: qm1
```

Bu, Kubernetes nesnesini, tipini ve adını tanımlar. Uyarılama gerektiren tek alan, name alanıdır.

```
spec:
  version: 9.2.5.0-r3
  license:
    accept: true
    license: L-RJON-C7QG3S
    use: "Production"
```

Bu, devreye alıma ilişkin sürüm ve lisans bilgilerine karşılık gelir. Bunu özelleştirmeniz gerekiyorsa, [“mq.ibm.com/v1beta1 için lisans başvurusu” sayfa 124](#) içinde sağlanan bilgileri kullanın.

```
pki:
  keys:
    - name: default
      secret:
        secretName: my-tls-migration
        items:
          - tls.key
          - tls.crt
```

Kuyruk yöneticisinin TLS kullanacak şekilde yapılandırılması için, ilgili sertifikalara ve anahtarlara başvurması gerekir. [secretName alanı, IBM MQ TLS kaynaklarını içe aktar kısmında](#) yaratılan

Kubernetes güvenlik dizgisini ve öğelerin listesini (tls.key ve tls.crt), oc create secret tls sözdizimi kullanılırken Kubernetes standart adlarıdır. Güvenli mağazaya eklemek için ek sertifikalar varsa, bunlar benzer bir şekilde eklenebilir, ancak öğeler içe aktarma sırasında kullanılan karşılık gelen dosya adlarıdır. Örneğin, güvenilirlik deposu sertifikalarını yaratmak için aşağıdaki kod kullanılabilir:

```
oc create secret generic my-extra-tls-migration --from-file=trust1.crt,trust2.crt,trust3.crt
```

```
pki:
  trust:
    - name: default
      secret:
        secretName: my-extra-tls-migration
        items:
          - trust1.crt
          - trust2.crt
          - trust3.crt
```

Önemli: TLS gerekli değilse, YAML ' nin TLS bölümünü silin.

```
web:
  enabled: true
```

Bu, devreye alma için web konsolunu etkinleştirir

```
queueManager:
  name: QM1
```

Kuyruk yöneticisinin adını QM1olarak tanımlar. Kuyruk yöneticisi, özgün kuyruk yöneticisi adı gibi, gereksinimlerinize göre uyarlanır.

```
mjsc:
  - configMap:
      name: my-mjsc-migrated
      items:
        - backup.mjsc
```

Önceki kod, IBM MQ yapılandırmasını içe aktar kısmında içe aktarılan kuyruk yöneticisi yapılandırmasındaki çekilir. Farklı adlar kullandıysanız, my-mjsc-migrated ve backup.mjsc' yi değiştirmeniz gerekir.

YAML örneğinin, Red Hat OpenShift ortamına ilişkin varsayılan depolama sınıfının bir RWX ya da RWO depolama sınıfı olarak tanımlandığı varsayıldığını unutmayın. Ortamınızda varsayılan bir varsayılan değer tanımlanmadıysa, kullanılacak depolama sınıfını belirtmeniz gerekir. Bunu, YAML ' yi aşağıda belirtilen şekilde genişleterek yapabilirsiniz:

```
queueManager:
  name: QM1
  storage:
    defaultClass: my_storage_class
    queueManager:
      type: persistent-claim
```

Vurgulanan metni, ortamınızla eşleşmesi için uyarlanmış sınıf özniteliği ile ekleyin. Ortamınızdaki depolama sınıfı adlarını keşfetmek için aşağıdaki komutu çalıştırın:

```
oc get storageclass
```

Burada, bu komutun döndürdüğü örnek çıktı:

NAME	PROVISIONER	RECLAIMPOLICY
aws-efs	openshift.org/aws-efs	Delete
gp2 (default)	kubernetes.io/aws-efs	Delete

Aşağıdaki kod, IBM MQ yapılandırmasını içe aktar bölümünde, içe aktarılan IBM MQ yapılandırmasına nasıl başvurulacağı gösterilmektedir. Farklı adlar kullandıysanız, my-mqsc-migrated ve backup.mqsc' yi değiştirmeniz gerekir.

```
mqsc:
  - configMap:
      name: my-mqsc-migrated
    items:
      - backup.mqsc
```

Tek yönetim ortamı kuyruk yöneticisini konuşlandırdınız. Bu, şablonu tamamlar. Artık yeni taşıyıcı konuşlandırmayı doğrulayın' a hazırsınız.

- Çok eşgörümlü bir kuyruk yöneticisini konuşlandırın.

Geçirilen kuyruk yöneticisi, bir YAML dosyası kullanarak Red Hat OpenShift ' a konuşlandırılır. Aşağıdaki örnek, önceki bölümlerde kullanılan adlara dayalıdır.

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: qm1mi
spec:
  version: 9.2.5.0-r3
  license:
    accept: true
    license: L-RJON-C7QG3S
    use: "Production"
  pki:
    keys:
      - name: default
        secret:
          secretName: my-tls-migration
          items:
            - tls.key
            - tls.crt
  web:
    enabled: true
  queueManager:
    name: QM1
    availability: MultiInstance
  storage:
    defaultClass: aws-efs
    persistedData:
      enabled: true
    queueManager:
      enabled: true
    recoveryLogs:
      enabled: true
  mqsc:
    - configMap:
        name: my-mqsc-migrated
      items:
        - backup.mqsc
```

Bu YAML ' nin bir açıklaması. Yapılandırmanın büyük bölümü, tek bir eşgörümlü kuyruk yöneticisinin konuşlandırılmasıyla aynı yaklaşımı izler; bu nedenle, burada yalnızca kuyruk yöneticisi kullanılabilirliği ve saklama yeri özellikleri açıklanır.

```
queueManager:
  name: QM1
  availability: MultiInstance
```

This specifies the queue manager name as QM1, and sets the deployment to be MultiInstance instead of the default single instance.

```
storage:
  defaultClass: aws-efs
  persistedData:
    enabled: true
  queueManager:
    enabled: true
```



```
recoveryLogs:
  enabled: true
```

IBM MQ çok eşgörunümlü bir kuyruk yöneticisi, RWX depolamaya bağlıdır. Varsayılan olarak, bir kuyruk yöneticisi tek bir yönetim ortamı kipinde konuşlandırılır ve bu nedenle çok yönetim ortamı kipine geçildiğinde ek saklama alanı seçenekleri de gereklidir. Önceki YAML örneğinde, üç depolama kalıcı birimi ve kalıcı bir birim sınıfı tanımlanır. Bu kalıcı birim sınıfının bir RWX depolama sınıfı olması gerekir. Ortamınızdaki depolama sınıfı adlarından emin değilseniz, bunları keşfetmek için aşağıdaki komutu çalıştırabilirsiniz:

```
oc get storageclass
```

Burada, bu komutun döndürdüğü örnek çıktı:

NAME	PROVISIONER	RECLAIMPOLICY
aws-efs	openshift.org/aws-efs	Delete
gp2 (default)	kubernetes.io/aws-ebs	Delete

Aşağıdaki kod, [IBM MQ yapılandırmasını içe aktar bölümünde](#), içe aktarılan IBM MQ yapılandırmasına nasıl başvurulacağı gösterilmektedir. Farklı adlar kullandıysanız, `my-mqsc-migrated` ve `backup.mqsc`' yi değiştirmeniz gerekir.

```
mqsc:
  - configMap:
      name: my-mqsc-migrated
    items:
      - backup.mqsc
```

Çoklu eşgörunüm kuyruk yöneticinizi konuşlandırdınız. Bu, şablonu tamamlar. Artık [yeni taşıyıcı konuşlandırmayı doğrulayın](#)' a hazırsınız.

OpenShift V 9.2.1 CD EUS Yeni taşıyıcı konuşlandırmasının doğrulanması

Now that IBM MQ is deployed on Red Hat OpenShift, you can verify the environment using the IBM MQ samples.

Başlamadan önce

Bu görev, [Red Hat OpenShift üzerinde yeni kuyruk yöneticisini yarattığınız varsayılar](#).

Önemli: Bu görev, kuyruk yöneticisinde TLS ' nin etkinleştirilmediğini varsayar.

Bu görev hakkında

In this task you run the IBM MQ samples from inside the migrated queue manager's container. Ancak, başka bir ortamdan çalışan kendi uygulamalarınızı kullanmayı tercih edebilirsiniz.

Aşağıdaki bilgilere gereksinim duyarsınız:

- LDAP Kullanıcı Adı
- LDAP Parolası
- IBM MQ Kanal adı
- Kuyruk adı

Bu örnek kod aşağıdaki ayarları kullanır. Ayarlarınızın farklı olacağını lütfen unutmayın.

- LDAP Kullanıcı Adı: mqapp
- LDAP Parolası: mqapp
- IBM MQ Kanal adı: DEV.APP.SVRCONN

- Kuyruk adı: Q1

Yordam

1. Çalışan IBM MQ taşıyıcısında dışa bağlantı sağlar.

Aşağıdaki komutu kullanın:

```
oc exec -it qm1-ibm-mq-0 /bin/bash
```

Burada qm1-ibm-mq-0 , “[Red Hat OpenShift üzerinde yeni kuyruk yöneticisi yaratılıyor](#)” sayfa 53 içinde yerleştirdiğimiz Pod 'dur. Devreye almayı farklı bir şey olarak çağırdıysanız, bu değeri özelleştirin.

2. Bir mesaj gönderin.

Aşağıdaki komutları çalıştırın:

```
cd /opt/mqm/samp/bin
export IBM MQSAMP_USER_ID=mqapp
export IBM MQSERVÉR=DEV.APP.SVRCONN/TCP/'localhost(1414)'
./amqsputc Q1 QM1
```

Bir parola girmeniz istenir, ardından bir ileti gönderebilirsiniz.

3. İletinin başarıyla alındığına dikkat edin.

GET örneğini çalıştırın:

```
./amqsgetc Q1 QM1
```

Sonuçlar

“[IBM MQ 'un IBM Cloud Pak for Integration' e geçirilmesi](#)” sayfa 36 işlemini tamamladınız.

Sonraki adım

Daha karmaşık geçiş senaryolarında size yardımcı olması için aşağıdaki bilgileri kullanın:

Kuyruğa alınan iletilerin geçirilmesi

Varolan kuyruğa alınmış iletileri yeni düzeye geçirmek için, yeni kuyruk yöneticisi yerine gelen iletileri dışa aktarmak ve içe aktarmak için aşağıdaki başlığı izleyin: [İki sistem arasında dmpmqmsg yardımcı programını kullanma](#).

Red Hat OpenShift ortamının dışından IBM MQ ile bağlantı kurulması

Konuşlandırılan kuyruk yöneticisi, Red Hat OpenShift ortamı dışında IBM MQ istemcilerine ve kuyruk yöneticilerine maruz kalınabilir. Bu işlem, Red Hat OpenShift ortamına bağlanmanın IBM MQ sürümüne bağlıdır. Bkz. “[Red Hat OpenShift kümesi dışından bir kuyruk yöneticisine bağlanmak için bir Rotanın yapılandırılması](#)” sayfa 107.

Red Hat OpenShift üzerinde IBM MQ Operator ürününün kurulması ve kaldırılması

IBM MQ Operator , Operator Hub olanağını kullanarak Red Hat OpenShift üzerine kurulabilir.

Yordam

- “[IBM MQ Operator bağımlılıkları](#)” sayfa 9.
- “[IBM MQ Operator için gerekli olan küme kapsamlı izinler](#)” sayfa 10.
- “[Installing the IBM MQ Operator using the Red Hat OpenShift web console](#)” sayfa 59.
- “[Installing the IBM MQ Operator using the Red Hat OpenShift CLI](#)” sayfa 61.

- “IBM MQ Operator ' in hava boşluğu ortamına kurulması” sayfa 65.

İlgili görevler

“Uninstalling the IBM MQ Operator using the Red Hat OpenShift web console” sayfa 60

IBM MQ Operator ürününü Red Hat OpenShiftolanağından kaldırmak için Red Hat OpenShift web konsolunu kullanabilirsiniz.

“Uninstalling the IBM MQ Operator using the Red Hat OpenShift CLI” sayfa 63

IBM MQ Operator CLI 'yı Red Hat OpenShiftolanağından kaldırmak için Red Hat OpenShift CLI' yı kullanabilirsiniz. Kaldırma işleminde, IBM MQ Operator ' un tek bir ad alanında kurulu olup olmadığına bağlı olarak ya da kümede kurulu ve tüm ad alanları için kullanılabilir olup olmadığına bağlı olarak, kaldırma işleminde farklılıklar vardır.

OpenShift CP4I Installing the IBM MQ Operator using the Red Hat OpenShift web console

IBM MQ Operator , Operator Hub olanağını kullanarak Red Hat OpenShift üzerine kurulabilir.

Başlamadan önce

Red Hat OpenShift küme web konsolunuzda oturum açın.

Yordam

1. EUS

İsteğe bağlı: Add the IBM Common Services Operators to the list of installable operators.

Not:

Bu adım, IBM MQ Operator 1.5 yayın düzeyleri ve önceki yayın düzeyleri için geçerlidir. Adım, ayrı bir Common Services kataloğunu ekler. İşleçten sonraki yayın düzeyleri için Common Services , IBM kataloğunun içinde yer alır.

- a) Ekranın sağ üst kısmındaki artı simgesini tıklatın. **YAML İçer Aktar** iletişim kutusunu görürsünüz.
- b) İletişim kutusuna aşağıdaki kaynak tanımlamasını yapıştırın.

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: opencloud-operators
  namespace: openshift-marketplace
spec:
  displayName: IBMCS Operators
  publisher: IBM
  sourceType: grpc
  image: icr.io/cpopen/ibm-common-service-catalog:latest
  updateStrategy:
    registryPoll:
      interval: 45m
```

- c) **Oluştur**'u tıklatın.

2. Kurulabilir işleçler listesine IBM işleçlerini ekleyin.

- a) Ekranın sağ üst kısmındaki artı simgesini tıklatın. **YAML İçer Aktar** iletişim kutusunu görürsünüz.
- b) İletişim kutusuna aşağıdaki kaynak tanımlamasını yapıştırın.

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: ibm-operator-catalog
  namespace: openshift-marketplace
spec:
  displayName: IBM Operator Catalog
  image: icr.io/cpopen/ibm-operator-catalog:latest
  publisher: IBM
  sourceType: grpc
  updateStrategy:
```

```
registryPoll:  
interval: 45m
```

- c) **Oluştur**'u tıklattın.
3. IBM MQ Operator için kullanmak üzere bir ad alanı oluşturun
IBM MQ Operator , tek bir ad alanına veya tüm ad alanlarına kapsamlı olarak kurulabilir. Bu adım, önceden var olmayan belirli bir ad alanına kuruluş yapmak istiyorsanız gereklidir.
 - a) Gezinme bölmesinden, **Ana Sayfa > Projeler** ' i tıklattın.
Projects (Projeler) sayfası görüntülenir.
 - b) **Proje Oluştur** ' u tıklattın. Bir Proje Yarat alanı görüntülenir.
 - c) Yaratmakta olduğunuz ad alanının ayrıntılarını girin. Örneğin, ad olarak "ibm-mq" belirtebilirsiniz.
 - d) **Oluştur**'u tıklattın. IBM MQ Operator ' nizi için ad alanı yaratılır.
4. IBM MQ Operator programını kurun.
 - a) Gezinme bölmesinden, **İşleçler > OperatorHub** seçeneklerini tıklattın.
OperatorHub sayfası görüntülenir.
 - b) **Tüm Öğeler** alanında "IBM MQ" değerini girin.
IBM MQ katalog girişi görüntülenir.
 - c) **IBM MQ** seçeneğini belirleyin.
IBM MQ penceresi görüntülenir.
 - d) **Kur**'u tıklattın.
İşletmen Aboneliği Oluştur sayfasını görürsünüz.
 - e) Seçilecek işletmen kanalını belirlemek için ["IBM MQ Operator için sürüm desteği"](#) sayfa 7 ögesini inceleyin.
 - f) Kuruluş Kipini oluşturduğunuz özel ad alanına ya da küme çapında kapsamı ayarlayın.
Farklı ad alanlarına bir İşleçten farklı sürümlerinin kurulması sorunlara yol açabileceğinden, küme genelindeki kapsamın seçilmesi önerilir. İşleçler, kontrol düzleminin uzantılarına sahip olmak üzere tasarlanmıştır.
 - g) **Abone Oldüğmesini** tıklattın.
Kurulu İşleçler sayfasında IBM MQ ' u göreceksiniz.
 - h) Kurulu İşleçler sayfasındaki İşletmen 'in durumunu denetleyin, kuruluş tamamlandığında durum, başarılı olacak şekilde değişir.

Sonraki adım

["Preparing your Red Hat OpenShift project for IBM MQ using the Red Hat OpenShift web console"](#) sayfa 79

Uninstalling the IBM MQ Operator using the Red Hat OpenShift web console

IBM MQ Operator ürününü Red Hat OpenShiftolanağından kaldırmak için Red Hat OpenShift web konsolunu kullanabilirsiniz.

Başlamadan önce

Red Hat OpenShift Cluster ürününüzün web konsolunuzda oturum açın.

IBM MQ Operator kümedeki tüm projelerde/ad alanlarına kurulduysa, kuyruk yöneticilerini silmek istediğiniz her proje için aşağıdaki yordamın 1-5 numaralı adımları yineleyin.

Yordam

1. **İşleçler > Kurulu İşleçler** öğelerini seçin.

2. **Proje** açılan listesinden bir proje seçin.
3. **IBM MQ** işlecini tıklatın.
4. Bu IBM MQ Operatortarafından yönetilen kuyruk yöneticilerini görüntülemek için **Kuyruk Yöneticileri** sekmesini tıklatın.
5. Bir ya da daha çok kuyruk yöneticisini silin.
Bu kuyruk yöneticilerinin çalışmaya devam etmesine rağmen, IBM MQ Operatorolmadan beklediği gibi çalışmayacaklarını unutmayın.
6. İsteğe bağlı: Uygunsa, kuyruk yöneticilerini silmek istediğiniz her proje için 1-5 arasındaki adımları yineleyin.
7. **İşleçler > Kurulu İşleçler'** e dönün.
8. **IBM MQ** işlecinin yanında, üç nokta menüsünü tıklatın ve **Kaldır İşleciseçeneğini** belirleyin.
9. Red Hat OpenShift Container Platform 4.7kullanıyorsanız, komut satırından doğrulayan web kancasını el ile silmeniz gerekebilir:

```
oc delete validatingwebhookconfiguration namespace.validator.queuemanagers.mq.ibm.com
```

OpenShift CP4I Installing the IBM MQ Operator using the Red Hat OpenShift CLI

IBM MQ Operator , Operator Hub olanağını kullanarak Red Hat OpenShift üzerine kurulabilir.

Başlamadan önce

Log into the Red Hat OpenShift command line interface (CLI) using **oc login**. Bu adımlar için bir küme yöneticisi olmanız gerekir.

Yordam

1. 

İsteğe bağlı: IBM Common Services Operatörleri için bir **CatalogSource** oluşturun.

Not:

Bu adım, IBM MQ Operator 1.5 yayın düzeyleri ve önceki yayın düzeyleri için geçerlidir. Adım, ayrı bir Common Services kataloğunu ekler. İşleçten sonraki yayın düzeyleri için Common Services , IBM kataloğunun içinde yer alır.

- a) **CatalogSource** kaynağını tanımlayan bir YAML dosyası oluşturun.

Aşağıdaki içerikle "operator-source-cs.yaml" adlı bir dosya oluşturun:

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: opencloud-operators
  namespace: openshift-marketplace
spec:
  displayName: IBMCS Operators
  publisher: IBM
  sourceType: grpc
  image: icr.io/cpopen/ibm-common-service-catalog:latest
  updateStrategy:
    registryPoll:
      interval: 45m
```

- b) Apply the **CatalogSource** to the server.

```
oc apply -f operator-source-cs.yaml -n openshift-marketplace
```

2. IBM İşleçleri için bir **CatalogSource** oluşturun

- a) **CatalogSource** kaynağını tanımlayan bir YAML dosyası yarat

Aşağıdaki içerikle "operator-source-ibm.yaml" adlı bir dosya oluşturun:

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: ibm-operator-catalog
  namespace: openshift-marketplace
spec:
  displayName: IBM Operator Catalog
  image: icr.io/cpopen/ibm-operator-catalog:latest
  publisher: IBM
  sourceType: grpc
  updateStrategy:
    registryPoll:
      interval: 45m
```

b) Apply the **CatalogSource** to the server.

```
oc apply -f operator-source-ibm.yaml -n openshift-marketplace
```

3. IBM MQ Operator için kullanılmak üzere bir ad alanı oluşturun

IBM MQ Operator , tek bir ad alanına veya tüm ad alanlarına kapsamlı olarak kurulabilir. Bu adım, önceden var olmayan belirli bir ad alanına kuruluş yapmak istiyorsanız gereklidir.

```
oc new-project ibm-mq
```

4. OperatorHub' dan küme için kullanılabilir olan İşleçlerin listesini görüntüleme

```
oc get packagemanifests -n openshift-marketplace
```

5. Desteklenen InstallModes ve kullanılabilir Kanalları doğrulamak için IBM MQ Operator ' i inceleyin.

```
oc describe packagemanifests ibm-mq -n openshift-marketplace
```

6. Bir **OperatorGroup** nesnesi YAML dosyası yaratır

OperatorGroup , **OperatorGroup** ile aynı ad alanındaki tüm İşleçler için gerekli olan RBAC erişiminin oluşturulacağı hedef ad alanlarını seçen bir OLM kaynağıdır.

The namespace to which you subscribe the Operator must have an **OperatorGroup** that matches the Operator's **InstallMode**, either the **AllNamespaces** or **SingleNamespace** mode. Kurmayı planladığınız İşletmen **AllNamespaces** ögesini kullanıyorsa, openshift-operators ad alanının zaten uygun bir **OperatorGroup** yerinde olması gerekir.

Ancak, Operator **SingleNamespace** kipini kullanıyorsa ve uygun bir **OperatorGroup** yerinde yoksa, bir tane yaratmanız gerekir.

a) Aşağıdaki içerikle "mq-operator-group.yaml" adlı bir dosya oluşturun:

```
apiVersion: operators.coreos.com/v1
kind: OperatorGroup
metadata:
  name: <operatorgroup_name>
  namespace: <namespace_name>
spec:
  targetNamespaces:
    - <namespace_name>
```

b) **OperatorGroup** nesnesini oluşturma

```
oc apply -f mq-operator-group.yaml
```

7. Bir ad alanını IBM MQ Operator' e abone olmak için bir **Subscription** nesnesi YAML dosyası oluşturun

a) Seçilecek işletmen kanalını belirlemek için "[IBM MQ Operator için sürüm desteği](#)" sayfa 7 ögesini inceleyin.

b) Create a file called "mq-sub.yaml" with the following contents, but changing **channel** to match the channel for the version of the IBM MQ Operator you want to install.

```
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: ibm-mq
  namespace: openshift-operators
spec:
  channel: <ibm-mq-operator-channel>
  name: ibm-mq
  source: ibm-operator-catalog
  sourceNamespace: openshift-marketplace
```

AllNamespaces **InstallMode** kullanımı için ad alanında **openshift-operators** değerini belirtin. Ters durumda, SingleNamespace **InstallMode** kullanımı için ilgili tek ad alanını belirtin. Note you should only change the **namespace** field, leaving the **sourceNamespace** field as is.

c) **Subscription** nesnesini oluşturma

```
oc apply -f mq-sub.yaml
```

8. İşlecin durumunu denetleyin

İşleç kurulumu başarılı olduğunda, pod durumu *Çalışıyor* olarak gösterilir. AllNamespaces **InstallMode** kullanımı için, ad alanı olarak **openshift-operators** değerini belirtin. Ters durumda, SingleNamespace **InstallMode** kullanımı için ilgili tek ad alanını belirtin.

```
oc get pods -n <namespace_name>
```

Sonraki adım

[“Preparing your Red Hat OpenShift project for IBM MQ using the Red Hat OpenShift CLI” sayfa 80](#)

Uninstalling the IBM MQ Operator using the Red Hat OpenShift CLI

IBM MQ Operator CLI 'yı Red Hat OpenShiftolanağından kaldırmak için Red Hat OpenShift CLI' yı kullanabilirsiniz. Kaldırma işleminde, IBM MQ Operator ' un tek bir ad alanında kurulu olup olmadığına bağlı olarak ya da kümede kurulu ve tüm ad alanları için kullanılabilir olup olmadığına bağlı olarak, kaldırma işleminde farklılıklar vardır.

Başlamadan önce

oc loginkomutunu kullanarak Red Hat OpenShift küğünüzde oturum açın.

Yordam

- IBM MQ Operator tek bir ad alanına kurulduysa, aşağıdaki alt adımları tamamlayın:

a) Doğru projede yer olduğunuzdan emin olun:

```
oc project <project_name>
```

b) Projeye kurulan kuyruk yöneticilerini görüntüle:

```
oc get qmgr
```

c) Bir ya da daha çok kuyruk yöneticisini sil:

```
oc delete qmgr <qmgr_name>
```

Bu kuyruk yöneticilerinin çalışmaya devam etmesine rağmen, IBM MQ Operator olmadan beklediği gibi çalışmayacaklarını unutmayın.

d) **ClusterServiceVersion** yönetim ortamlarını görüntüleyin:

```
oc get csv
```

e) IBM MQ **ClusterServiceVersion** u silin:

```
oc delete csv <ibm_mq_csv_name>
```

f) Abonelikleri görüntüle:

```
oc get subscription
```

g) Tüm abonelikleri sil:

```
oc delete subscription <ibm_mq_subscription_name>
```

h) İsteğe bağlı: Ortak hizmetler başka bir şey kullanmıyorsa, ortak hizmetler işlecini kaldırmak ve işletmen grubunu silmek isteyebilirsiniz:

a. Uninstall the common services operator, by following the instructions in [Ortak hizmetlerin kaldırılması](#) in the IBM Cloud Pak foundational services product documentation.

b. İşletmen grubunu görüntüle:

```
oc get operatorgroup
```

c. İşleç grubunu sil:

```
oc delete OperatorGroup <operator_group_name>
```

- IBM MQ Operator kuruluysa ve kümedeki tüm ad alanları için kullanılabilir durumda ise, aşağıdaki alt adımları tamamlayın:

a) Kurulu tüm kuyruk yöneticilerini görüntüle:

```
oc get qmgr -A
```

b) Bir ya da daha çok kuyruk yöneticisini sil:

```
oc delete qmgr <qmgr_name> -n <namespace_name>
```

Bu kuyruk yöneticilerinin çalışmaya devam etmesine rağmen, IBM MQ Operator olmadan beklediği gibi çalışmayacaklarını unutmayın.

c) **ClusterServiceVersion** yönetim ortamlarını görüntüleyin:

```
oc get csv -A
```

d) IBM MQ **ClusterServiceVersion** u kümeden silin:

```
oc delete csv <ibm_mq_csv_name> -n openshift-operators
```

e) Abonelikleri görüntüle:

```
oc get subscription -n openshift-operators
```

f) Abonelikleri sil:

```
oc delete subscription <ibm_mq_subscription_name> -n openshift-operators
```

g) Red Hat OpenShift Container Platform 4.7 kullanıyorsanız, web kancası doğrulamayı el ile silmeniz gerekebilir:

```
oc delete validatingwebhookconfiguration namespace.validator.queuemanagers.mq.ibm.com
```

h) İsteğe bağlı: Ortak hizmetler başka bir şey kullanmıyorsa, ortak hizmetler işletmenini kaldırmak isteyebilirsiniz:

IBM Cloud Pak foundational services ürün belgelerindeki [Uninstalling common services](#) (Ortak hizmetlerin kaldırılması) başlıklı konu yönergelerini izleyin.

kurulması

This tutorial guides you in installing IBM MQ Operator into a Red Hat OpenShift cluster that has no internet connectivity. You can install IBM MQ Operator in an airgap environment using a portable storage device, or using a bastion machine.

Installing the IBM MQ Operator in an airgap environment using a portable storage device

Kuruluşu tamamlamaya ilişkin adımlar için, IBM Cloud Pak for Integration belgelerinde [Mirroring images with a portable storage device](#) başlıklı konuya bakın. Yalnızca IBM MQ kuruyorsanız, aşağıdaki ortam değişkenlerinin tüm oluşturmalarını burada verilen değerlerle değiştirin:

```
export CASE_NAME=ibm-mq
export CASE_ARCHIVE_VERSION=version_number
export CASE_INVENTORY_SETUP=ibmMQOperator
```

Burada *version_number* , hava boşluğu kuruluşunu yapmak için kullanmak istediğiniz vakanın sürümüdür. Kullanılabilir vaka sürümlerine ilişkin bir liste için bkz. <https://github.com/IBM/cloud-pak/tree/master/repo/case/ibm-mq>. Seçilecek işletmen kanalını belirlemek için [“IBM MQ Operator için sürüm desteği” sayfa 7](#) ögesini inceleyin.

Installing the IBM MQ Operator in an airgap environment using a bastion machine

1. [“Önkoşullar” sayfa 65](#)
2. [“Docker kaydı hazırlama” sayfa 65](#)
3. [“Bir basma anasistemi hazırlayın” sayfa 66](#)
4. [“Kuruluş programı ve görüntü dökümü için ortam değişkenleri yarat” sayfa 67](#)
5. [“IBM MQ kuruluş programını ve görüntü dökümünü yükleyin” sayfa 67](#)
6. [“Log in to the Red Hat OpenShift Container Platform cluster as a cluster administrator” sayfa 67](#)
7. [“IBM MQ Operator için bir Kubernetes ad alanı oluşturun” sayfa 68](#)
8. [“Görüntüleri ikizle ve kümeyi yapılandır” sayfa 68](#)
9. [“IBM MQ Operator programını kurun.” sayfa 70](#)
10. [“Deploy IBM MQ Queue Manager” sayfa 70](#)

Önkoşullar

1. Bir Red Hat OpenShift Container Platform kümesinin kurulması gerekir. Desteklenen Red Hat OpenShift Container Platform sürümleri için bkz. [“IBM MQ Operator için sürüm desteği” sayfa 7](#).
2. Bir Docker kaydı kullanılabilir olmalıdır. Daha fazla bilgi için bkz [“Docker kaydı hazırlama” sayfa 65](#).
3. Bir basma sunucusu yapılandırılmalıdır. Daha fazla bilgi için bkz [“Bir basma anasistemi hazırlayın” sayfa 66](#).

Docker kaydı hazırlama

Yerel ortamınızdaki tüm görüntüleri saklamak için yerel bir Docker kayıt dosyası kullanılır. Böyle bir kayıt dosyası oluşturmalı ve bunun aşağıdaki gereksinimleri karşıladığından emin olmanız gerekir:

- [Docker Manifest V2, Şema 2](#)' yi destekler.
- Çok mimarlı görüntüleri destekler.
- Hem bastion sunucusundan hem de Red Hat OpenShift Container Platform küme düğümlerinden erişilebilir.

- Bastion anasisteminden hedef kayıt dosyasına yazabilen bir kullanıcının kullanıcı adı ve parolasına sahip olmalıdır.
- Red Hat OpenShift küme düğümlerinde bulunan hedef kayıt defterinden okuyabilen bir kullanıcının kullanıcı adı ve parolasına sahip olmalıdır.
- Resim adında yol ayırıcılarına izin verir.

Docker kaydını oluşturduktan sonra, kayıt defterini yapılandırmanız gerekir:

1. Kayıt ad alanları yarat

- `ibmcom` -Tüm resimleri `dockerhub.io/ibmcom` ad alanından depolamak için ad alanı.
`ibmcom` ad alanı, genel kullanıma açık olan ve kimlik bilgilerinin çekilmesini gerektirmeyen tüm IBM görüntüleri içindir.
- `cp` - IBM görüntülerini `cp.icr.io/cp` havuzundan depolamak için ad alanı.
`cp` ad alanı, ürün yetkisi anahtarı ve çekilecek kimlik bilgileri gerektiren IBM Yetkili Kayıt Defterinde yer alan resimlerin içindir. Yetki anahtarınızı almak için yetkili yazılımla ilişkili IBM kimliği ve parolası ile [MyIBM Container Software Library](#) ' de oturum açın. **Yetki anahtarları** bölümünde, yetki anahtarını panoya kopyalamak için **Copy key** (Anahtar kopyala) seçeneğini belirleyin, daha sonra aşağıdaki adımlarda kullanmak üzere saklayın.
- `opencloudio` -Resimlerin `quay.io/opencloudio`' den saklanacak ad alanı.
The `opencloudio` namespace is for select IBM open source component images that are available on [quay.io](#). IBM Cloud Pak foundational services resimleri, `opencloudio` üzerinde barındırılır.

2. Her bir ad alanının aşağıdaki gereksinimleri karşıladığını doğrulayın:

- Otomatik havuz oluşturmayı destekler.
- Havuz yazabilen ve havuz oluşturabilen bir kullanıcının kimlik bilgilerine sahip. Basma anasistemi bu kimlik bilgilerini kullanır.
- Tüm havuzları okuyabilen bir kullanıcının kimlik bilgileri var. Red Hat OpenShift Container Platform kümesi bu kimlik bilgilerini kullanır.

Bir basma anasistemi hazırlayın

Red Hat OpenShift Container Platform kümesine, yerel Docker kaydına ve Internet 'e erişebilen bir basma anasistemi hazırlayın. Bastion anasistemi, IBM Cloud Pak CLI ve Red Hat OpenShift Container Platform CLI desteğinin bulunduğu herhangi bir işletim sistemine sahip bir Linux for x86-64 platformunda olmalıdır.

Bastion düğümünüzde bu adımları tamamlayın:

1. OpenSSL sürüm 1.11.1 ya da sonraki sürümünü kurun.
2. Bastion düğümüne Docker ya da Podman ' ı kurun.

- Docker' u kurmak için şu komutları çalıştırın:

```
yum check-update
yum install docker
```

- Podman' ı kurmak için bkz. [Podman Kuruluş Yönergeleri](#)

3. Bastion düğümüne skeopo sürümünü 1.x.x kurun. Skropeo programını kurmak için şu komutları çalıştırın:

```
yum check-update
yum install skopeo
```

4. IBM Cloud Pak CLI ' yı kurun. Altyapınıza ilişkin ikili dosyanın en son sürümünü kurun. Daha fazla bilgi için bkz. [cloud-pak-cli](#).

- a. İkili dosyayı karşıdan yükleyin.

```
wget https://github.com/IBM/cloud-pak-cli/releases/download/vversion-number/binary-file-name
```

Örneğin:

```
wget https://github.com/IBM/cloud-pak-cli/releases/latest/download/cloudctl-linux-amd64.tar.gz
```

b. İkili dosyayı açın.

```
tar -xf binary-file-name
```

c. Dosyayı değiştirmek ve taşımak için aşağıdaki komutları çalıştırın:

```
chmod 755 file-name  
mv file-name /usr/local/bin/cloudctl
```

d. cloudctl 'in kurulu olduğunu onaylayın:

```
cloudctl --help
```

5. oc Red Hat OpenShift Container Platform CLI aracını kurun.

Daha fazla bilgi için bkz. [Red Hat OpenShift Container Platform CLI araçları](#)

6. Çevrimdışı mağaza olarak hizmet veren bir dizin oluşturun.

Aşağıda örnek bir dizin yer almaktadır. Bu örnek sonraki adımlarda kullanılır.

```
mkdir $HOME/offline
```

Not: Bu çevrimdışı depo, verileri bir kereden fazla aktarmamak için kalıcı olmalıdır. Kalıcılık aynı zamanda, ikizleme işlemini birden çok kez ya da bir zaman çizelgesi üzerinde çalıştırılmasına yardımcı olur.

Kuruluş programı ve görüntü dökümü için ortam değişkenleri yarat

Kuruluş programı görüntü adı ve görüntü dökümü ile aşağıdaki ortam değişkenlerini oluşturun:

```
export CASE_ARCHIVE_VERSION=version_number  
export CASE_ARCHIVE=ibm-mq-$CASE_ARCHIVE_VERSION.tgz  
export CASE_INVENTORY=ibmMQoperator
```

Burada *sürüm_numarası* , hava boşluğu kuruluşunu gerçekleştirmek için kullanmak istediğiniz vakanın sürümüdür. Kullanılabilir vaka sürümlerinin bir listesi için bkz. <https://github.com/IBM/cloud-pak/tree/master/repo/case/ibm-mq>. Hangi işletmen kanalının seçileceğini belirlemek için [IBM MQ Operator için sürüm desteği](#) başlıklı konuya bakın.

IBM MQ kuruluş programını ve görüntü dökümünü yükleyin

ibm-mq kuruluş programını ve görüntü dökümünü bastion anasistemine yükleyin:

```
cloudctl case save \  
--case https://github.com/IBM/cloud-pak/raw/master/repo/case/ibm-mq/$CASE_ARCHIVE_VERSION/  
$CASE_ARCHIVE \  
--outputdir $HOME/offline/
```

Log in to the Red Hat OpenShift Container Platform cluster as a cluster administrator

Aşağıda, Red Hat OpenShift Container Platform kümesinde oturum açmak için örnek bir komut verilmiştir:

```
oc login cluster_host:port --username=cluster_admin_user --password=cluster_admin_password
```

IBM MQ Operator için bir Kubernetes ad alanı oluşturun

IBM MQ Operator'ı kurabilmek için bir ad alanı içeren bir ortam değişkeni yaratın ve ad alanını yaratın:

```
export NAMESPACE=ibm-mq-test
oc create namespace ${NAMESPACE}
```

Görüntüleri ikizle ve kümeyi yapılandır

Görüntüleri ikizlemek ve kümenizi yapılandırmak için bu adımları tamamlayın:

Not: Herhangi bir komutta çift tırnak işareti içinde tilde kullanmayın. Örneğin, `args "--registry registry --user registry_userid --pass registry_password --inputDir ~/offline"` kullanmayın. Tilde genişlemez ve komutlarınız başarısız olabilir.

1. Tüm kaynak Docker kayıtları için kimlik doğrulama kimlik bilgilerini depolayın.

Tüm IBM Cloud Platform Common Services, IBM MQ Operator görüntüsü ve IBM MQ Advanced Developer görüntüsü, kimlik doğrulaması gerektirmeyen genel kayıt defterlerinde saklanır. Ancak, IBM MQ Advanced Server (Geliştirici olmayan), diğer ürünler ve üçüncü kişi bileşenleri için bir ya da daha fazla kimlik doğrulaması yapılmış kayıt dosyası gerekir. Aşağıdaki kayıt dosyaları kimlik doğrulaması gerektiriyor:

- cp.icr.io
- registry.redhat.io
- registry.access.redhat.com

Bu kayıtlar hakkında daha fazla bilgi için bkz. [Kayıt dosyası ad alanları yaratılması](#).

Kimlik doğrulaması gerektiren tüm kayıt dosyaları için kimlik bilgilerini yapılandırmak üzere aşağıdaki komutu çalıştırmanız gerekir. Komutu, bu tür her kayıt dosyası için ayrı ayrı çalıştırın:

```
cloudctl case launch \
--case $HOME/offline/${CASE_ARCHIVE} \
--inventory ${CASE_INVENTORY} \
--action configure-creds-airgap \
--namespace ${NAMESPACE} \
--args "--registry registry --user registry_userid --pass registry_password --inputDir $HOME/offline"
```

Komut, kayıt dosyası kimlik bilgilerini `$HOME/.airgap/secrets` konumundaki dosya sisteminizdeki bir dosyada saklar ve önbelleğe alır.

2. Yerel Docker kayıt defteri bağlantı bilgileriyle ortam değişkenleri oluşturun.

```
export LOCAL_DOCKER_REGISTRY=IP_or_FQDN_of_local_docker_registry
export LOCAL_DOCKER_USER=username
export LOCAL_DOCKER_PASSWORD=password
```

Not: Docker kayıt dosyası 80 ya da 443 gibi standart kapıları kullanır. Docker kayıt defteriniz standart olmayan bir kapı kullanıyorsa, `host:ports` sözdizimini kullanarak kapıyı belirtin. Örneğin:

```
export LOCAL_DOCKER_REGISTRY=myregistry.local:5000
```

3. Yerel Docker kaydı için bir kimlik doğrulama güvenlik dizgisi yapılandırın.

Not: Bu adımın yalnızca bir kez yapılması gerekir.

```
cloudctl case launch \
--case $HOME/offline/${CASE_ARCHIVE} \
--inventory ${CASE_INVENTORY} \
--action configure-creds-airgap \
--namespace ${NAMESPACE} \
--args "--registry ${LOCAL_DOCKER_REGISTRY} --user ${LOCAL_DOCKER_USER} --pass ${LOCAL_DOCKER_PASSWORD}"
```

Komut, kayıt dosyası kimlik bilgilerini \$HOME/.airgap/secrets konumundaki dosya sisteminizdeki bir dosyada saklar ve ön belleğe alır.

4. Genel bir görüntü çekme güvenlik dizgisini ve **ImageContentSourcePolicy**'i yapılandırın.

a. Düğümü yeniden başlatmanın gerekli olup olmadığını denetleyin.

- Red Hat OpenShift Container Platform sürüm 4.4 ve sonraki sürümlerde ve hava boşluğu kullanılarak yeni bir IBM MQ Operator kuruluşunda bu adım, tüm küme düğümlerini yeniden başlatır. Yeni çekme güvenlik dizgisi uygulanıncaya kadar küme kaynakları kullanılamıyor olabilir.
- IBM MQ Operator 1.8' de CASE , görüntülere ilişkin ek bir ikizleme kaynağı içerecek şekilde güncellenir. Bu nedenle, önceki IBM MQ Operator sürümlerinden 1.8 sürümüne ya da üstüne yükseltme yaptığınızda, bir düğüm yeniden başlatma tetiklenir.
- Bu adımın düğüm yeniden başlatması gerekip gerekmediğini denetlemek için bu adımın koduna `--dry-run` seçeneğini ekleyin. Bu, en son **ImageContentSourcePolicy** 'i oluşturur ve konsol penceresinde (**stdout**) görüntüler. Bu **ImageContentSourcePolicy** yapılandırılan kümeden farklıysa **ImageContentSourcePolicy**, yeniden başlatma gerçekleşir.

```
cloudctl case launch \  
--case $HOME/offline/${CASE_ARCHIVE} \  
--inventory ${CASE_INVENTORY} \  
--action configure-cluster-airgap \  
--namespace ${NAMESPACE} \  
--args "--registry ${LOCAL_DOCKER_REGISTRY} --user ${LOCAL_DOCKER_USER} --pass $  
{LOCAL_DOCKER_PASSWORD} --inputDir $HOME/offline --dryRun"
```

b. Genel resim çekme güvenlik dizgisini ve **ImageContentSourcePolicy**'i yapılandırmak için, bu adımın kodunu `--dry-run` seçeneği olmadan çalıştırın:

```
cloudctl case launch \  
--case $HOME/offline/${CASE_ARCHIVE} \  
--inventory ${CASE_INVENTORY} \  
--action configure-cluster-airgap \  
--namespace ${NAMESPACE} \  
--args "--registry ${LOCAL_DOCKER_REGISTRY} --user ${LOCAL_DOCKER_USER} --pass $  
{LOCAL_DOCKER_PASSWORD} --inputDir $HOME/offline"
```

5. **ImageContentSourcePolicy** kaynağının yaratıldığını doğrulayın.

```
oc get imageContentSourcePolicy
```

6. İsteğe bağlı: Güvenli olmayan bir kayıt dosyası kullanıyorsanız, yerel kaydı küme **insecureRegistries** listesine eklemeniz gerekir.

```
oc patch image.config.openshift.io/cluster --type=merge -p '{"spec":{"registrySources":  
{"insecureRegistries":["${LOCAL_DOCKER_REGISTRY}"]}}'
```

7. Küme düğümü durumunuzu doğrulayın.

```
oc get nodes
```

imageContentsourcePolicy ve genel resim çekme güvenlik dizgisi uygulandıktan sonra, düğüm durumunu **Ready**, **Scheduling** ya da **Disabled** olarak görebilirsiniz. Tüm düğümlerin **Ready** durumunu göstermesini bekleyin.

8. Görüntüleri yerel kayda imolur ve yapmak ya ya ya ya) izoluşturmanızı sağlar.

```
cloudctl case launch \  
--case $HOME/offline/${CASE_ARCHIVE} \  
--inventory ${CASE_INVENTORY} \  
--action mirror-images \  
--namespace ${NAMESPACE} \  
--args "--registry ${LOCAL_DOCKER_REGISTRY} --user ${LOCAL_DOCKER_USER} --pass $  
{LOCAL_DOCKER_PASSWORD} --inputDir $HOME/offline"
```

IBM MQ Operatorprogramını kurun.

1. Red Hat OpenShift küme web konsolunuzda oturum açın.
2. Bir katalog kaynağı yaratın. Önceki adımları yürüten aynı uçbirimi kullanın.

```
cloudctl case launch \  
--case $HOME/offline/${CASE_ARCHIVE} \  
--inventory ${CASE_INVENTORY} \  
--action install-catalog \  
--namespace ${NAMESPACE} \  
--args "--registry ${LOCAL_DOCKER_REGISTRY} --recursive"
```

3. Verify that the **CatalogSource** is created for the Common Services Installer Operator.

```
oc get pods -n openshift-marketplace  
oc get catalogsource -n openshift-marketplace
```

4. IBM MQ Operator komutunu OLMkullanarak kurun.

- a. Gezinme bölmesinden, **İşleçler** > **OperatorHub**seçeneklerini tıklatın.

OperatorHub sayfası görüntülenir.

- b. **Tüm Öğeler** alanına IBM MQgirin.

IBM MQ katalog girişi görüntülenir.

- c. **IBM MQ**seçeneğini belirleyin.

IBM MQ penceresi görüntülenir.

- d. **Kur**'u tıklatın.

İşletmen Aboneliği Oluştur sayfası görüntülenir.

- e. Seçilecek işletmen kanalını belirlemek için [“IBM MQ Operatoriçin sürüm desteği” sayfa 7](#) ögesini inceleyin.

- f. **Kuruluş Kipi** ' u oluşturduğunuz özel ad alanına ya da küme geniş kapsamına ayarlayın.

- g. **Abone Oldüğmesini** tıklatın.

IBM MQ , Installed Operator (Kurulu İşleçler) sayfasına eklenir.

- h. **Kurulu İşleçler** sayfasındaki operatörün durumunu denetleyin. Kuruluş tamamlandığında durum **Succeeded** olarak değişir.

Deploy IBM MQ Queue Manager

Kurulu işletmen altında yeni bir kuyruk yöneticisi yaratmak için [“Deploying and configuring queue managers using the IBM MQ Operator” sayfa 78](#)konusuna bakın.

İlgili görevler

[“IBM MQ Operator ya da kuyruk yöneticisini bir hava boşluğu ortamında yükseltme hazırlığı yapıyor” sayfa 71](#)

İnternet bağlantısı olmayan bir Red Hat OpenShift kümesinde, IBM MQ Operator' ı büyütmeden önce yapmanız gereken hazırlık adımları vardır.

OpenShift CP4I IBM MQ Operator ve kuyruk yöneticilerinin yükseltilmesi

IBM MQ Operator ' ın yükseltilmesi, kuyruk yöneticilerinizi yükseltmenize olanak sağlar.

Yordam

- [“Red Hat OpenShift web konsolunu kullanarak IBM MQ Operator yükseltiyor” sayfa 74.](#)
- [“Upgrading the IBM MQ Operator using the Red Hat OpenShift CLI” sayfa 75.](#)
- [“Red Hat OpenShift web konsolu kullanılarak bir IBM MQ kuyruk yöneticisinin yükseltilmesi” sayfa 76.](#)

- “Red Hat OpenShift CLI kullanılarak bir IBM MQ kuyruk yöneticisinin yükseltilmesi” sayfa 77.

OpenShift CP4I Linux IBM MQ Operator ya da kuyruk yöneticisini bir hava boşluğu ortamında yükseltme hazırlığı yapılıyor

İnternet bağlantısı olmayan bir Red Hat OpenShift kümesinde, IBM MQ Operator' ı büyütmeden önce yapmanız gereken hazırlık adımları vardır.

Başlamadan önce

Bu konu, daha önce yayınlanan IBM Cloud Pak for Integration görüntülerinin ikizlendiği yerel bir görüntü kaydı yapılandırıldığını varsayar.

Bu görev hakkında

IBM MQ Operator ya da kuyruk yöneticisini bir hava boşluğu ortamında büyütmeden önce en son IBM Cloud Pak for Integration görüntülerini ikizlemeniz gerekir.

Bu görevdeki ilk dört adımın, “IBM MQ Operator ' ın hava boşluğu ortamına kurulması” sayfa 65 ile aynı adımla aynı olduğunu unutmayın.

Yordam

1. Kuruluş programı ve görüntü dökümü için ortam değişkenleri yaratın.

Kuruluş programı görüntü adı ve görüntü dökümü ile aşağıdaki ortam değişkenlerini oluşturun:

```
export CASE_ARCHIVE_VERSION=version_number
export CASE_ARCHIVE=ibm-mq-CASE_ARCHIVE_VERSION.tgz
export CASE_INVENTORY=ibmMQoperator
```

Burada *sürüm_numarası* , hava boşluğu kuruluşunu gerçekleştirmek için kullanmak istediğiniz vakanın sürümüdür. Kullanılabilir vaka sürümlerinin bir listesi için bkz. <https://github.com/IBM/cloud-pak/tree/master/repo/case/ibm-mq>. Hangi işletmen kanalının seçileceğini belirlemek için [IBM MQ Operator için sürüm desteği](#) başlıklı konuya bakın.

2. IBM MQ kuruluş programını ve görüntü dökümünü yükleyin.

ibm-mq kuruluş programını ve görüntü dökümünü bastion anasistemine yükleyin:

```
cloudctl case save \
  --case https://github.com/IBM/cloud-pak/raw/master/repo/case/ibm-mq/
  $CASE_ARCHIVE_VERSION/$CASE_ARCHIVE \
  --outputdir $HOME/offline/
```

3. Red Hat OpenShift Container Platform kümesinde küme yöneticisi olarak oturum açın.

Aşağıda, Red Hat OpenShift Container Platform kümesinde oturum açmak için örnek bir komut verilmiştir:

```
oc login cluster_host:port --username=cluster_admin_user --password=cluster_admin_password
```

4. Görüntüleri ikizle ve kümeyi yapılandırın.

Görüntüleri ikizlemek ve kümenizi yapılandırmak için bu adımları tamamlayın:

Not: Herhangi bir komutta çift tırnak işareti içinde tilde kullanmayın. Örneğin, `args "--registry registry --user registry_userid --pass registry_password --inputDir ~/offline"` kullanmayın. Tilde genişlemez ve komutlarınız başarısız olabilir.

- a. Tüm kaynak Docker kayıtları için kimlik doğrulama kimlik bilgilerini depolayın.

Tüm IBM Cloud Platform Common Services, IBM MQ Operator görüntüsü ve IBM MQ Advanced Developer görüntüsü, kimlik doğrulaması gerektirmeyen genel kayıt defterlerinde saklanır. Ancak, IBM MQ Advanced Server (Geliştirici olmayan), diğer ürünler ve üçüncü kişi bileşenleri için bir ya

da daha fazla kimlik doğrulaması yapılmış kayıt dosyası gerekir. Aşağıdaki kayıt dosyaları kimlik doğrulaması gerektiriyor:

- cp.icr.io
- registry.redhat.io
- registry.access.redhat.com

Bu kayıtlar hakkında daha fazla bilgi için bkz. [Kayıt dosyası ad alanları yaratılması](#).

Kimlik doğrulaması gerektiren tüm kayıt dosyaları için kimlik bilgilerini yapılandırmak üzere aşağıdaki komutu çalıştırmanız gerekir. Komutu, bu tür her kayıt dosyası için ayrı ayrı çalıştırın:

```
cloudctl case launch \  
--case $HOME/offline/${CASE_ARCHIVE} \  
--inventory ${CASE_INVENTORY} \  
--action configure-creds-airgap \  
--namespace ${NAMESPACE} \  
--args "--registry registry --user registry_userid --pass registry_password --inputDir $HOME/offline"
```

Komut, kayıt dosyası kimlik bilgilerini \$HOME/.airgap/secrets konumundaki dosya sisteminizdeki bir dosyada saklar ve önbelleğe alır.

b. Yerel Docker kayıt defteri bağlantı bilgileriyle ortam değişkenleri oluşturun.

```
export LOCAL_DOCKER_REGISTRY=IP_or_FQDN_of_local_docker_registry  
export LOCAL_DOCKER_USER=username  
export LOCAL_DOCKER_PASSWORD=password
```

Not: Docker kayıt dosyası 80 ya da 443 gibi standart kapıları kullanır. Docker kayıt defteriniz standart olmayan bir kapı kullanıyorsa, *host:ports* sözdizimini kullanarak kapıyı belirtin. Örneğin:

```
export LOCAL_DOCKER_REGISTRY=myregistry.local:5000
```

c. Yerel Docker kaydı için bir kimlik doğrulama güvenlik dizgisi yapılandırın.

Not: Bu adımın yalnızca bir kez yapılması gerekir.

```
cloudctl case launch \  
--case $HOME/offline/${CASE_ARCHIVE} \  
--inventory ${CASE_INVENTORY} \  
--action configure-creds-airgap \  
--namespace ${NAMESPACE} \  
--args "--registry ${LOCAL_DOCKER_REGISTRY} --user ${LOCAL_DOCKER_USER} --pass $  
{LOCAL_DOCKER_PASSWORD}"
```

Komut, kayıt dosyası kimlik bilgilerini \$HOME/.airgap/secrets konumundaki dosya sisteminizdeki bir dosyada saklar ve önbelleğe alır.

d. Genel bir görüntü çekme güvenlik dizgisini ve **ImageContentSourcePolicy**' i yapılandırın.

i) Düğümü yeniden başlatmanın gerekli olup olmadığını denetleyin.

- Red Hat OpenShift Container Platform sürüm 4.4 ve sonraki sürümlerde ve hava boşluğu kullanılarak yeni bir IBM MQ Operator kuruluşunda bu adım, tüm küme düğümlerini yeniden başlatır. Yeni çekme güvenlik dizgisi uygulanıncaya kadar küme kaynakları kullanılamıyor olabilir.
- IBM MQ Operator 1.8' de CASE , görüntülere ilişkin ek bir ikizleme kaynağı içerecek şekilde güncellenir. Bu nedenle, önceki IBM MQ Operator sürümlerinden 1.8 sürümüne ya da üstüne yükseltme yaptığınızda, bir düğüm yeniden başlatma tetiklenir.
- Bu adımın düğüm yeniden başlatması gerekip gerekmediğini denetlemek için bu adımın koduna --dry-run seçeneğini ekleyin. Bu, en son **ImageContentSourcePolicy** ' i oluşturur ve konsol penceresinde (**stdout**) görüntüler. Bu **ImageContentSourcePolicy**

yapılandırılan kümeden farklıysa **ImageContentSourcePolicy**, yeniden başlatma gerçekleşir.

```
cloudctl case launch \  
--case $HOME/offline/${CASE_ARCHIVE} \  
--inventory ${CASE_INVENTORY} \  
--action configure-cluster-airgap \  
--namespace ${NAMESPACE} \  
--args "--registry ${LOCAL_DOCKER_REGISTRY} --user ${LOCAL_DOCKER_USER} --pass $  
{LOCAL_DOCKER_PASSWORD} --inputDir $HOME/offline --dryRun"
```

ii) Genel resim çekme güvenlik dizgisini ve **ImageContentSourcePolicy**' i yapılandırmak için, bu adımın kodunu `--dry-run` seçeneği olmadan çalıştırın:

```
cloudctl case launch \  
--case $HOME/offline/${CASE_ARCHIVE} \  
--inventory ${CASE_INVENTORY} \  
--action configure-cluster-airgap \  
--namespace ${NAMESPACE} \  
--args "--registry ${LOCAL_DOCKER_REGISTRY} --user ${LOCAL_DOCKER_USER} --pass $  
{LOCAL_DOCKER_PASSWORD} --inputDir $HOME/offline"
```

e. **ImageContentSourcePolicy** kaynağının yaratıldığını doğrulayın.

```
oc get imageContentSourcePolicy
```

f. İsteğe bağlı: Güvenli olmayan bir kayıt dosyası kullanıyorsanız, yerel kaydı küme **insecureRegistries** listesine eklemeniz gerekir.

```
oc patch image.config.openshift.io/cluster --type=merge -p '{"spec":{"registrySources":  
{"insecureRegistries":["${LOCAL_DOCKER_REGISTRY}"]}}'
```

g. Küme düğümü durumunuzu doğrulayın.

```
oc get nodes
```

imageContentsourcePolicy ve genel resim çekme güvenlik dizgisi uygulandıktan sonra, düğüm durumunu **Ready, Scheduling** ya da **Disabled** olarak görebilirsiniz. Tüm düğümlerin **Ready** durumunu göstermesini bekleyin.

h. Görüntüleri yerel kayda imolur ve yapmak ya ya ya ya) izoluşturmanızı sağlar.

```
cloudctl case launch \  
--case $HOME/offline/${CASE_ARCHIVE} \  
--inventory ${CASE_INVENTORY} \  
--action mirror-images \  
--namespace ${NAMESPACE} \  
--args "--registry ${LOCAL_DOCKER_REGISTRY} --user ${LOCAL_DOCKER_USER} --pass $  
{LOCAL_DOCKER_PASSWORD} --inputDir $HOME/offline"
```

5. Katalog kaynağını büyütün.

Önceki adımları yürüten aynı uçbirimi kullanın.

```
cloudctl case launch \  
--case $HOME/offline/${CASE_ARCHIVE} \  
--inventory ${CASE_INVENTORY} \  
--action install-catalog \  
--namespace ${NAMESPACE} \  
--args "--registry ${LOCAL_DOCKER_REGISTRY} --recursive"
```

Sonraki adım

Şimdi, aşağıdaki görevlerden birini tamamlayarak IBM MQ Operator ve kuyruk yöneticisini yükseltmeye hazırsınız:

- [“Red Hat OpenShift web konsolunu kullanarak IBM MQ Operator yükseltiliyor” sayfa 74](#)
- [“Upgrading the IBM MQ Operator using the Red Hat OpenShift CLI” sayfa 75](#)
- [“Red Hat OpenShift web konsolu kullanılarak bir IBM MQ kuyruk yöneticisinin yükseltilmesi” sayfa 76](#)

- “Red Hat OpenShift CLI kullanılarak bir IBM MQ kuyruk yöneticisinin yükseltilmesi” sayfa 77
- “Platform Navigator'ı kullanarak Red Hat OpenShift ' ta bir IBM MQ kuyruk yöneticisinin yükseltilmesi” sayfa 78

Red Hat OpenShift web konsolunu kullanarak IBM MQ Operator yükseltiyor

IBM MQ Operator , Operator Hub kullanılarak yükseltilebilir.

Başlamadan önce

Red Hat OpenShift küme web konsolunuzda oturum açın.

IBM MQ Operator ürününü hava boşluğu ortamında büyütmeden önce en son IBM Cloud Pak for Integration görüntülerini ikizlemeniz gerekir. Bkz. [IBM MQ Operator ya da kuyruk yöneticisini hava boşluğu ortamında büyütme hazırlık](#).

Yordam

1. Yükseltilecek işletmen kanalını belirlemek için [“IBM MQ Operator için sürüm desteği” sayfa 7 ' i](#) inceleyin.
2. İsteğe bağlı: If you are upgrading from a version of IBM MQ Operator that is older than 1.5 to IBM MQ Operator 1.5 or later, you must first upgrade the version of IBM Cloud Pak foundational services.
Daha fazla bilgi için bkz [“Upgrading IBM Cloud Pak foundational services using the Red Hat OpenShift web console” sayfa 74](#).
3. IBM MQ Operator ürününü yükseltin. Yeni ana ya da ikincil IBM MQ Operator sürümleri, yeni Abonelik Kanalları aracılığıyla teslim edilir. İşlecinizi yeni bir ana ya da ikincil sürüme yükseltmek için, IBM MQ Operator Aboneliğinizde seçilen kanalı güncellemeniz gerekir.
 - a) Gezinme bölmesinden, **İşleçler** > **Kurulu İşleçler** öğelerini tıklatın.
Belirtilen projeden kurulu tüm Operatörler görüntülenir.
 - b) **IBM MQ İşleç** seçeneğini belirleyin.
 - c) **Abonelik** sekmesine gidin
 - d) **Kanal** ' ı tıklatın.
Abonelik Güncelleme Kanalını Değiştir penceresi görüntülenir.
 - e) İstedığınız kanalı seçin ve **Save**(Kaydet) düğmesini tıklatın.
İşleç, yeni kanala uygun en son sürüme yükselir. Bkz. [“IBM MQ Operator için sürüm desteği” sayfa 7](#).

Sonraki adım

IBM Cloud Pak foundational services 3.7 düzeyine yükselttiyseniz, IBM Cloud Pak for Integration lisansı kullanan kuyruk yöneticilerine yükseltmeli ya da yeniden başlatılmalıdır. Bunun nasıl yapacağına ilişkin daha fazla bilgi için bkz. [“Red Hat OpenShift web konsolu kullanılarak bir IBM MQ kuyruk yöneticisinin yükseltilmesi” sayfa 76](#).

Upgrading IBM Cloud Pak foundational services using the Red Hat OpenShift web console

If you are upgrading from a version of IBM MQ Operator that is older than 1.5 to IBM MQ Operator 1.5 or later, you must first upgrade the version of IBM Cloud Pak foundational services.

Başlamadan önce

Not: Bu görevi, 1.5 ' tan daha eski bir IBM MQ Operator sürümünden IBM MQ Operator 1.5 ya da sonraki bir sürüme yükseltiyorsanız bu görevi tamamlamanız gerekir.

CP4I IBM Cloud Pak for Integration lisansı kullanan kuyruk yöneticileriniz varsa, bu büyütmeden sonra web konsoluna erişmek için kuyruk yöneticisi yeniden başlatılmalıdır; web konsolunda diğer hatalar oturma açma işlemini de görürsünüz. İşleç büyütme tamamlandıktan sonra, seçtiğiniz IBM MQ sürümünüz için en son `.spec.version` değerine yükselterek bu hataları düzeltebilirsiniz.

CP4I Var olan kuyruk yöneticileriniz varsa ve IBM Cloud Pak for Integration Operations Dashboard olanağını kullanıyorsanız, yükseltme işleminden önce “IBM Cloud Pak for Integration 2021.4’ünde Operations Dashboard bütünleşmesi ile IBM MQ 9.2.2 ya da 9.2.3 ürününün konuşlandırılması ya da büyütülmesi” sayfa 110 konusuna bakın.

Yordam

1. Red Hat OpenShift küme web konsolunuzda oturum açın.
2. Gezinme bölmesinden, **İşleçler** > **Kurulu İşleçler** öğelerini tıklayın.
Belirtilen projeden kurulu tüm Operatörler görüntülenir.
3. **IBM Cloud Pak foundational services İşleci**' yi seçin. Note that prior to version 3.7, this was called the **IBM Common Services İşleci**
4. **Abonelik** sekmesine gidin.
5. **Kanal**' ı tıklayın.
Abonelik Güncelleme Kanalını Değiştir penceresi görüntülenir.
6. **v3** kanalını seçin ve **Save**(Kaydet) düğmesini tıklayın.
IBM Cloud Pak foundational services işletmeni, yeni kanalda kullanılabilen en son sürüme yükselir.
Bkz. “IBM MQ Operator için sürüm desteği” sayfa 7.

Sonraki adım

Artık IBM MQ Operator ürününü büyütün' a hazırsınız.

OpenShift **CP4I** **Upgrading the IBM MQ Operator using the Red Hat OpenShift CLI**

IBM MQ Operator , komut satırından yükseltilebilir.

Başlamadan önce

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

IBM MQ Operator ürününü hava boşluğu ortamında büyütmeden önce en son IBM Cloud Pak for Integration görüntülerini ikizlemeniz gerekir. Bkz. IBM MQ Operator ya da kuyruk yöneticisini hava boşluğu ortamında büyütme hazırlık.

Yordam

1. Yükseltilecek işletmen kanalını belirlemek için “IBM MQ Operator için sürüm desteği” sayfa 7 ' i inceleyin.
2. İsteğe bağlı: If you are upgrading from a version of IBM MQ Operator that is older than 1.5 to IBM MQ Operator 1.5 or later, you must first upgrade the version of IBM Cloud Pak foundational services.
Daha fazla bilgi için bkz “Upgrading IBM Cloud Pak foundational services using the Red Hat OpenShift CLI” sayfa 76.
3. IBM MQ Operator ürününü yükseltin. Yeni ana/küçük IBM MQ Operator sürümleri, yeni Abonelik Kanalları aracılığıyla teslim edilir. İşlecinizi yeni bir ana/ikincil sürüme yükseltmek için, IBM MQ Operator Aboneliğinizde seçilen kanalı güncellenmeniz gerekir.
 - a) Gerekli IBM MQ Operator Upgrade Channel 'ın kullanılabilir olduğundan emin olun.

```
oc get packagemanifest ibm-mq -o=jsonpath='{.status.channels[*].name}'
```

- b) Subscription 'ı istenen güncelleme kanalına (burada *vX*) hareket ettirecek şekilde yama yap.Y , önceki adımda belirtilen güncelleme kanalıdır.

```
oc patch subscription ibm-mq --patch '{"spec":{"channel":"vX.Y"}}' --type=merge
```

Sonraki adım


IBM Cloud Pak foundational services 3.7 düzeyine yükselttiyseniz, IBM Cloud Pak for Integration lisansı kullanan kuyruk yöneticilerine yükseltilmeli ya da yeniden başlatılmalıdır. Bunun nasıl yapacağına ilişkin daha fazla bilgi için bkz. [“Red Hat OpenShift CLI kullanılarak bir IBM MQ kuyruk yöneticisinin yükseltilmesi” sayfa 77.](#)


Upgrading IBM Cloud Pak foundational services using the Red Hat OpenShift CLI

If you are upgrading from a version of IBM MQ Operator that is older than 1.5 to IBM MQ Operator 1.5 or later, you must first upgrade the version of IBM Cloud Pak foundational services.

Başlamadan önce

Not: Bu görevi, 1.5 ' tan daha eski bir IBM MQ Operator sürümünden IBM MQ Operator 1.5 ya da sonraki bir sürüme yükseltiyorsanız bu görevi tamamlamanız gerekir.

 IBM Cloud Pak for Integration lisansı kullanan kuyruk yöneticileriniz varsa, bu büyütmeden sonra web konsoluna erişmek için kuyruk yöneticisi yeniden başlatılmalıdır; web konsolunda [diğer hatalar](#) oturum açma işlemini de görürsünüz. İşleç büyütme tamamlandıktan sonra, seçtiğiniz IBM MQ sürümünüz için en son .spec.version değerine yükselterek bu hataları düzeltebilirsiniz.

 Var olan kuyruk yöneticileriniz varsa ve IBM Cloud Pak for Integration Operations Dashboard olanağını kullanıyorsanız, yükseltme işleminden önce [“IBM Cloud Pak for Integration 2021.4 içinde Operations Dashboard bütünleştirilmesi ile IBM MQ 9.2.2 ya da 9.2.3 ürününün konuşlandırılması ya da büyütülmesi” sayfa 110](#) konusuna bakın.

Yordam

1. Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.
2. v3 IBM Cloud Pak foundational services Upgrade Channel 'ın kullanılabilir olduğundan emin olun.

```
oc get packagemanifest -n ibm-common-services ibm-common-service-operator -o=jsonpath='{.status.channels[*].name}'
```

3. Patch the Subscription to move to the desired update channel: v3

```
oc patch subscription ibm-common-service-operator --patch '{"spec":{"channel":"v3"}}' --type=merge
```

Sonraki adım

Artık [IBM MQ Operator ürününü büyütün](#) 'a hazırsınız.

Red Hat OpenShift web konsolu kullanılarak bir IBM MQ kuyruk yöneticisinin yükseltilmesi

IBM MQ Operator kullanılarak konuşlandırılan bir IBM MQ kuyruk yöneticisi, Operator Hub kullanılarak Red Hat OpenShift içinde büyütülebilir.

Başlamadan önce

- Red Hat OpenShift küme web konsolunuzda oturum açın.

- IBM MQ Operator ' inizin istenen Güncelleme Kanalını kullandığını doğrulayın. Bkz. [“IBM MQ Operator ve kuyruk yöneticilerinin yükseltilmesi” sayfa 70.](#)

Bir hava boşluğu ortamında kuyruk yöneticisini yükseltebilmeniz için en son IBM Cloud Pak for Integration görüntülerini ikizlemeniz gerekir. Bkz. [IBM MQ Operator ya da kuyruk yöneticisini hava boşluğu ortamında büyümeye hazırlık.](#)

Yordam

1. Gezinme bölmesinden, **İşleçler > Kurulu İşleçler** öğelerini tıklatın.
Belirtilen projeden kurulu tüm Operatörler görüntülenir.
2. **IBM MQ Operator** ' ı seçin.
IBM MQ Operator penceresi görüntülenir.
3. **Kuyruk Yöneticisi** sekmesine gidin.
Kuyruk Yöneticisi Ayrıntıları penceresi görüntülenir.
4. Yükseltmek istediğiniz kuyruk yöneticisini seçin.
5. YAML sekmesine gidin.
6. İstenen IBM MQ kuyruk yöneticisi sürümü büyütmesini eşleştirmek için gereken yerlerde aşağıdaki alanları güncelleyin.
 - spec.version
 - spec.license.licenceKanalların IBM MQ Operator sürümleri ve IBM MQ kuyruk yöneticisi sürümlerine eşlenmesine ilişkin [“IBM MQ Operator için sürüm desteği” sayfa 7](#) konusuna bakın.
7. Güncellenmiş kuyruk yöneticisi YAML ' yi kaydedin.

Red Hat OpenShift CLI kullanılarak bir IBM MQ kuyruk yöneticisinin yükseltilmesi

IBM MQ Operator kullanılarak konuşlandırılan bir IBM MQ kuyruk yöneticisi, komut satırı kullanılarak Red Hat OpenShift içinde yükseltilebilir.

Başlamadan önce

Bu adımları tamamlamak için bir küme yöneticisi olmanız gerekir.

- Log in to the Red Hat OpenShift command line interface (CLI) using `oc login`.
- IBM MQ Operator ' inizin istenen Güncelleme Kanalını kullandığını doğrulayın. Bkz. [“IBM MQ Operator ve kuyruk yöneticilerinin yükseltilmesi” sayfa 70.](#)

Bir hava boşluğu ortamında kuyruk yöneticisini yükseltebilmeniz için en son IBM Cloud Pak for Integration görüntülerini ikizlemeniz gerekir. Bkz. [IBM MQ Operator ya da kuyruk yöneticisini hava boşluğu ortamında büyümeye hazırlık.](#)

Yordam

desired kaynağını, istenen IBM MQ kuyruk yöneticisi sürümü yükseltmesi ile eşleşecek şekilde, gereken yerlerde güncellemek için **QueueManager** kaynağını düzenleyin.

- spec.version
- spec.license.licence

Kanalların IBM MQ Operator sürümleri ve IBM MQ kuyruk yöneticisi sürümlerine eşlenmesine ilişkin [“IBM MQ Operator için sürüm desteği” sayfa 7](#) konusuna bakın.

Aşağıdaki komutu kullanın:

```
oc edit queuemanager my_qmgr
```

Burada *my_qmgr* , yükseltmek istediğiniz QueueManager kaynağının adıdır.

CP4I Platform Navigator'ı kullanarak Red Hat OpenShift ' ta bir IBM MQ kuyruk yöneticisinin yükseltilmesi

An IBM MQ queue manager, deployed using the IBM MQ Operator, can be upgraded in Red Hat OpenShift using the IBM Cloud Pak for Integration Platform Navigator.

Başlamadan önce

- Yükseltmek istediğiniz kuyruk yöneticisini içeren ad alanındaki IBM Cloud Pak for Integration Platform Navigator ' da oturum açın.
- IBM MQ Operator ' inizin istenen Güncelleme Kanalını kullandığını doğrulayın. Bkz. [“IBM MQ Operator ve kuyruk yöneticilerinin yükseltilmesi” sayfa 70.](#)

Bir hava boşluğu ortamında kuyruk yöneticisini yükseltebilmeniz için en son IBM Cloud Pak for Integration görüntülerini ikizlemeniz gerekir. Bkz. [IBM MQ Operator ya da kuyruk yöneticisini hava boşluğu ortamında büyütme hazırlık.](#)

Yordam

1. IBM Cloud Pak for Integration Platform Navigator giriş sayfasından, **Runtimes** (Runtimes) sekmesini tıklatın.
2. Kullanılabilir büyütme sahipli kuyruk yöneticilerine, **Sürüm**yanında mavi bir **i** bulunur. **Yeni sürüm kullanılabilir**'i göstermek için **i** ' yi tıklatın.
3. Yükseltmek istediğiniz kuyruk yöneticisinin sağ uzaktaki üç noktayı tıklatın ve **Sürümü değiştir**ögesini tıklatın.
4. **Select a new channel or version**(Yeni kanal ya da sürüm seçin) altında, gereken yükseltme sürümünü seçin.
5. **Sürümü değiştir**ögesini tıklatın.

Sonuçlar

Kuyruk yöneticisi yükseltiliyor.

OpenShift CP4I Deploying and configuring queue managers using the IBM MQ Operator

IBM MQ 9.1.5 ve daha sonraki düzeyler, IBM MQ Operator kullanılarak Red Hat OpenShift ' ye konuşlandırılır.

Bu görev hakkında

Yordam

- [“Preparing your Red Hat OpenShift project for IBM MQ” sayfa 78.](#)
- [“Kuyruk yöneticisinin bir Red Hat OpenShift Container Platform kümesine konuşlandırılması” sayfa 80.](#)

OpenShift CP4I Preparing your Red Hat OpenShift project for IBM MQ

Prepare your Red Hat OpenShift Container Platform cluster so that it's ready to deploy a queue manager.

Yordam

- [“Preparing your Red Hat OpenShift project for IBM MQ using the Red Hat OpenShift web console” sayfa 79.](#)
- [“Preparing your Red Hat OpenShift project for IBM MQ using the Red Hat OpenShift CLI” sayfa 80.](#)

İlgili görevler

[“Kuyruk yöneticisinin bir Red Hat OpenShift Container Platform kümesine konuşlandırılması” sayfa 80](#)
Bir kuyruk yöneticisini bir Red Hat OpenShift Container Platform kümesine konuşlandırmak için QueueManager özel kaynağını kullanın.

Preparing your Red Hat OpenShift project for IBM MQ using the Red Hat OpenShift web console

Prepare your Red Hat OpenShift Container Platform cluster, so that it's ready to deploy a queue manager using the IBM MQ Operator. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır.

Başlamadan önce

Not: IBM MQ ' u başka IBM Cloud Pak for Integration bileşenleriyle birlikte bir projede kullanmayı planlıyorsanız, bu yönergeleri izlemenize gerek yoktur.

Red Hat OpenShift küme web konsolunuzda oturum açın.

Bu görev hakkında

IBM MQ Operator görüntüleri, lisans hakkı denetimi gerçekleştiren bir kapsayıcı kayıt dosyasından çekiliyor. Bu denetim, bir `docker-registry` çekme sırasında saklanan bir yetki anahtarını gerektirir. Henüz bir yetki anahtarınız yoksa, bir yetki anahtarı almak ve bir çekme güvenlik dizgisi yaratmak için bu yönergeleri izleyin.

Yordam

1. Kimliğinize atanan yetki anahtarını alın.
 - a) Yetkili yazılımla ilişkili IBM kimliği ve parolasıyla [MyIBM Kapsayıcı Yazılım Kitaplığı](#) olanağında oturum açın.
 - b) In the **Yetki anahtarları** section, select **Kopyalama anahtarı** to copy the entitlement key to the clipboard.
2. Kuyruk yöneticinizi konuşlandırmak istediğiniz projede, yetki anahtarınızı içeren bir güvenlik dizgisi yaratın.
 - a) Gezinme bölmesinden **İş Yükleri > Gizli** ' i tıklatın.
Sırlar sayfası görüntülenir.
 - b) **Proje** açılan listesinde, IBM MQürününü kurmak istediğiniz projeyi seçin.
 - c) **Oluştur** düğmesini tıklatın ve **Resim Çekme Gizliliği** seçeneğini belirleyin.
 - d) **Ad** alanına `ibm-entitlement-key` girin.
 - e) **Registry Server Address** (Kayıt Dosyası Sunucu Adresi) alanına `cp.icr.io` girin.
 - f) **Kullanıcı adı** alanına `cp` girin.
 - g) **Parola** alanına, önceki adımda kopyaladığınız yetki anahtarını girin.
 - h) **E-posta** alanına, yetkili yazılımla ilişkili IBM kimliğini girin.

Sonraki adım

[“Deploying a queue manager using the Red Hat OpenShift web console” sayfa 82](#)

Red Hat OpenShift CLI

Prepare your Red Hat OpenShift Container Platform cluster, so that it's ready to deploy a queue manager using the IBM MQ Operator. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır.

Başlamadan önce

Not: IBM MQ ' u başka IBM Cloud Pak for Integration bileşenleriyle birlikte bir projede kullanmayı planlıyorsanız, bu yönergeleri izlemenize gerek yoktur.

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

Bu görev hakkında

IBM MQ Operator görüntüleri, lisans hakkı denetimi gerçekleştiren bir kapsayıcı kayıt dosyasından çekiliyor. Bu denetim, bir `docker-registry` çekme sırasında saklanan bir yetki anahtarını gerektirir. Henüz bir yetki anahtarınız yoksa, bir yetki anahtarı almak ve bir çekme güvenlik dizgisi yaratmak için bu yönergeleri izleyin.

Yordam

1. Kimliğinize atanan yetki anahtarını alın.

- Yetkili yazılımla ilişkili IBM kimliği ve parolasıyla [MyIBM Kapsayıcı Yazılım Kitaplığı](#) olanağında oturum açın.
- In the **Yetki anahtarları** section, select **Kopyalama anahtarı** to copy the entitlement key to the clipboard.

2. Kuyruk yöneticinizi konuşlandırmak istediğiniz projede, yetki anahtarınızı içeren bir güvenlik dizgisi yaratın.

Şu komutu çalıştırın; burada `<entitlement-key>` , 1. adımda alınan anahtardır ve `<user-email>` , yetkili yazılımla ilişkilendirilen IBM kimliğidir.

```
oc create secret docker-registry ibm-entitlement-key \
--docker-server=cp.icr.io \
--docker-username=cp \
--docker-password=<entitlement-key> \
--docker-email=<user-email>
```

Sonraki adım

[“Deploying a queue manager using the Red Hat OpenShift CLI” sayfa 83](#)

Platform kümesine konuşlandırılması

Bir kuyruk yöneticisini bir Red Hat OpenShift Container Platform kümesine konuşlandırmak için QueueManager özel kaynağını kullanın.

Yordam

- [CP4I](#)
[“Deploying a queue manager using the IBM Cloud Pak for Integration Platform Navigator” sayfa 81.](#)
- [OpenShift](#)
[“Deploying a queue manager using the Red Hat OpenShift web console” sayfa 82.](#)
- [OpenShift](#)
[“Deploying a queue manager using the Red Hat OpenShift CLI” sayfa 83.](#)

İlgili görevler

“Kuyruk yöneticisi yapılandırılmasına ilişkin örnekler” sayfa 84

Kuyruk yöneticisi, QueueManager özel kaynağının içeriği ayarlanarak yapılandırılabilir.

CP4I *Deploying a queue manager using the IBM Cloud Pak for Integration*

Platform Navigator

Bir kuyruk yöneticisini IBM Cloud Pak for Integration Platform Navigator kullanarak bir Red Hat OpenShift Container Platform kümesine konuşlandırmak için QueueManager özel kaynağını kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır

Başlamadan önce

Bir tarayıcıda IBM Cloud Pak for Integration Platform Navigator uygulamasını başlatın.

Bu, bir kuyruk yöneticisini bu Red Hat OpenShift projesinde ilk kez konuşlandırdıysa, [“Preparing your Red Hat OpenShift project for IBM MQ” sayfa 78](#) için adımları izleyin.

Yordam

1. Kuyruk yöneticisini konuşlandırın.

Aşağıdaki örnek, ephemeral (kalıcı olmayan) depolama kullanan ve MQ güvenliğini devre dışı döndüren bir "quick start" kuyruk yöneticisini konuşlandırır. Kuyruk yöneticisinin yeniden başlatılması sırasında iletiler kalıcı olarak saklanmaz. Birçok kuyruk yöneticisi ayarlarını değiştirmek için yapılandırmayı ayarlayabilirsiniz.

a) IBM Cloud Pak for Integration Platform Navigator' ta **Yönetim** ve **Tümleştirme Yürütme Ortamları** öğelerini tıklatın. In older versions of the IBM Cloud Pak for Integration Platform Navigator, click **Yürütme ortamı ve eşgörünümler**.

b) **Eşgörünüm yarat** öğesini tıklatın.

c) **Messaging** seçeneğini belirleyin ve **Next** (İleri) düğmesini tıklatın. IBM Cloud Pak for Integration Platform Navigator'ın eski sürümlerinde **Kuyruk Yöneticisi** ' u tıklatın ve **Sonraki** ' yi tıklatın. Bir QueueManager yönetim ortamı yaratmak için form görüntülenir.

Not: QueueManager yapılandırması YAML 'yi görüntülemek ya da değiştirmek için **Kod** ' u da tıklatabilirsiniz.

d) **Ayrıntılar** bölümünde, **Ad** alanını işaretleyin ya da güncelleştirin ve kuyruk yöneticisi örneğinin yaratılacağı **Ad alanı** ' yı belirtin.

e) IBM Cloud Pak for Integration lisans sözleşmesini kabul ederseniz, **Lisans kabulü** seçeneğini **Açık** olarak değiştirin.

Bir kuyruk yöneticisini konuşlandırmak için lisansı kabul etmeniz gerekir.

f) **Kuyruk Yöneticisi** bölümünde, temeldeki kuyruk yöneticisinin **Name** (Ad) değerini denetleyin ya da güncelleyin. In older versions of the IBM Cloud Pak for Integration Platform Navigator, use the **Kuyruk Yöneticisi Yapılandırma** section.

By default, the name of the queue manager used by IBM MQ client applications will be the same as the name of the QueueManager, but with any invalid characters (such as hyphens) removed.

g) **Yarat** düğmesini tıklatın.

Yürürlükteki projeden (ad alanı) kuyruk yöneticilerinin listesi görüntülenir. Yeni QueueManager , Pending durumunda olmalıdır.

2. Kuyruk yöneticisinin çalışır durumda olup olmadığını denetleyin

The creation is complete when the QueueManager status is Running.

İlgili görevler

“Red Hat OpenShift kümesi dışından bir kuyruk yöneticisine bağlanmak için bir Rotanın yapılandırılması” sayfa 107

Bir uygulamayı Red Hat OpenShift kümesi dışından IBM MQ kuyruk yöneticisine bağlamak için bir Red Hat OpenShift Rotası gerekir. SNI yalnızca TLS 1.2 ya da daha yüksek bir iletişim kuralı kullanıldığında

TLS iletişim kuralında kullanılabilirdiğinden, IBM MQ kuyruk yöneticisi ve istemci uygulamanızda TLS 'yi etkinleştirmeniz gerekir. Red Hat OpenShift Container Platform Router , istekleri IBM MQ kuyruk yöneticisine yönlendirmek için SNI kullanır.

“Red Hat OpenShift kümesinde konuşlandırılmış IBM MQ Console ile bağlantı kurulması” sayfa 113 Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

Deploying a queue manager using the Red Hat OpenShift web console

Bir kuyruk yöneticisini Red Hat OpenShift Web konsolunu kullanarak bir Red Hat OpenShift Container Platform kümesine konuşlandırmak için QueueManager özel kaynağını kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır

Başlamadan önce

Red Hat OpenShift küme web konsolunuzda oturum açın. Kullanmak için var olan bir Proje (ad alanı) seçmeniz ya da yeni bir proje yaratmanız gerekir.

Bu, bir kuyruk yöneticisini bu Red Hat OpenShift projesinde ilk kez konuşlandırdıysa, “Preparing your Red Hat OpenShift project for IBM MQ” sayfa 78 için adımları izleyin.

Yordam

1. Kuyruk yöneticisini konuşlandırın.

Aşağıdaki örnek, ephemeral (kalıcı olmayan) depolama kullanan ve MQ güvenliğini devre dışı döndüren bir "quick start" kuyruk yöneticisini konuşlandırır. Kuyruk yöneticisinin yeniden başlatılması sırasında iletiler kalıcı olarak saklanmaz. Birçok kuyruk yöneticisi ayarlarını değiştirmek için yapılandırmayı ayarlayabilirsiniz.

- Red Hat OpenShift web konsolunda, gezinme bölmesinden **İşleçler > Kurulu İşleçler** seçeneklerini tıklatın.
- IBM MQ** seçeneğini tıklatın.
- Kuyruk Yöneticisi** etiketini tıklatın.
- Create QueueManager** (QueueManager Yarat) düğmesini tıklatın.

A YAML editor is displayed, containing example YAML for a QueueManager resource.

Not: Ayrıca, QueueManager yapılandırmasını görüntülemek ya da değiştirmek için **Formu Düzenle** 'yi de tıklatabilirsiniz.

- Lisans sözleşmesini kabul ederseniz, **Lisans kabulü** seçeneğini **Açık** olarak değiştirin.
IBM MQ , farklı lisanslar altında kullanılabilir. Geçerli lisanslarla ilgili daha fazla bilgi için bkz. [“mq.ibm.com/v1beta1 için lisans başvurusu” sayfa 124](#). Bir kuyruk yöneticisini konuşlandırmak için lisansı kabul etmeniz gerekir.
- Yarat** düğmesini tıklatın.
Yürürlükteki projeden (ad alanı) kuyruk yöneticilerinin listesi görüntülenir. Yeni QueueManager , bir Pending durumunda olmalıdır.

2. Kuyruk yöneticisinin çalışır durumda olup olmadığını denetleyin

The creation is complete when the QueueManager status is Running.

İlgili görevler

[“Red Hat OpenShift kümesi dışından bir kuyruk yöneticisine bağlanmak için bir Rotanın yapılandırılması” sayfa 107](#)

Bir uygulamayı Red Hat OpenShift kümesi dışından IBM MQ kuyruk yöneticisine bağlamak için bir Red Hat OpenShift Rotası gerekir. SNI yalnızca TLS 1.2 ya da daha yüksek bir iletişim kuralı kullanıldığında TLS iletişim kuralında kullanılabilirdiğinden, IBM MQ kuyruk yöneticisi ve istemci uygulamanızda TLS 'yi etkinleştirmeniz gerekir. Red Hat OpenShift Container Platform Router , istekleri IBM MQ kuyruk yöneticisine yönlendirmek için SNI kullanır.

“Red Hat OpenShift kümesinde konuşlandırılmış IBM MQ Console ile bağlantı kurulması” sayfa 113
Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ
Console ile bağlantı kurma yöntemi.

OpenShift CP4I **Deploying a queue manager using the Red Hat OpenShift CLI**

Bir kuyruk yöneticisini komut satırı arabirimi (CLI) kullanarak bir Red Hat OpenShift Container Platform kümesine konuşlandırmak için QueueManager özel kaynağını kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır

Başlamadan önce

Red Hat OpenShift Container Platform komut satırı arabirimi' yi kurmanız gerekir.

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

Bu, bir kuyruk yöneticisini bu Red Hat OpenShift projesinde ilk kez konuşlandırdıysa, “Preparing your Red Hat OpenShift project for IBM MQ” sayfa 78 için adımları izleyin.

Yordam

1. Kuyruk yöneticisini konuşlandırın.

Aşağıdaki örnek, ephemeral (kalıcı olmayan) depolama kullanan ve MQ güvenliğini devre dışı döndüren bir "quick start" kuyruk yöneticisini konuşlandırır. Kuyruk yöneticisinin yeniden başlatılması sırasında iletiler kalıcı olarak saklanmaz. Bir çok kuyruk yöneticisi ayarlarını değiştirmek için YAML içeriğini ayarlayabilirsiniz.

a) QueueManager YAML dosyası yarat

Örneğin, IBM Cloud Pak for Integration içinde temel bir kuyruk yöneticisi kurmak için, aşağıdaki içerikle "mq-quickstart.yaml" dosyasını yaratın:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: quickstart-cp4i
spec:
  version: 9.2.5.0-r3
  license:
    accept: false
    license: L-RJON-C7QG3S
    use: NonProduction
  web:
    enabled: true
  queueManager:
    name: "QUICKSTART"
    storage:
      queueManager:
        type: ephemeral
  template:
    pod:
      containers:
        - name: qmgr
          env:
            - name: MQSNOAUT
              value: "yes"
```

Önemli: IBM Cloud Pak for Integration lisans sözleşmesini kabul ederseniz, `accept: false` seçeneğini `accept: true` olarak değiştirin. Lisansa ilişkin ayrıntılar için “mq.ibm.com/v1beta1 için lisans başvurusu” sayfa 124 ' e bakın.

Bu örnek ayrıca, kuyruk yöneticisiyle birlikte konuşlandırılan bir web sunucusunu da içerir; web konsolu, IBM Cloud Pak Identity and Access Manager ile Tekli Oturum Açma özelliği etkinleştirilmiş olarak etkinleştirilmiştir.

Temel kuyruk yöneticisini IBM Cloud Pak for Integration' den bağımsız olarak kurmak için, aşağıdaki içerikle "mq-quickstart.yaml" dosyasını yaratın:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: quickstart
spec:
  version: 9.2.5.0-r3
  license:
    accept: false
    license: L-APIG-BZDDDY
  web:
    enabled: true
  queueManager:
    name: "QUICKSTART"
    storage:
      queueManager:
        type: ephemeral
  template:
    pod:
      containers:
        - name: qmgr
          env:
            - name: MQSNOAUT
              value: "yes"
```

Önemli: MQ lisans sözleşmesini kabul ederseniz, `accept: false` ögesini `accept: true` olarak değiştirin. Lisansa ilişkin ayrıntılar için ["mq.ibm.com/v1beta1 için lisans başvurusu"](#) sayfa 124 ' e bakın.

b) QueueManager nesnesini oluşturma

```
oc apply -f mq-quickstart.yaml
```

2. Kuyruk yöneticisinin çalışır durumda olup olmadığını denetleyin
Konuşlandırmanın geçerliliğini denetleyerek doğrulayabilirsiniz.

```
oc describe queuemanager <QueueManagerResourceName>
```

Ve sonra durumu kontrol ediyorum.

Örneğin,

```
oc describe queuemanager quickstart
```

, and check that the `status.Phase` field indicates `Running`

İlgili görevler

["Red Hat OpenShift kümesi dışından bir kuyruk yöneticisine bağlanmak için bir Rotanın yapılandırılması"](#) sayfa 107

Bir uygulamayı Red Hat OpenShift kümesi dışından IBM MQ kuyruk yöneticisine bağlamak için bir Red Hat OpenShift Rotası gerekir. SNI yalnızca TLS 1.2 ya da daha yüksek bir iletişim kuralı kullanıldığında TLS iletişim kuralında kullanılabilir olduğundan, IBM MQ kuyruk yöneticisi ve istemci uygulamanızda TLS ' yi etkinleştirmeniz gerekir. Red Hat OpenShift Container Platform Router , istekleri IBM MQ kuyruk yöneticisine yönlendirmek için SNI kullanır.

["Red Hat OpenShift kümesinde konuşlandırılmış IBM MQ Console ile bağlantı kurulması"](#) sayfa 113
Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

Kuyruk yöneticisi yapılandırılmasına ilişkin örnekler

Kuyruk yöneticisi, QueueManager özel kaynağının içeriği ayarlanarak yapılandırılabilir.

Bu görev hakkında

Use the following examples to help you configure a queue manager using the QueueManager YAML file.

Yordam

- “Örnek: MQSC ve INI dosyaları sağlar” sayfa 85
- “Örnek: TLS ' nin yapılandırılması” sayfa 86

OpenShift CP4I Örnek: MQSC ve INI dosyaları sağlar

Bu örnek, iki MQSC dosyası ve bir INI dosyası içeren bir Kubernetes ConfigMap oluşturur. Daha sonra, bu MQSC ve INI dosyalarını işleyen bir kuyruk yöneticisi konuşlandırılır.

Bu görev hakkında

Bir kuyruk yöneticisi konuşlandırıldığında, MQSC ve INI kütükleri sağlanabilir. MQSC ve INI verileri bir ya da daha çok Kubernetes ConfigMaps ve Secrets(Sırlar) içinde tanımlanmalıdır. Bunlar, kuyruk yöneticisini konuşlandıracağınız ad alanı (proje) içinde yaratılmalıdır.

Not: MQSC ya da INI dosyası duyarlı veriler içerdiğinde bir Kubernetes Secret kullanılmalıdır.

Bu şekilde MQSC ve INI sağlamak için IBM MQ Operator 1.1 ya da üstü gereklidir.

Örnek

Aşağıdaki örnek, iki MQSC dosyası ve bir INI dosyası içeren bir Kubernetes ConfigMap oluşturur. Daha sonra, bu MQSC ve INI dosyalarını işleyen bir kuyruk yöneticisi konuşlandırılır.

Örnek ConfigMap -kümeünüzde aşağıdaki YAML ' yi uygulayın:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: mqsc-ini-example
data:
  example1.mqsc: |
    DEFINE QLOCAL('DEV.QUEUE.1') REPLACE
    DEFINE QLOCAL('DEV.QUEUE.2') REPLACE
  example2.mqsc: |
    DEFINE QLOCAL('DEV.DEAD.LETTER.QUEUE') REPLACE
  example.ini: |
    Channels:
      MQIBindType=FASTPATH
```

Örnek QueueManager -komut satırını kullanarak ya da IBM Cloud Pak for Integration Platform Navigatorkomutunu kullanarak kuyruk yöneticinizi aşağıdaki yapılandırlarla konuşlandırın:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: mqsc-ini-cp4i
spec:
  version: 9.2.5.0-r3
  license:
    accept: false
    license: L-RJON-C7QG3S
    use: NonProduction
  web:
    enabled: true
  queueManager:
    name: "MQSCINI"
    mqsc:
      - configMap:
          name: mqsc-ini-example
          items:
            - example1.mqsc
            - example2.mqsc
    ini:
      - configMap:
          name: mqsc-ini-example
          items:
            - example.ini
  storage:
    queueManager:
      type: ephemeral
```

Önemli: IBM Cloud Pak for Integration lisans sözleşmesini kabul ederseniz, `accept: false` seçeneğini `accept: true` olarak değiştirin. Lisansa ilişkin ayrıntılar için mq.ibm.com/v1beta1 için lisans başvurusu ' e bakın.

Ek bilgi:

- Kuyruk yöneticisi, tek bir Kubernetes ConfigMap ya da Secret (bu örnekte gösterildiği gibi) ya da birden çok Kubernetes ConfigMaps ve Secrets (Sırlar) kullanacak şekilde yapılandırılabilir.
- Tüm MQSC ve INI verilerini bir Kubernetes ConfigMap ya da Secret (örnekte gösterildiği gibi) içinden kullanmayı ya da her bir kuyruk yöneticisini kullanılabılır dosyaların yalnızca bir alt kümesini kullanacak şekilde yapılandırmanızı seçebilirsiniz.
- MQSC ve INI dosyaları, anahtarlarına dayalı olarak alfabetik sırayla işlenir. So `example1.mqsc` will always be processed before `example2.mqsc`, regardless of the order in which they appear in the queue manager configuration.
- Birden çok Kubernetes ConfigMaps ya da Secrets arasında birden çok MQSC ya da INI dosyası varsa, bu kütük kümesi, kütüklerin kuyruk yöneticisi yapılandırıldığı tanımlarına dayalı olarak işlenir.

OpenShift CP4I Linux **Örnek: TLS ' nin yapılandırılması**

Bu örnek, IBM MQ Operator komutunu kullanarak bir kuyruk yöneticisini Red Hat OpenShift Container Platform içine konuşturur. Örnek bir istemci ile kuyruk yöneticisi arasında tek yönlü TLS iletişimi yapılandırılır. Bu örnek, iletileri koyarak ve alma yoluyla başarılı bir yapılandırma gösterir.

Başlamadan önce

Bu örneği tamamlamak için, önce aşağıdaki önkoşulları tamamlamanız gerekir:

- IBM MQ client'yi kurun ve `PATH`' inize `sample/bin` ve `bin` ekleyin. You need the **runmqadm**, **amqspu**tc and **amqsgetc** applications, which can be installed as part of the IBM MQ client as follows:
 - **Linux** **Windows** Windows ve Linux için: İşletim sisteminize ilişkin IBM MQ yeniden dağıtılabilir istemcisini <https://ibm.biz/mq92redistclients> olanağundan kurun.
 - **mac OS** Mac için: IBM MQ MacOS Toolkit dosyasını yükleyin ve ayarlayın: <https://developer.ibm.com/tutorials/mq-macos-dev/>
- İşletim sisteminize ilişkin OpenSSL aracını kurun.
- Bu örnek için bir Red Hat OpenShift Container Platform (OCP) projesi/ad alanı yaratın.
- Komut satırında, İşletmen Denetim Panosu kümesinde oturum açın ve yukarıdaki ad alanına geçin.
- IBM MQ Operator ' in yukarıdaki ad alanında kurulu ve kullanılabılır olduğundan emin olun.

Bu görev hakkında

Bu örnek, Red Hat OpenShift Container Platform' a konuşturılacak kuyruk yöneticisini tanımlayan bir özel kaynak YAML sağlar. Ayrıca, kuyruk yöneticisini TLS etkinleştirilmiş olarak konuşturmak için gereken ek adımları da ayrıntılarıyla içerir. Tamamlanmasının ardından, iletilerin yerleştirilmesi ve alınması kuyruk yöneticisinin TLS ile yapılandırılıp yapılandırıldığı doğrulanır.

IBM MQ sunucusu için bir TLS özel anahtarı ve sertifikaları oluşturma

Aşağıdaki kod örnekleri, kuyruk yöneticisi için kendinden onaylı sertifika yaratılmasını ve istemcinin istemci için güvenli depo olarak işlev görmeleri için anahtar veritabanına nasıl ekleneceğini göstermektedir. Zaten özel bir anahtar ve sertifikanız varsa, bunları kullanabilirsiniz.

Kendinden imzalı sertifikaların yalnızca geliştirme amaçlı olarak kullanılması gerektiğini unutmayın.

Yürürlükteki dizinde kendinden onaylı bir özel anahtar ve genel sertifika yaratır

Aşağıdaki komutu çalıştırın:

```
openssl req -newkey rsa:2048 -nodes -keyout tls.key -subj "/CN=localhost" -x509 -days 3650
-out tls.crt
```

Sunucu genel anahtarını bir istemci anahtar veritabanına ekle

Anahtar veritabanı, istemci uygulaması için güvenli depo olarak kullanılır.

İstemci anahtar veri tabanını yaratın:

```
runmqacm -keydb -create -db clientkey.kdb -pw password -type cms -stash
```

Önceden oluşturulan genel anahtarı istemci anahtarı veritabanına ekle:

```
runmqacm -cert -add -db clientkey.kdb -label mqservercert -file tls.crt -format ascii
-stashed
```

Kuyruk Yöneticisi konuşlandırması için TLS Sertifikalarını yapılandır

Böylece, kuyruk yöneticinizin anahtarı ve sertifikayı gönderip uygulayabilir, yukarıda yaratılan dosyalara gönderme yapan bir Kubernetes TLS güvenlik dizgisi yaratabilirler. Bunu yaparken, bu göreve başlamadan önce oluşturduğunuz ad alanında bulunduğunuzdan emin olun.

```
oc create secret tls example-tls-secret --key="tls.key" --cert="tls.crt"
```

MQSC komutlarını içeren bir yapılandırma eşlemi yaratır

Yeni bir kuyruk ve bir SVRCONN Kanalı yaratmak ve yalnızca *Kimse* adlı kullanıcıları engelleyerek kanala erişime izin veren bir kanal kimlik doğrulaması kaydı eklemek için, MQSC komutlarını içeren bir Kubernetes yapılandırma eşlemi yaratın.

Bu yaklaşımın yalnızca geliştirme amaçları için kullanılması gerektiğini unutmayın.

Daha önce oluşturduğunuz ad alanında olduğunuzu doğrulayın (bkz. [Başlamadan önce](#)), ardından OCP UI 'ye aşağıdaki YAML' yi girin ya da komut satırını kullanın.

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: example-tls-configmap
data:
  tls.mqsc: |
    DEFINE QLOCAL('EXAMPLE.QUEUE') REPLACE
    DEFINE CHANNEL(SECUREQMCHL) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCAUTH(OPTIONAL)
    SSLCIPH('ANY_TLS12_OR_HIGHER')
    SET CHLAUTH(SECUREQMCHL) TYPE(BLOCKUSER) USERLIST('nobody') ACTION(ADD)
```

Gerekli OCP rotasını oluşturun

Bu göreve başlamadan önce, oluşturduğunuz ad alanında olduğunuzu doğrulayın, daha sonra OCP UI 'ye aşağıdaki YAML' yi girin ya da komut satırını kullanın.

```
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: example-tls-route
spec:
  host: secureqmchl.ch1.mq.ibm.com
  to:
    kind: Service
    name: secureqm-ibm-mq
  port:
    targetPort: 1414
  tls:
    termination: passthrough
```

Note that the Red Hat OpenShift Container Platform Router uses SNI for routing requests to the IBM MQ queue manager. Yapılandırma eşleminde daha önce yaratılan MQSC ' de belirtilen kanal adını değiştirirseniz, anasistem alanını da burada değiştirmeniz ve CCDT dosyasında daha sonra yaratılmış olması gerekir. Daha fazla bilgi için bkz [“Red Hat OpenShift kümesi dışından bir kuyruk yöneticisine bağlanmak için bir Rotanın yapılandırılması” sayfa 107.](#)

Kuyruk yöneticisini konuşlandır

Önemli: Bu örnekte, kuyruk yöneticisinde yetkilendirmeyi devre dışı bırakmak için *MQSNOAUT* değişkenini kullanırız; bu değişken, bir istemciyi TLS 'yi kullanarak bağlamak için gereken adımlara odaklanabilmemizi sağlar. This is not recommended in a production deployment of IBM MQ, because it causes any applications connecting to have full administrative powers, with no mechanism to lower the permissions for individual applications.

Aşağıdaki özel kaynak YAML 'yi kullanarak yeni bir kuyruk yöneticisi yaratın. *MQSNOAUT* değişkeninin yanı sıra, daha önce yaratılan yapılandırma eşlemine ve gizli yaratıya gönderme yaptığı unutulmalıdır.

Bu göreve başlamadan önce oluşturduğunuz ad alanında bulunduğunuzdan emin olun, daha sonra komut satırını kullanarak ya da IBM Cloud Pak for Integration Platform Navigatorkomutunu kullanarak OCP UI 'ye aşağıdaki YAML' yi girin. Doğru lisansın belirtilip belirtildiğini denetleyin ve `false` ile `true` arasındaki değeri değiştirerek lisansı kabul edin.

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: secureqm
spec:
  license:
    accept: false
    license: L-RJON-C7QG3S
    use: Production
  queueManager:
    name: SECUREQM
  mqsc:
    - configMap:
        name: example-tls-configmap
        items:
          - tls.mqsc
    storage:
      queueManager:
        type: ephemeral
  template:
    pod:
      containers:
        - env:
            - name: MQSNOAUT
              value: 'yes'
          name: qmgr
  version: 9.2.5.0-r3
  web:
    enabled: true
  pki:
    keys:
      - name: example
        secret:
          secretName: example-tls-secret
          items:
            - tls.key
            - tls.crt
```

Kuyruk yöneticisinin çalışır durumda olduğunu doğrulayın

Kuyruk yöneticisi şimdi konuşlandırılıyor. Confirm it is in Running state before proceeding. Örneğin:

```
oc get qmgr secureqm
```

Kuyruk yöneticisiyle bağlantıyı test edin

Kuyruk yöneticisinin tek yönlü TLS iletişimi için yapılandırılıp yapılandırıldığı doğrulamak için, **amqspu`t`c** ve **amqsget`c`** örnek uygulamalarını kullanın:

Kuyruk yöneticisi anasistem adını bul

`secureqm-ibm-mq-qm`rotasına ilişkin kuyruk yöneticisini tam olarak nitelenmiş anasistem adını bulmak için aşağıdaki komutu kullanın:

```
oc get routes secureqm-ibm-mq-qm
```


Kuyruk yöneticisi ayrıntılarını belirtin

Create a file CCDT . JSON that specifies the queue manager details. Anasistem değerini, önceki adımdaki anasistem adıyla değiştirin.

```
{
  "channel":
  [
    {
      "name": "SECUREQMCHL",
      "clientConnection":
      {
        "connection":
        [
          {
            "host": "<hostname from previous step>",
            "port": 443
          }
        ],
        "queueManager": "SECUREQM"
      },
      "transmissionSecurity":
      {
        "cipherSpecification": "ECDHE_RSA_AES_128_CBC_SHA256"
      },
      "type": "clientConnection"
    }
  ]
}
```

Ortam değişkenlerini dışa aktar

İşletim sisteminiz için uygun şekilde aşağıdaki ortam değişkenlerini dışa aktarın. Bu değişkenler **amqsputc** ve **amqsgetc** tarafından okunacaktır.

Sisteminizdeki kütüklerin yolunu güncelleyin:

```
export MQCCDTURL='<full path to file>/CCDT.JSON'
export MQSSLKEYR='<full path to file>/clientkey'
```

İletileri kuyruğa koy

Aşağıdaki komutu çalıştırın:

```
amqsputc EXAMPLE.QUEUE SECUREQM
```

Kuyruk yöneticisiyle bağlantı başarılı olursa, aşağıdaki yanıt çıkışa gelir:

```
target queue is EXAMPLE.QUEUE
```

Kuyruğa birkaç ileti koyun, bir metin girerek her seferinde **Enter** tuşuna basın.

İşlemi tamamlamak için iki kez **Enter** tuşuna basın.

İletileri kuyruktan al

Aşağıdaki komutu çalıştırın:

```
amqsgetc EXAMPLE.QUEUE SECUREQM
```

Önceki adımda eklediğiniz iletiler tüketildi ve çıkışa yazılıyor.

Birkaç saniye sonra, komuta kapanır.

Tebrikler, TLS 'nin etkinleştirilmiş olduğu bir kuyruk yöneticisini başarıyla yerleştirdiniz ve bir istemciden kuyruk yöneticisine güvenli bir şekilde ileti alabileceğiniz ve güvenli bir şekilde ileti alabileceğiniz gösterilmiştir.

OpenShift

CP4I

Örnek: Lisans hizmeti ek açıklamalarının uyarlanması

IBM MQ Operator , konuşlandırılan kaynaklara IBM License Service ek açıklamalarını otomatik olarak ekler. Bunlar IBM License Servicetarafından izlenir ve gerekli yetkinin karşılığı olan raporlar oluşturulur.

Bu görev hakkında

IBM MQ Operator tarafından eklenen ek açıklamalar, standart durumlarda beklenenlerdir ve kuyruk yöneticisinin konuşlandırılırken seçilen lisans değerlerini temel alır.

Örnek

License , L-RJON-BZFQU2 (IBM Cloud Pak for Integration 2021.2.1) olarak ayarlandıysa ve **Use** , NonProduction olarak ayarlandıysa, aşağıdaki ek açıklamalar uygulanır:

- cloudpakId: c8b82d189e7545f0892db9ef2731b90d
- cloudpakName: IBM Cloud Pak for Integration
- productChargedKapsayıcıları: qmgr
- productCloudpakOran: '4:1'
- productID: 21dfe9a0f00f444f888756d835334909
- productName: Üretim Dışı için IBM MQ Advanced
- productMetric: VIRTUAL_PROCESSOR_CORE
- productVersion: 9.2.3.0

IBM Cloud Pak for Integration içinde, IBM App Connect Enterprise ' un konuşlandırmaları, IBM MQ için sınırlı bir yetki içerir. Bu tür durumlarda, IBM License Service ' in doğru kullanımı yakaladığı konusunda emin olmak için bu ek açıklamaların geçersiz kılınması gerekir. Bunu yapmak için, [“Kuyruk yöneticisi kaynaklarına özel ek açıklamalar ve etiketler eklenmesi” sayfa 112](#) içinde açıklanan yaklaşımı kullanın.

Örneğin, IBM App Connect Enterprise yetki alanı altında IBM MQ devreye alındıysa, aşağıdaki kod parçasında gösterilen yaklaşımı kullanın:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: mq4ace
  namespace: cp4i
spec:
  annotations:
    productMetric: FREE
```

Lisans ek açıklamalarının değiştirilmesi gerekmesinin diğer iki ortak nedeni vardır:

1. IBM MQ Advanced , başka bir IBM ürününün yetkisinde yer alır.
 - Bu durumda, IBM App Connect Enterprise için daha önce açıklanan yaklaşımını kullanın.
2. IBM MQ , bir IBM Cloud Pak for Integration lisansı altında devreye alınır.
 - Bir IBM Cloud Pak for Integration lisansınız varsa, bir kuyruk yöneticisini IBM MQ ya da IBM MQ Advanced oranı altında konuşlandırmanıza karar verebilirsiniz. IBM MQ oranı altında konuşlandırarsanız, Native HA ya da Advanced Message Security gibi gelişmiş yetenekleri kullanmadığınızdan emin olmanız gerekir.
 - Bu durumda, üretim kullanımı için aşağıdaki ek açıklamaları kullanın:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: mq4ace
  namespace: cp4i
spec:
  annotations:
    productID: c661609261d5471fb4ff8970a36bccea
    productCloudpakRatio: '4:1'
    productName: IBM MQ for Production
    productMetric: VIRTUAL_PROCESSOR_CORE
```

- Üretim dışı kullanım için aşağıdaki ek açıklamaları kullanın:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
```

```
metadata:
  name: mq4ace
  namespace: cp4i
spec:
  annotations:
    productID: 151bec68564a4a47a14e6fa99266deff
    productCloudpakRatio: '8:1'
    productName: IBM MQ for Non-Production
    productMetric: VIRTUAL_PROCESSOR_CORE
```

OpenShift > CP4I IBM MQ Operatorkullanarak kuyruk yöneticileri için yüksek kullanılabilirliğin yapılandırılması

Bu görev hakkında

Yordam

- [V 9.2.3](#)
“Yerel HA” sayfa 91.
- [V 9.2.3](#)
“Örnek: Yerel HA kuyruk yöneticisinin yapılandırılması” sayfa 93.
- [“Örnek: Çok eşgörünümlü bir kuyruk yöneticisinin yapılandırılması” sayfa 101.](#)

CP4I > V 9.2.3 > CD **Yerel HA**

Yerel HA, IBM MQ için bulut bloğu depolamasıyla kullanılmaya uygun bir yerel (yerleşik) yüksek kullanılabilirlik çözümdür.

Yerel HA yapılandırması, kurtarılabilir MQ verilerinin (örneğin, iletiler) birden çok depolama kümesi boyunca eşlendiği, depolama hatalarının kaybolmasını önleyen, yüksek düzeyde kullanılabilir bir kuyruk yöneticisi sağlar. Kuyruk yöneticisi birden çok çalışan eşgörünümünden oluşur; biri lider, diğerleri hata durumunda hızla devralmaya hazır, kuyruk yöneticisine ve iletilerine erişimi en üst düzeye çıkarır.

Yerel HA yapılandırması, her biri kuyruk yöneticisinin bir örneğine sahip üç Kubernetes bölmesinden oluşur. Bir yönetim ortamı, etkin kuyruk yöneticisidir, iletileri işleme ve kurtarma günlüğüne yazma. Kurtarma günlüğü ne zaman yazılırsa, etkin kuyruk yöneticisi verileri eşlemeler olarak bilinen diğer iki yönetim ortamı için gönderir. Her kopya, kendi kurtarma günlüğüne yazar, verileri kabul eder ve daha sonra, kendi kuyruk verilerini eşlenmiş kurtarma günlüğünden günceller. Etkin kuyruk yöneticisini çalıştıran pod başarısız olursa, kuyruk yöneticisinin eşleme eşgörünümlerinden biri etkin rolü devralıyor ve çalışmakta olan verileri içeriyor.

Günlük tipi 'eşlenmiş günlük' olarak bilinir. Eşlenmiş bir günlük, otomatik günlük yönetimi ve otomatik ortam görüntüleri etkin olduğunda, temelde doğrusal bir günlüktür. Bkz. [Günlüğe kaydetme tipleri](#). Doğrusal bir günlüğü yönetmek için kullandığınız eşlenmiş günlüğü yönetmek için aynı teknikleri kullanıyorsunuz.

Bir Kubernetes Service , TCP/IP istemci bağlantılarını yürürlükteki etkin yönetim ortamına yönlendirmek için kullanılır; bu, ağ trafiği için hazır olan tek bölge olarak tanımlanır. Bu durum, istemci uygulamasının farklı eşgörünümlerin farkında olması gerekmeksizin gerçekleşir.

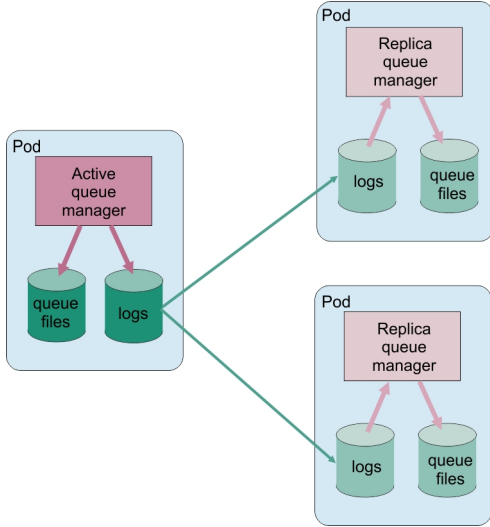
Üç bölme, bölünen beyin durumunun ortaya çıkma olasılığını büyük ölçüde azaltmak için kullanılır. İki bölmedeki yüksek kullanılabilirlikli sistem bölmeleri, iki bölme arasındaki bağlantı kesildiğinde ortaya çıkabilir. Bağlanırlık olmadan, her iki bölme de aynı anda kuyruk yöneticisini çalıştırabiliyordu ve farklı veriler toplanabiliyordu. Bağlantı geri yüklendiğinde, verilerin iki farklı sürümü (bir 'splitt-brain') olacaktır ve hangi veri kümesinin saklanacağını ve hangilerinin atılacağını belirlemek için el ile müdahale gerekir.

Yerel HA, bölünen beyin durumundan kaçınmak için yeterli çoğunluk ile üç pod sistemi kullanır. Diğer bölmelerden en az biriyle iletişim kurabilen kapsüller bir çekirdek oluşturur. Bir kuyruk yöneticisi, yalnızca yeterli yetersayı olan bir bölmedeki etkin eşgörünüm olabilir. Kuyruk yöneticisi, en az bir diğer modüle bağlı olmayan bir pod üzerinde etkin duruma gelemez; bu nedenle, aynı anda iki etkin eşgörünüm olamaz:

- Tek bir pod arızalanırsa, diğer iki bölmenin birinde kuyruk yöneticisi devralabilir. İki bölme başarısız olursa, pod yeterli sayıda çekirdek olmadığından kuyruk yöneticisi etkin eşgörünüm haline gelemmez (kalan bölme diğer iki bölme işleminin başarısız olup olmadığını ya da hala çalışır durumda olduğunu ve bağlantırlığı yitirdiğini gösteremez).
- Tek bir pod bağlantırlığı kaybederse, pod çekirdeğinin olmadığı için kuyruk yöneticisi bu pod üzerinde etkin duruma gelemmez. Kalan iki bölmenin birinde kuyruk yöneticisi devralabilir, bu da yeterli yetersayı içerir. Tüm bölmelerdeki bağlantırlığı kaybederse, kuyruk yöneticisi kapsüllerin hiçbirinde etkin duruma gelemmez, çünkü kapsüllerin hiçbiri yeterli sayıda çekirdek olamaz.

Etkin bir bölme başarısız olursa ve daha sonra kurtarılsa, grup eşleme rolündeki gruba yeniden katılabilir.

Aşağıdaki şekil, üç kapsayıcıda konuşlandırılan bir kuyruk yöneticisinin üç eşgörünümü ile tipik bir devreye alımın gösterilmesini göstermektedir.



Şekil 1. Yerel HA yapılandırması örneği

CP4I V 9.2.3 CD Configuring Native HA using the IBM MQ Operator

Yerel HA, QueueManager API kullanılarak yapılandırılır ve gelişmiş seçenekler bir INI dosyası kullanılarak kullanılabilir.

Yerel HA, QueueManager API 'nin `.spec.queueManager.availability` kullanılarak yapılandırılır; örneğin:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: nativeha-example
spec:
  license:
    accept: false
    license: L-RJON-C7QG3S
    use: Production
  queueManager:
    availability:
      type: NativeHA
    version: 9.2.5.0-r3
```

`.spec.queueManager.availability.type` alanı, NativeHA olarak ayarlanmalıdır.

Yerel HA, IBM MQ 9.2.3 ya da sonraki bir sürümü için kullanılabilir.

`.spec.queueManager.availability` altında, bir TLS güvenlik dizgisi ve şifrelemeleri, eşleme sırasında kuyruk yöneticisi eşgörünümleri arasında kullanılacak şekilde de yapılandırabilirsiniz. This is strongly recommended, and a step-by-step guide is available in [“Örnek: Yerel HA kuyruk yöneticisinin yapılandırılması” sayfa 93.](#)

İlgili başvurular

“Örnek: Yerel HA kuyruk yöneticisinin yapılandırılması” sayfa 93


This example shows how you deploy a queue manager using the native high availability feature into the Red Hat OpenShift Container Platform (OCP) using the IBM MQ Operator.

 Örnek: Yerel HA kuyruk yöneticisinin yapılandırılması

This example shows how you deploy a queue manager using the native high availability feature into the Red Hat OpenShift Container Platform (OCP) using the IBM MQ Operator.

Başlamadan önce

Bu örneği tamamlamak için, önce aşağıdaki önkoşulları tamamlamanız gerekir:

- IBM MQ client' u kurun ve kurulu samp/bin ve bin dizinlerini PATHdizinimize ekleyin. İstemci, bu örnek için gereken **runmqacm**, **amqsputc** ve **amqsgetc** uygulamalarını sağlar. IBM MQ client ürününü aşağıdaki gibi kurun:
 -   Windows ve Linux için: İşletim sisteminize ilişkin IBM MQ yeniden dağıtılabilir istemcisini <https://ibm.biz/mq92redistclientsolanağ>ndan kurun.
 -  Mac için: IBM MQ MacOS Toolkit dosyasını karşıdan yükleyin ve ayarlayın. Bkz. <https://ibm.biz/mqdevmacclient>.
- İşletim sisteminize ilişkin OpenSSL aracını kurun. Önceden özel bir anahtar ve sertifika yoksa, kuyruk yöneticisi için kendinden imzalı bir sertifika oluşturmanız gerekir.
- Create a Red Hat OpenShift Container Platform (OCP) project/namespace for this example and follow the steps in the task “Preparing your Red Hat OpenShift project for IBM MQ” sayfa 78
- Komut satırında, İşletmen Denetim Panosu kümesinde oturum açın ve az önce oluşturduğunuz ad alanına geçin.
- Ensure the IBM MQ Operator is installed and available in the namespace.
- Kuyruk yöneticiniz tarafından kullanılacak OCP ' de varsayılan bir depolama sınıfı yapılandırın. Bu eğitimde varsayılan bir depolama sınıfı ayarlamadan tamamlamak istiyorsanız bkz. [Not 2: Varsayılan olmayan bir depolama sınıfı kullanılması](#).

Bu görev hakkında

Yerel HA kuyruk yöneticileri etkin ve iki eşleme Kubernetes Pod içerir. Bunlar, tam olarak üç eşleme ve bir Kubernetes Kalıcı Birim kümesi ile Kubernetes Stateful Set 'in bir parçası olarak çalıştırılıyor. Yerel HA kuyruk yöneticilerine ilişkin daha fazla bilgi için bkz. [“Taşıyıcılarda IBM MQ için yüksek kullanılabilirlik”](#) sayfa 15.

Bu örnek, kalıcı saklama alanı kullanan ve TLS ile yapılandırılmış bir yerel HA kuyruk yöneticisini tanımlayan bir özel kaynak YAML sağlar. Kuyruk yöneticisini OCP ' ye konuşlandırdıktan sonra, etkin kuyruk yöneticisi kapsülünün hata benzetimi gerçekleştirdiniz. Otomatik kurtarma gerçekleşir ve başarısızlığın ardından iletilerin yerleştirilerek ve iletilmesiyle başarılı olduğunu kanıtlayın.

Örnek

MQ sunucusu için bir TLS özel anahtarı ve sertifikaları oluşturma

Kuyruk yöneticisi için kendinden onaylı bir sertifika yaratabilir ve sertifikayı, istemci için güvenli depo olarak işlev görebilmesi için anahtar veritabanına ekleyebilirsiniz. Zaten özel bir anahtar ve sertifikanız varsa, bunları kullanabilirsiniz. Geliştirme amaçları için yalnızca kendinden onaylı sertifikalar kullanmanız gerektiğini unutmayın.

Yürürlükteki dizinde kendinden onaylı bir özel anahtar ve genel sertifika yaratmak için aşağıdaki komutu çalıştırın:

```
openssl req -newkey rsa:2048 -nodes -keyout tls.key -subj "/CN=localhost" -x509 -days 3650
-out tls.crt
```

Yerel HA tarafından iç kullanım için bir TLS özel anahtarı ve sertifikaları oluşturma

Yerel HA kuyruk yöneticisinde bulunan üç bölme, verileri ağ üzerinden eşkopyalar. Dahili olarak eşlenirken kullanmak üzere kendinden imzalı bir sertifika oluşturabilirsiniz. Geliştirme amaçları için yalnızca kendinden onaylı sertifikalar kullanmanız gerektiğini unutmayın.

Yürürlükteki dizinde kendinden onaylı bir özel anahtar ve genel sertifika yaratmak için aşağıdaki komutu çalıştırın:

```
openssl req -newkey rsa:2048 -nodes -keyout nativeha.key -subj "/CN=localhost" -x509 -days
3650 -out nativeha.crt
```

Kuyruk yöneticisi genel anahtarını bir istemci anahtar veritabanına ekle

İstemci anahtarı veritabanı istemci uygulaması için güvenli depo olarak kullanılır.

İstemci anahtar veri tabanını yaratın:

```
runmqakm -keydb -create -db clientkey.kdb -pw password -type cms -stash
```

Önceden oluşturulan genel anahtarı istemci anahtarı veritabanına ekle:

```
runmqakm -cert -add -db clientkey.kdb -label mqservercert -file tls.crt -format ascii
-stashed
```

Kuyruk yöneticisi konuşlandırması için TLS sertifikalarını içeren bir güvenlik dizgisi yarat

Böylece, kuyruk yöneticinizin anahtarı ve sertifikayı gönderip uygulayabilir, yukarıda yaratılan dosyalara gönderme yapan bir Kubernetes TLS güvenlik dizgisi yaratabilirler. Bunu yaparken, bu göreve başlamadan önce oluşturduğunuz ad alanında bulunduğunuzdan emin olun.

```
oc create secret tls example-ha-secret --key="tls.key" --cert="tls.crt"
```

İç yerli HA TLS sertifikasını ve anahtarını içeren bir güvenlik dizgisi yarat

Böylece, kuyruk yöneticinizin anahtarı ve sertifikayı gönderip uygulayabilir, yukarıda yaratılan dosyalara gönderme yapan bir Kubernetes TLS güvenlik dizgisi yaratabilirler. Bunu yaparken, bu göreve başlamadan önce oluşturduğunuz ad alanında bulunduğunuzdan emin olun.

```
oc create secret tls example-ha-secret-internal --key="nativeha.key" --cert="nativeha.crt"
```

MQSC komutlarını içeren bir yapılandırma eylemi yaratır

Yeni bir kuyruk ve bir SVRCONN Kanalı yaratmak ve yalnızca *hiç kimse* adı verilen kullanıcıları engelleyerek kanala erişime izin veren bir kanal kimlik doğrulaması kaydı eklemek için, MQSC komutlarını içeren bir Kubernetes yapılandırma eylemi yaratın.

Bu yaklaşımın yalnızca geliştirme amaçları için kullanılması gerektiğini unutmayın.

Daha önce oluşturduğunuz ad alanında olduğunuzu doğrulayın (bkz. [“Başlamadan önce” sayfa 93](#)), ardından OCP UI 'ye aşağıdaki YAML' yi girin ya da komut satırını kullanın:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: example-mi-configmap
data:
  tls.mqsc: |
    DEFINE QLOCAL('EXAMPLE.QUEUE') DEFPSIST(YES) REPLACE
    DEFINE CHANNEL(HAQMCHL) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCAUTH(OPTIONAL)
    SSLCIPH('ANY_TLS12_OR_HIGHER')
    SET CHLAUTH(HAQMCHL) TYPE(BLOCKUSER) USERLIST('nobody') ACTION(ADD)
```

Yönlendirmeyi yapılandır

IBM MQ 9.2.1 ya da daha sonraki bir yayın düzeyinde IBM MQ client ya da araç takımı kullanıyorsanız, kuyruk yöneticisi için bir kuyruk yöneticisi yapılandırma dosyası (INI dosyası) kullanarak yönlendirme yöneticisi olarak yönlendirmeyi yapılandırabilirsiniz. Dosya içinde, *OutboundSNI* değişkenini, kanal adı yerine anasistem adı temelinde yönlendirecek şekilde ayarladınız.

Şu metni tam olarak içeren `mqclient.ini` adlı, komutları çalıştırdığınız dizinde bir dosya yaratın:

```
SSL:
  OutboundSNI=HOSTNAME
```

Bu INI kütüğündeki hiçbir değeri değiştirmeyin. Örneğin, `HOSTNAME` dizgisi değiştirilmemelidir.

Daha fazla ayrıntı için [İstemci yapılandırma dosyasının SSL stanzası](#) başlıklı konuya bakın.

IBM MQ 9.2.1' dan önceki bir IBM MQ client ya da araç takımı kullanıyorsanız, önceki yapılanış dosyası yerine bir OCP rotası yaratmanız gerekir. [Not 1: Rotanın yaratılması](#) başlıklı konu başlıklı konuyu izleyin.

Kuyruk yöneticisini konuşlandır

Önemli: Bu örnekte, kuyruk yöneticisinde yetkilendirmeyi devre dışı bırakmak için `MQSNOAUT` değişkenini kullanırsınız; bu değişken, bir istemciyi TLS 'yi kullanarak bağlamak için gereken adımlara odaklanabilmemizi sağlar. This is not recommended in a production deployment of IBM MQ, because it causes any applications connecting to have full administrative powers, with no mechanism to lower the permissions for individual applications.

Aşağıdaki YAML 'yi kopyalayın ve güncelleştirin.

- Doğru lisansın belirtildiğinden emin olun. Bkz. mq.ibm.com/v1beta1 için lisans başvurusu. In IBM Cloud Pak for Integration 2021.1.1, the license must be the evaluation license L-RJON-BYRMYW
- Accept the license by changing `false` to `true`.

Kuyruk yöneticisi özel kaynağı YAML:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: nativeha-example
spec:
  license:
    accept: false
    license: L-RJON-C7QG3S
    use: Production
  queueManager:
    name: HAEXAMPLE
    availability:
      type: NativeHA
    tls:
      secretName: example-ha-secret-internal
  mqsc:
    - configMap:
        name: example-mi-configmap
        items:
          - tls.mqsc
  template:
    pod:
      containers:
        - env:
            - name: MQSNOAUT
              value: 'yes'
          name: qmgr
  version: 9.2.5.0-r3
  pki:
    keys:
      - name: example
        secret:
          secretName: example-ha-secret
          items:
            - tls.key
            - tls.crt
```

Daha önce yaratılan ad alanında olduğunuzu doğrulayan, güncellenmiş YAML 'yi konuşlandırmanızı, Red Hat OpenShift Container Platform web konsolunu, komut satırını ya da IBM Cloud Pak for Integration Platform Navigator komutunu kullanmanızı sağlar.

Yerel HA kuyruk yöneticisini yapılandırdıktan sonra, kuyruk yöneticisinin kullanılabilir olması gerektiği için kısa bir gecikme olur.

Geçerlilik Denetimi

Bu bölümde, kuyruk yöneticisinin beklendiği gibi davrandığını doğrularız.

Kuyruk yöneticisinin çalışır durumda olduğunu doğrulayın

Kuyruk yöneticisi şimdi konuşlandırılıyor. Confirm it is in Running state before proceeding. Örneğin:

```
oc get qmgr nativeha-example
```

Kuyruk yöneticisiyle bağlantıyı test edin

Kuyruk yöneticisinin tek yönlü TLS iletişimi için yapılandırılıp yapılandırıldığı doğrulamak için, **amqsputc** ve **amqsgetc** örnek uygulamalarını kullanın:

Kuyruk yöneticisi anasistem adını bul

`nativeha-example-ibm-mq-qm`rotasına ilişkin kuyruk yöneticisi anasistem adını bulmak için aşağıdaki komutu çalıştırın. Anasistem adı, HOST alanında döndürülür.

```
oc get routes nativeha-example-ibm-mq-qm
```

Kuyruk yöneticisi ayrıntılarını belirtin

Create a file `CCDT`. JSON that specifies the queue manager details. Anasistem değerini, önceki adımın döndürdüğü anasistem adı ile değiştirin.

```
{
  "channel":
  [
    {
      "name": "HAQMCHL",
      "clientConnection":
      {
        "connection":
        [
          {
            "host": "<host from previous step>",
            "port": 443
          }
        ],
        "queueManager": "HAEXAMPLE"
      },
      "transmissionSecurity":
      {
        "cipherSpecification": "ECDHE_RSA_AES_128_CBC_SHA256"
      },
      "type": "clientConnection"
    }
  ]
}
```

Ortam değişkenlerini dışa aktar

İşletim sisteminiz için uygun şekilde aşağıdaki ortam değişkenlerini dışa aktarın. Bu değişkenler **amqsputc** ve **amqsgetc** tarafından okunacaktır.

Sisteminizdeki kütüklerin yolunu güncelleyin:

```
export MQCCDTURL='<full_path_to_file>/CCDT.JSON'
export MQSSLKEYR='<full_path_to_file>/clientkey'
```

İletileri kuyruğa koy

Aşağıdaki komutu çalıştırın:

```
amqsputc EXAMPLE.QUEUE HAEXAMPLE
```

Kuyruk yöneticisiyle bağlantı başarılı olursa, aşağıdaki yanıt çıkışa gelir:

```
target queue is EXAMPLE.QUEUE
```

Kuyruğa birkaç ileti koyun, bir metin girerek her seferinde **Enter** tuşuna basın.

İşlemi tamamlamak için iki kez **Enter** tuşuna basın.

İletileri kuyruktan al

Aşağıdaki komutu çalıştırın:


```
amqsgetc EXAMPLE.QUEUE HAEXAMPLE
```

Önceki adımda eklediğiniz iletiler tüketildi ve çıkışa yazılıyor.

Birkaç saniye sonra, komuta kapanır.

Etkin kapsülü başarısız olmasına zorlar

Kuyruk yöneticisinin otomatik olarak kurtarılması işlemini doğrulamak için, bir pod hatası benzetimi gerçekleştirin:

Etkin ve beklemedeki bölmeleri görüntüle

Aşağıdaki komutu çalıştırın:

```
oc get pods --selector app.kubernetes.io/instance=nativeha-example
```

READY alanında, eşleme bölmelerinin 0/1değerini döndürürken etkin bölme 1/1değerini geri döndüreceğini unutmayın.

Etkin kapsülü sil

Etkin bölmenin tam adını belirterek aşağıdaki komutu çalıştırın:

```
oc delete pod nativeha-example-ibm-mq-<value>
```

Pod durumunu yeniden görüntüle

Aşağıdaki komutu çalıştırın:

```
oc get pods --selector app.kubernetes.io/instance=nativeha-example
```

Kuyruk yöneticisi durumunu görüntüle

Diğer bölmelerden birinin tam adını belirterek aşağıdaki komutu çalıştırın:

```
oc exec -t Pod -- dspmq -o nativeha -x -m HAEXAMPLE
```

Durumu görmemiz gerekir; örneğin, etkin yönetim ortamının değiştiğini gösterir.

```
QMNAME(HAEXAMPLE) ROLE(Active) INSTANCE(inst1) INSYNC(Yes) QUORUM(3/3)
INSTANCE(inst1) ROLE(Active) REPLADDR(9.20.123.45) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
```

İletileri yeniden al ve yeniden al

Beklemedeki bölme etkin bölme olduktan sonra (READY alan değeri 1/1olduktan sonra), daha önce açıklandığı gibi, kuyruk yöneticisine ileti koymak için aşağıdaki komutları yeniden kullanın ve ardından kuyruk yöneticisinden iletileri alın:

```
amqsputc EXAMPLE.QUEUE HAEXAMPLE
```

```
amqsgetc EXAMPLE.QUEUE HAEXAMPLE
```

Tebrikler, yerel bir HA kuyruk yöneticisini başarıyla yerleştirdiniz ve bir pod hatasından otomatik olarak kurtarılabildiğinizi gösterdiniz.

Ek bilgi

Not 1: Yol yaratılması

IBM MQ 9.2.1' dan önceki bir IBM MQ client ya da araç takımını kullanıyorsanız, bir Rota yaratmanız gerekir.

Rotayı oluşturmak için, daha önce oluşturduğunuz ad alanında olduğunuzu doğrulayın (bkz. “Başlamadan önce” sayfa 93), daha sonra Red Hat OpenShift Container Platform web konsolunda aşağıdaki YAML 'yi girin ya da komut satırını kullanın:

```
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: example-mi-route
spec:
  host: hamqchl.ch1.mq.ibm.com
  to:
    kind: Service
    name: nativeha-example-ibm-mq
  port:
    targetPort: 1414
  tls:
    termination: passthrough
```

Note that the Red Hat OpenShift Container Platform Router uses SNI for routing requests to the IBM MQ queue manager. MQSC komutlarını içeren yapılandırma eylemi' da belirtilen kanal adını değiştirirseniz, anasistem alanını burada ve Kuyruk yöneticisi ayrıntılarını belirten CCDT . JSON dosyası içinde de değiştirmeniz gerekir. Daha fazla bilgi için bkz “Red Hat OpenShift kümesi dışından bir kuyruk yöneticisine bağlanmak için bir Rotanın yapılandırılması” sayfa 107.

Not 2: Varsayılan olmayan bir depolama sınıfı kullanılması

This example expects a default storage class to have been configured in Red Hat OpenShift Container Platform, therefore no storage information is required in the kuyruk yöneticisi özel kaynağı YAML. Varsayılan olarak yapılandırılmış bir depolama sınıfınız yoksa ya da farklı bir depolama sınıfı kullanmak istiyorsanız, spec.queueManager.storagealtına defaultClass: <storage_class_name> ekleyin.

Depolama sınıfı adı, zaten var olan bir depolama sınıfının adıyla tam olarak eşleşmelidir. Yani, bu, oc get storageclass komutunun döndürdüğü adla eşleşmelidir. Bu, ReadWriteMany' i de desteklemelidir. Daha fazla bilgi için bkz “IBM MQ Operatör ile ilgili depolama konuları” sayfa 11.

İlgili görevler

“IBM MQ sertifikalı kapsayıcılar için Yerel HA kuyruk yöneticilerinin durumunu görüntüleme” sayfa 98
IBM MQ sertifikalı kapsayıcılar için, çalışmakta olan Pds 'lerden birinde **dspmqr** komutunu çalıştırarak Yerel HA eşgörünümlerinin durumunu görüntüleyebilirsiniz.

 IBM MQ sertifikalı kapsayıcılar için Yerel HA kuyruk yöneticilerinin durumunu görüntüleme

IBM MQ sertifikalı kapsayıcılar için, çalışmakta olan Pds 'lerden birinde **dspmqr** komutunu çalıştırarak Yerel HA eşgörünümlerinin durumunu görüntüleyebilirsiniz.

Bu görev hakkında

Önemli:

Bir kuyruk yöneticisi eşgörünümünün çalışma durumunu görüntülemek için, çalışmakta olan PDS 'lerden birinde **dspmqr** komutunu kullanabilirsiniz. Döndürülen bilgiler, eşgörünümün etkin mi, yoksa eşleme mi olduğuna bağlıdır. Etkin örnek tarafından sağlanan bilgiler kesin, eşleme düğümlerinden alınan bilgiler güncel olmayabilir.

Aşağıdaki işlemleri gerçekleştirebilirsiniz:

- Geçerli düğümdeki kuyruk yöneticisi örneğinin etkin mi, yoksa eşleme mi olduğunu görüntüler.
- Yürürlükteki düğümdeki yönetim ortamının Yerel HA çalışma durumunu görüntüler.
- Yerel HA yapılandırmasındaki tüm üç örneğin çalışma durumunu görüntüler.

Yerel HA yapılandırma durumunu raporlamak için aşağıdaki durum alanları kullanılır:

Görev

Eşgörünümün geçerli rolünü belirtir ve Active, Replicaya da Unknown' den biridir.

yönetim ortamı

The name provided for this instance of the queue manager when it was created using the **-lr** option of the **crtmqm** command.

İNSYNC

Yönetim ortamının gerektiğinde etkin yönetim ortamı olarak kabul edilip edilemeyeceğini belirtir.

ÇEKİRDEK

Yetersayı durumunu *number_of_instances_in-sync/number_of_instances_confitcale* biçiminde raporlar.

YANITLA

Kuyruk yöneticisi örneğinin eşleme adresi.

CONNACTV

Düğümün etkin eşgörünümle bağlı olup olmadığını belirtir.

BACKLOG

Eşgörünümün arkasında olduğu KB sayısını gösterir.

CONNINST

Belirtilen yönetim ortamının bu yönetim ortamına bağlı olup olmadığını gösterir.

TARİH DEĞERİ

Bu bilgilerin en son güncellendiği tarihi gösterir (hiçbir zaman güncellenmediyse boş değer).

ALTTIME

Bu bilgilerin en son güncellendiği zamanı belirtir (hiçbir zaman güncellenmediyse boş değer).

Yordam

- Kuyruk yöneticinizin bir parçası olan bölmeleri bulun.

```
oc get pod --selector app.kubernetes.io/instance=nativeha-qm
```

- Kapsüllerden birinde `dspmqr` komutunu çalıştırın.

```
oc exec -t Pod dspmqr
```

```
oc ish Pod
```

`dspmqr` 'u doğrudan çalıştırabileceğiniz etkileşimli bir kabuk için.

- Bir kuyruk yöneticisi örneğinin etkin eşgörünüm olarak mı, yoksa eşleme olarak mı çalıştırılacağını belirlemek için:

```
oc exec -t Pod dspmqr -o status -m QMgrName
```

BOB adlı bir kuyruk yöneticisinin etkin bir eşgörünümü aşağıdaki durumu bildirecektir:

```
QMNAME(BOB)          STATUS(Running)
```

BOB adlı bir kuyruk yöneticisinin eşleme eşgörünümü aşağıdaki durumu bildirecektir:

```
QMNAME(BOB)          STATUS(Replica)
```

Etkin olmayan bir yönetim ortamı aşağıdaki durumu bildirecektir:

```
QMNAME(BOB)          STATUS(Ended Immediately)
```

- Belirtilen bölmede eşgörünümün yerel HA çalışma durumunu belirlemek için:

```
oc exec -t Pod dspmqr -o nativeha -m QMgrName
```

BOB adlı bir kuyruk yöneticisinin etkin örneği, aşağıdaki durumu bildirebilir:

```
QMNAME(BOB)          ROLE(Active) INSTANCE(inst1) INSYNC(Yes) QUORUM(3/3)
```

BOB adlı bir kuyruk yöneticisinin eşleme örneği aşağıdaki durumu bildirebilir:

```
QMNAME(BOB)          ROLE(Replica) INSTANCE(inst2) INSYNC(Yes) QUORUM(2/3)
```

BOB adlı bir kuyruk yöneticisinin etkin olmayan bir eşgörünümü aşağıdaki durumu bildirebilir:

```
QMNAME(BOB)          ROLE(Unknown) INSTANCE(inst3) INSYNC(no) QUORUM(0/3)
```

- Yerel HA yapılandırmasındaki tüm eşgörünümlerin Yerel HA işletim durumunu belirlemek için:

```
oc exec -t Pod dspmq -o nativeha -x -m QMgrName
```

Bu komutu, kuyruk yöneticisi BOB ' un etkin örneğini çalıştıran düğümde yayınyorsanız, aşağıdaki durumu alabilirsiniz:

```
QMNAME(BOB)          ROLE(Active) INSTANCE(inst1) INSYNC(Yes) QUORUM(3/3)
INSTANCE(inst1) ROLE(Active) REPLADDR(9.20.123.45) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
```

Bu komutu, kuyruk yöneticisi BOB ' un eşleme örneğini çalıştıran bir düğümde yayınyorsanız, eşlemelerden birinin geride kaldığını belirten aşağıdaki durumu alabilirsiniz:

```
QMNAME(BOB)          ROLE(Replica) INSTANCE(inst2) INSYNC(Yes) QUORUM(2/3)
INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst1) ROLE(Active) REPLADDR(9.20.123.45) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(No) BACKLOG(435)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
```

Bu komutu, kuyruk yöneticisi BOB ' un etkin olmayan bir örneğini çalıştıran bir düğümde yayınyorsanız, aşağıdaki durumu alabilirsiniz:

```
QMNAME(BOB)          ROLE(Unknown) INSTANCE(inst3) INSYNC(no) QUORUM(0/3)
INSTANCE(inst1) ROLE(Unknown) REPLADDR(9.20.123.45) CONNACTV(Unknown) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATA() ALTTIME()
INSTANCE(inst2) ROLE(Unknown) REPLADDR(9.20.123.46) CONNACTV(Unknown) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATA() ALTTIME()
INSTANCE(inst3) ROLE(Unknown) REPLADDR(9.20.123.47) CONNACTV(No) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATA() ALTTIME()
```

Yönetim ortamları hala etkin olan ve eşlemeler olan bir yönetim ortamı üzerinde çalışmaya devam ederken komutu yayınlıyorsanız, aşağıdaki durumu alırsınız:

```
QMNAME(BOB)          STATUS(Negotiating)
```

İlgili başvurular

[dspmq \(kuyruk yöneticilerini görüntüle\) komutu](#)

“Örnek: Yerel HA kuyruk yöneticisinin yapılandırılması” sayfa 93

This example shows how you deploy a queue manager using the native high availability feature into the Red Hat OpenShift Container Platform (OCP) using the IBM MQ Operator.

CP4I **V 9.2.3** **CD** *Yerel HA için gelişmiş ayarlama*

Zamanlamaları ve aralıkları ayarlamak için gelişmiş ayarlar. Varsayılan değerler sisteminizin gereksinimleriyle eşleşmediği bilinmiyorsa, bu ayarları kullanmaya gerek kalmamalıdır.

Yerel HA 'nın yapılandırılmasına ilişkin temel seçenekler, IBM MQ Operator ' in temel kuyruk yöneticisi INI dosyalarını sizin için yapılandırmak için kullandığı QueueManager API kullanılarak işlenir. Yalnızca bir INI dosyası kullanılarak, [NativeHALocalYönetim Ortamı-Stanzaaltında yapılandırılabilir olan bazı gelişmiş seçenekler vardır](#). INI dosyasının yapılandırılmasına ilişkin ek bilgi için bkz. “[Örnek: MQSC ve INI dosyaları sağlar](#)” sayfa 85 .

HeartbeatInterval

Sağlıklı işletim bildirim aralığı, bir Native HA kuyruk yöneticisinin etkin bir yönetim ortamının milisaniye olarak ne sıklıkta ağ sinyali gönderdiğini tanımlar. Sağlıklı işletim bildirim aralığı değeri 500 (0.5 saniye) ile 60000 (1 dakika) arasında bir değer, bu aralığın dışındaki bir değer kuyruk yöneticisinin başlamasına neden olur. Bu öznitelik atılırsa, varsayılan değer 5000 (5 saniye) kullanılır. Her eşgörünüm aynı sağlıklı işletim bildirim aralığını kullanmalıdır.

HeartbeatTimeout

Sağlıklı işletim bildirim zaman aşımı, etkin örneğe yanıt vermemeye karar vermeden önce Yerel HA kuyruk yöneticisinin eşleme eşgörünümünün ne kadar süreyle bekleyeceğini tanımlar. Sağlıklı işletim bildirim aralığı zaman aşımı değerinin geçerli aralığı 500 (0.5 saniye) ile 120000 (2 dakika) arasındadır. Sağlıklı işletim bildirim zaman aşımının değeri, sağlıklı işletim bildirim aralığından büyük ya da ona eşit olmalıdır.

Geçersiz bir değer, kuyruk yöneticisinin başlatılmamasına neden olur. Bu öznitelik atılırsa, yeni bir etkin yönetim ortamı seçmek için işlemi başlatmadan önce 2 x HeartbeatInterval için bekleyen bir bekleme işlemi bekler. Her eşgörünüm aynı sağlıklı işletim bildirim zaman aşımını kullanmalıdır.

RetryInterval

Yeniden deneme aralığı, Native HA kuyruk yöneticisinin ne sıklıkta başarısız bir eşleme bağlantısını yeniden denemesi gerektiğini milisaniye olarak tanımlar. Yeniden deneme aralığı için geçerli aralık 500 (0.5 saniye) ile 120000 (2 dakika) arasındadır. Bu öznitelik atılırsa, başarısız bir eşleme bağlantısını yeniden denemeden önce 2 x HeartbeatInterval için bekleyen bir bekleme işlemi bekler.

CP4I Yerel HA kuyruk yöneticilerinin sona erdiriliyor

Yerel bir HA grubunun bir parçası olan etkin ya da eşleme kuyruk yöneticisini sona erdirmek için **endmqm** komutunu kullanabilirsiniz.

Yordam

- Bir kuyruk yöneticisinin etkin yönetim ortamını sona erdirmek için, bu belgelerin Configuring (Yapılandırma) bölümünde [Finding Native HA queue manager \(Yerel HA kuyruk yöneticilerinin sona erdirilmesi\)](#)

CP4I **V 9.2.2** **CD** *Evaluating the Native HA feature in IBM Cloud Pak for Integration*

2021.1.1

IBM Cloud Pak for Integration 2021.1.1 Yerel HA değerlendirme dönemi sona erdi. Please use the updated Native HA feature that is available from IBM Cloud Pak for Integration 2021.2.1, using IBM MQ Operator 1.6 or higher with IBM MQ 9.2.3 or higher.

İlgili görevler

“IBM MQ sertifikalı kapsayıcılar için Yerel HA kuyruk yöneticilerinin durumunu görüntüleme” sayfa 98
IBM MQ sertifikalı kapsayıcılar için, çalışmakta olan Pds ' lerden birinde **dspm** komutunu çalıştırarak Yerel HA eşgörünümlerinin durumunu görüntüleyebilirsiniz.

İlgili başvurular

“Örnek: Yerel HA kuyruk yöneticisinin yapılandırılması” sayfa 93

This example shows how you deploy a queue manager using the native high availability feature into the Red Hat OpenShift Container Platform (OCP) using the IBM MQ Operator.

OpenShift **CP4I** **Örnek: Çok eşgörünümlü bir kuyruk yöneticisinin**

yapılandırılması

This example shows how you deploy a multi-instance queue manager into the Red Hat OpenShift Container Platform (OCP) using the IBM MQ Operator. Bu örnekte, bir örnek istemci ile kuyruk yöneticisi arasında TLS iletişimi için de bir yöntem yapılandırın. Bu örnek, benzetimli bir bölme hatasından önce ve sonra iletileri koyarak ve iletileri alarak başarılı bir yapılandırmayı gösterir.

Başlamadan önce

Bu örneği tamamlamak için, önce aşağıdaki önkoşulları tamamlamanız gerekir:

- IBM MQ client' u kurun ve kurulu samp/bin ve bin dizinlerini *PATH*dizininize ekleyin. İstemci, bu örnek için gereken **runmqacm**, **amqspuic** ve **amqsgetc** uygulamalarını sağlar. IBM MQ client ürününü aşağıdaki gibi kurun:
 -   Windows ve Linux için: İşletim sisteminize ilişkin IBM MQ yeniden dağıtılabilir istemcisini <https://ibm.biz/mq92redistclientsolanağında> kurun.
 -  Mac için: IBM MQ MacOS Toolkit dosyasını karşıdan yükleyin ve ayarlayın. Bkz. <https://developer.ibm.com/tutorials/mq-macos-dev/>.
- İşletim sisteminize ilişkin OpenSSL aracını kurun. Önceden özel bir anahtar ve sertifika yoksa, kuyruk yöneticisi için kendinden imzalı bir sertifika oluşturmanız gerekir.
- Bu örnek için bir Red Hat OpenShift Container Platform (OCP) projesi/ad alanı yaratın.
- Komut satırında, İşletmen Denetim Panosu kümesinde oturum açın ve yukarıdaki ad alanına geçin.
- IBM MQ Operator ' in yukarıdaki ad alanında kurulu ve kullanılabilir olduğundan emin olun.
- Kuyruk yöneticiniz tarafından kullanılacak OCP ' de varsayılan bir depolama sınıfı yapılandırın. Bu eğitmeni varsayılan bir depolama sınıfı ayarlamadan tamamlamak istiyorsanız bkz. [Not 2: Varsayılan olmayan bir depolama sınıfı kullanılması](#).

Bu görev hakkında

Çok eşgörünümlü kuyruk yöneticileri etkin ve beklemedeki bir Kubernetes Pod 'unu içerir. Bunlar, tam olarak iki eşle ve bir Kubernetes Kalıcı Birim kümesiyle Kubernetes Stateful Set 'in bir parçası olarak çalıştırılıyor. Çok eşgörünümlü kuyruk yöneticilerine ilişkin daha fazla bilgi için bkz. [“Taşıyıcılarda IBM MQ için yüksek kullanılabilirlik” sayfa 15](#).

Bu örnek, kalıcı depolamaya sahip çok eşgörünümlü bir kuyruk yöneticisini tanımlayan ve TLS ile yapılandırılmış bir özel kaynak YAML sağlar. Kuyruk yöneticisini OCP ' ye konuşlandırdıktan sonra, etkin kuyruk yöneticisi kapsülünün hata benzetimi gerçekleştirdiniz. Otomatik kurtarma gerçekleşir ve başarısızlığın ardından iletilerin yerleştirilerek ve iletilmesiyle başarılı olduğunu kanıtlayın.

Örnek

MQ sunucusu için bir TLS özel anahtarı ve sertifikaları oluşturma

Bu kısım, kuyruk yöneticisi için kendinden imzalı sertifika yaratılmasını ve istemcinin istemci için güvenli depo olarak işlev görmeleri için anahtar veri tabanına nasıl ekleneceğini belgelemektedir. Zaten özel bir anahtar ve sertifikanız varsa, bunları kullanabilirsiniz. Geliştirme amaçları için yalnızca kendinden onaylı sertifikalar kullanmanız gerektiğini unutmayın.

Yürürlükteki dizinde kendinden onaylı bir özel anahtar ve genel sertifika yaratmak için aşağıdaki komutu çalıştırın:

```
openssl req -newkey rsa:2048 -nodes -keyout tls.key -subj "/CN=localhost" -x509 -days 3650 -out tls.crt
```

Kuyruk yöneticisi genel anahtarını bir istemci anahtar veritabanına ekle

İstemci anahtarı veritabanı istemci uygulaması için güvenli depo olarak kullanılır.

İstemci anahtar veri tabanını yaratın:

```
runmqacm -keydb -create -db clientkey.kdb -pw password -type cms -stash
```

Önceden oluşturulan genel anahtarı istemci anahtarı veritabanına ekle:

```
runmqacm -cert -add -db clientkey.kdb -label mqservercert -file tls.crt -format ascii -stashed
```

Kuyruk yöneticisi konuşlandırması için TLS sertifikalarını içeren bir güvenlik dizgisi yarat

Böylece, kuyruk yöneticinizin anahtarı ve sertifikayı gönderip uygulayabilir, yukarıda yaratılan dosyalara gönderme yapan bir Kubernetes TLS güvenlik dizgisi yaratabilirler. Bunu yaparken, bu göreve başlamadan önce oluşturduğunuz ad alanında bulunduğunuzdan emin olun.

```
oc create secret tls example-mi-secret --key="tls.key" --cert="tls.crt"
```

MQSC komutlarını içeren bir yapılandırma eşlemi yaratır

Yeni bir kuyruk ve bir SVRCONN Kanalı yaratmak ve yalnızca *hiç kimse* adı verilen kullanıcıları engelleyerek kanala erişime izin veren bir kanal kimlik doğrulaması kaydı eklemek için, MQSC komutlarını içeren bir Kubernetes yapılandırma eşlemi yaratın.

Bu yaklaşımın yalnızca geliştirme amaçları için kullanılması gerektiğini unutmayın.

Daha önce oluşturduğunuz ad alanında olduğunuzu doğrulayın (bkz. "[Başlamadan önce](#)" sayfa 102), ardından OCP UI 'ye aşağıdaki YAML' yi girin ya da komut satırını kullanın:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: example-mi-configmap
data:
  tls.mqsc: |
    DEFINE QLOCAL('EXAMPLE.QUEUE') DEFPSIST(YES) REPLACE
    DEFINE CHANNEL(MIQMCHL) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCAUTH(OPTIONAL)
    SSLCIPH('ANY_TLS12_OR_HIGHER')
    SET CHLAUTH(MIQMCHL) TYPE(BLOCKUSER) USERLIST('nobody') ACTION(ADD)
```

Yönlendirmeyi yapılandır

IBM MQ 9.2.1 ya da daha sonraki bir yayın düzeyinde IBM MQ client ya da araç takımı kullanıyorsanız, kuyruk yöneticisi için bir kuyruk yöneticisi yapılandırma dosyası (INI dosyası) kullanarak yönlendirme yöneticisi olarak yönlendirmeyi yapılandırabilirsiniz. Dosya içinde, *OutboundSNI* değişkenini, kanal adı yerine anasistem adı temelinde yönlentilecek şekilde ayarladınız.

Şu metni içeren 'mqclient.ini' adlı dizinde çalıştırdığınız komutların bulunduğu dizinde bir dosya yaratın:

```
## Module Name: mqclient.ini ##
## Type      : IBM MQ MQI client configuration file ##
## Function  : Define the configuration of a client ##
## ##
##*****##
## Notes    : ##
## 1) This file defines the configuration of a client ##
## ##
##*****##
SSL:
  OutboundSNI=HOSTNAME
```

Not: Bu sayfadaki herhangi bir değeri değiştirmeyin. For example the string HOSTNAME should be left as is.

Daha fazla ayrıntı için [İstemci yapılandırma dosyasının SSL stanzası](#) başlıklı konuya bakın.

IBM MQ 9.2.1' dan önceki bir IBM MQ client ya da araç takımı kullanıyorsanız, önceki yapılandırma dosyası yerine bir OCP rotası yaratmanız gerekir. [Not 1: Rotanın yaratılması](#) başlıklı konu başlıklı konuyu izleyin.

Kuyruk yöneticisini konuşlandır

Önemli: Bu örnekte, kuyruk yöneticisinde yetkilendirmeyi devre dışı bırakmak için *MQSNOAUT* değişkenini kullanırsınız; bu değişken, bir istemciyi TLS ' yi kullanarak bağlamak için gereken adımlara odaklanabilmemizi sağlar. This is not recommended in a production deployment of IBM MQ, because it causes any applications connecting to have full administrative powers, with no mechanism to lower the permissions for individual applications.

Aşağıdaki YAML ' yi kopyalayın ve güncelleştirin.

- Doğru lisansın belirtildiğinden emin olun. Bkz. mq.ibm.com/v1beta1 için lisans başvurusu.

- Accept the license by changing `false` to `true`.
- IBM Cloud File Storage kullanıyorsanız, bkz. [Not 3: IBM Cloud File Storage' un kullanılması](#)

Kuyruk yöneticisi özel kaynağı YAML:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: miexample
spec:
  license:
    accept: false
    license: L-RJON-C7QG3S
    use: NonProduction
  queueManager:
    name: MIEXAMPLE
    availability:
      type: MultiInstance
    mqsc:
      - configMap:
          name: example-mi-configmap
          items:
            - tls.mqsc
  template:
    pod:
      containers:
        - env:
            - name: MQSNOAUT
              value: 'yes'
          name: qmgr
  version: 9.2.5.0-r3
  web:
    enabled: true
  pki:
    keys:
      - name: example
      secret:
        secretName: example-mi-secret
        items:
          - tls.key
          - tls.crt
```

Daha önce oluşturduğunuz ad alanında olduğundan emin olmak için, güncellenmiş YAML 'yi OCP UI' de, komut satırını kullanarak ya da IBM Cloud Pak for Integration Platform Navigatorkomutunu kullanarak konuşlandırın.

Geçerlilik Denetimi

Kısa bir gecikmeden sonra, birden çok yönetim ortamı kuyruk yöneticisi yapılandırılmalı ve kullanılabilir olmalıdır. Bu bölümde, kuyruk yöneticisinin beklediği gibi davrandığını doğrularız.

Kuyruk yöneticisinin çalışır durumda olduğunu doğrulayın

Kuyruk yöneticisi şimdi konuşlandırılıyor. Confirm it is in Running state before proceeding. Örneğin:

```
oc get qmgr miexample
```

Kuyruk yöneticisiyle bağlantıyı test edin

Kuyruk yöneticisinin tek yönlü TLS iletişimi için yapılandırılıp yapılandırıldığı doğrulamak için, **amqspu**tc ve **amqsge**tc örnek uygulamalarını kullanın:

Kuyruk yöneticisi anasistem adını bul

miexample-ibm-mq-qmrotasına ilişkin kuyruk yöneticisi anasistem adını bulmak için aşağıdaki komutu çalıştırın. Anasistem adı, HOST alanında döndürülür.

```
oc get routes miexample-ibm-mq-qm
```


Kuyruk yöneticisi ayrıntılarını belirtin

Create a file CCDT.JSON that specifies the queue manager details. Anasistem değerini, önceki adımın döndürdüğü anasistem adı ile değiştirin.

```
{
  "channel":
  [
    {
      "name": "MIQMCHL",
      "clientConnection":
      {
        "connection":
        [
          {
            "host": "<host from previous step>",
            "port": 443
          }
        ],
        "queueManager": "MIEXAMPLE"
      },
      "transmissionSecurity":
      {
        "cipherSpecification": "ECDHE_RSA_AES_128_CBC_SHA256"
      },
      "type": "clientConnection"
    }
  ]
}
```

Ortam değişkenlerini dışa aktar

İşletim sisteminiz için uygun şekilde aşağıdaki ortam değişkenlerini dışa aktarın. Bu değişkenler **amqsputc** ve **amqsgetc** tarafından okunacaktır.

Sisteminizdeki kütüklerin yolunu güncelleyin:

```
export MQCCDTURL='<full_path_to_file>/CCDT.JSON'
export MQSSLKEYR='<full_path_to_file>/clientkey'
```

İletileri kuyruğa koy

Aşağıdaki komutu çalıştırın:

```
amqsputc EXAMPLE.QUEUE MIEXAMPLE
```

Kuyruk yöneticisiyle bağlantı başarılı olursa, aşağıdaki yanıt çıkışa gelir:

```
target queue is EXAMPLE.QUEUE
```

Kuyruğa birkaç ileti koyun, bir metin girerek her seferinde **Enter** tuşuna basın.

İşlemi tamamlamak için iki kez **Enter** tuşuna basın.

İletileri kuyruktan al

Aşağıdaki komutu çalıştırın:

```
amqsgetc EXAMPLE.QUEUE MIEXAMPLE
```

Önceki adımda eklediğiniz iletiler tüketildi ve çıkışa yazılıyor.

Birkaç saniye sonra, komuta kapanır.

Etkin kapsülü başarısız olmasına zorlar

Çok eşgörünümlü kuyruk yöneticisinin otomatik olarak kurtarılması işlemini doğrulamak için, bir pod hatası benzetimi gerçekleştirin:

Etkin ve beklemedeki bölmeleri görüntüle

Aşağıdaki komutu çalıştırın:

```
oc get pods
```

READY alanında etkin pod değeri 1/1 değerini döndürür, ancak beklemedeki bölme 0/1 değerini döndürür.

Etkin kapsülü sil

Etkin bölmenin tam adını belirterek aşağıdaki komutu çalıştırın:

```
oc delete pod miexample-ibm-mq-<value>
```

Pod durumunu yeniden görüntüle

Aşağıdaki komutu çalıştırın:

```
oc get pods
```

Bekleme bölmesi günlükünü görüntüle

Bekleme bölmesinin tam adını belirterek aşağıdaki komutu çalıştırın:

```
oc logs miexample-ibm-mq-<value>
```

Aşağıdaki iletiyi görmeniz gerekir:

```
IBM MQ queue manager 'MIEXAMPLE' becoming the active instance.
```

İletileri yeniden al ve yeniden al

Beklemedeki bölme etkin bölme olduktan sonra (READY alan değeri 1/1olduktan sonra), daha önce açıklandığı gibi, kuyruk yöneticisine ileti koymak için aşağıdaki komutları yeniden kullanın ve ardından kuyruk yöneticisinden iletileri alın:

```
amqsputc EXAMPLE.QUEUE MIEXAMPLE
```

```
amqsgetc EXAMPLE.QUEUE MIEXAMPLE
```

Tebrikler, çok eşgörünümlü bir kuyruk yöneticisini başarıyla yerleştirdiniz ve bir pod hatasından otomatik olarak kurtarılabildiğinizi gösterdiniz.

Ek bilgi

Not 1: Yol yaratılması

IBM MQ 9.2.1' dan önceki bir IBM MQ client ya da araç takımı kullanıyorsanız, bir OCP rotası yaratmanız gerekir.

Rotayı oluşturmak için, daha önce oluşturduğunuz ad alanında olduğunuzu doğrulayın (bkz. [“Başlamadan önce” sayfa 102](#)), OCP UI 'ye aşağıdaki YAML' yi girin ya da komut satırını kullanın:

```
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: example-mi-route
spec:
  host: miqmchl.ch1.mq.ibm.com
  to:
    kind: Service
    name: miexample-ibm-mq
  port:
    targetPort: 1414
  tls:
    termination: passthrough
```

Note that the Red Hat OpenShift Container Platform Router uses SNI for routing requests to the IBM MQ queue manager. [MQSC komutlarını içeren yapılandırma işlemi](#) da belirtilen kanal adını değiştirirseniz, anasistem alanını burada ve Kuyruk yöneticisi ayrıntılarını belirtenCCDT . JSON dosyasıiçinde de değiştirmeniz gerekir. Daha fazla bilgi için bkz [“Red Hat OpenShift kümesi dışından bir kuyruk yöneticisine bağlanmak için bir Rotanın yapılandırılması” sayfa 107](#).

Not 2: Varsayılan olmayan bir depolama sınıfı kullanılması

Bu örnek, OCP 'de varsayılan bir depolama sınıfının yapılandırılmasını bekler; bu nedenle, [queue manager custom resource YAML](#) ' da herhangi bir depolama bilgisi gerekmez. Varsayılan olarak

yapılandırılmış bir depolama sınıfınız yoksa ya da farklı bir depolama sınıfı kullanmak istiyorsanız, `spec.queueManager.storagealtina defaultClass: <storage_class_name>` ekleyin.

Depolama sınıfı adı, OCP sisteminizde var olan bir depolama sınıfının adıyla tam olarak eşleşmelidir. Yani, bu, `oc get storageclass` komutunun döndürdüğü adla eşleşmelidir. Bu, `ReadWriteMany`'i de desteklemelidir. Daha fazla bilgi için bkz “IBM MQ Operatorile ilgili depolama konuları” sayfa 11.

Not 3: IBM Cloud File Storage komutunu kullanma

Bazı durumlarda (örneğin, IBM Cloud File Storage kullanırken), kuyruk yöneticisi özel kaynağı YAML alanında **securityGroups** alanını da belirtmeniz gerekir. Örneğin, aşağıdaki alt alanı doğrudan `spec:altina` ekleyerek

```
securityContext:
  supplementalGroups: [99]
```

Daha fazla bilgi için bkz “IBM MQ Operatorile ilgili depolama konuları” sayfa 11.

OpenShift CP4I CD Red Hat OpenShift kümesi dışından bir kuyruk yöneticisine bağlanmak için bir Rotanın yapılandırılması

Bir uygulamayı Red Hat OpenShift kümesi dışından IBM MQ kuyruk yöneticisine bağlamak için bir Red Hat OpenShift Rotası gerekir. SNI yalnızca TLS 1.2 ya da daha yüksek bir iletişim kuralı kullanıldığında TLS iletişim kuralında kullanılabilir olduğundan, IBM MQ kuyruk yöneticisi ve istemci uygulamanızda TLS 'yi etkinleştirmeniz gerekir. Red Hat OpenShift Container Platform Router , istekleri IBM MQ kuyruk yöneticisine yönlendirmek için SNI kullanır.

Bu görev hakkında



Uyarı: Bu belge, IBM MQ istemcilerinin 9.2.1 Continuous Delivery ve sonraki sürümleri için geçerlidir. İstemciniz 9.2.0 Long Term Support sürümünü ya da önceki bir sürümü kullanıyorsa, Red Hat OpenShift kümesinde konuşlandırılmış bir kuyruk yöneticisine bağlanma IBM MQ 9.1 belge sayfasına bakın.

V 9.2.1 Gerekli Red Hat OpenShift Rotası yapılandırması, istemci uygulamanızın Sunucu Adı Göstergesi (SNI) davranışına bağlıdır. IBM MQ , yapılandırmaya ve istemci tipine bağlı olarak iki farklı SNI üstbilgi ayarını destekler. Bir SNI Üstbilgisi, istemcinin hedefinin anasistem adına ya da diğer bir seçenek olarak IBM MQ kanal adına ayarlanır. IBM MQ ' in bir kanal adını bir anasistem adıyla nasıl eşlediği hakkında bilgi için bkz. Nasıl IBM MQ birden çok sertifika yeteneği sağlar.

V 9.2.1 Bir SNI üstbilgisinin IBM MQ kanal adına mı, yoksa bir anasistem adına mı ayarlandığı **OutboundSNI** özniteliği kullanılarak denetlenir. Olası değerler: `OutboundSNI=CHANNEL` (varsayılan değer) ya da `OutboundSNI=HOSTNAME`. Daha fazla bilgi için bkz. İstemci yapısı kütüğünün SSL kısmı. CHANNEL ve HOSTNAME değerlerinin kullandığınız tam değerler olduğunu ve bunların gerçek bir kanal adı ya da anasistem adıyla değiştirdiğiniz değişken adlar olmadığını unutmayın.

V 9.2.1

Farklı OutboundSNI ayarlarına sahip istemci davranışları

OutboundSNI , HOSTNAME olarak ayarlanırsa, bağlantı adında bir anasistem adı sağlandığı sürece, aşağıdaki istemciler bir anasistem adı SNI değeri ayarlar:

- C İstemcileri
- Yönetilmeyen kipte .NET İstemcileri
- Java/JMS İstemcileri

OutboundSNI değeri HOSTNAME olarak ayarlanırsa ve bağlantı adında bir IP adresi kullanılırsa, aşağıdaki istemciler boş bir SNI üstbilgisi gönderir:

- C İstemcileri

- Yönetilmeyen kipte .NET İstemcileri
- Java/JMS İstemciler (anasistem adının ters DNS araması yapamayan)

OutboundSNI CHANNELolarak ayarlanırsa ya da ayarlanmazsa, bunun yerine bir IBM MQ kanal adı kullanılır ve ister anasistem adı, ister IP adresi bağlantı adı kullanılsın, her zaman gönderilir.

Aşağıdaki istemci tipleri, bir SNI üstbilgisinin IBM MQ kanal adına ayarlanmasını desteklemez ve bu nedenle, **OutboundSNI** ayarına bakılmaksızın SNI üstbilgisini her zaman bir anasistem adı olarak ayarlamayı dener:

- AMQP istemcileri
- XR İstemcileri
- Yönetilen kipteki .NET istemcileri (Long Term Support için IBM MQ 9.2.0 Fix Pack 4 ' den önce ve IBM MQ 9.2.3 for Continuous Delivery' dan önce.)

▶ V 9.2.3 ▶ V 9.2.0.4

OutboundSNI özelliği ANASISTEM ADıolarak ayarlanırsa, Long Term Support için IBM MQ 9.2.0 Fix Pack 4 ve IBM MQ 9.2.3 for Continuous Deliveryolanağından, IBM MQ yönetilen .NET istemcisi SUNUCU ADı değerini ilgili anasistem adına ayarlayacak şekilde güncellenmiştir. Bu ayar, IBM MQ yönetilen .NET bir istemcinin Red Hat OpenShift rotalarını kullanarak bir kuyruk yöneticisine bağlanmasına olanak tanır. IBM MQ 9.2.0 Fix Pack 4içinde **OutboundSNI** özelliğinin yalnızca mqclient.ini dosyasından eklendiğini ve desteklendiğini unutmayın; özelliği .NET uygulamasından ayarlayamazsınız.

▶ V 9.2.5

Bir istemci uygulaması IBM MQ Internet Pass-Thru (MQIPT) aracılığıyla bir Red Hat OpenShift kümesinde konuşlandırılan bir kuyruk yöneticisine bağlanıyorsa, MQIPT , rota tanımlamasında SSLClientOutboundSNI özelliğini kullanarak SNI ' yi anasistem adına ayarlayacak şekilde yapılandırılabilir.

OutboundSNI, birden çok sertifika ve Red Hat OpenShift rotaları

IBM MQ , birden çok sertifika işlevselliği sağlamak için SNI üstbilgisini kullanır. Bir uygulama CERTLABL alanı aracılığıyla farklı bir sertifika kullanacak şekilde yapılandırılmış bir IBM MQ kanalına bağlanıyorsa, uygulamanın CHANNEL 'ın **OutboundSNI** ayarına bağlanması gerekir.

Red Hat OpenShift Rota yapılandırmanız HOSTNAME SNI gerektiriyorsa, IBM MQ ' un birden çok sertifika işlevini kullanamazsınız ve herhangi bir IBM MQ kanal nesnesinde CERTLABL ayarı ayarlayamıyorsunuz.

OutboundSNI ayarı dışındaki bir uygulama, yapılandırılmış bir sertifika etiketiyle bir kanala bağlanırsa, uygulama bir MQRC_SSL_INITIALIZATION_ERROR ile reddedilir ve kuyruk yöneticisi hata günlüklerine AMQ9673 iletisi yazdırılır.

IBM MQ ' ın birden çok sertifika işlevini nasıl sağladığı hakkında daha fazla bilgi için bkz. [Nasıl IBM MQ birden çok sertifika yeteneği sağlar](#) .

Örnek

SNI ' yi MQ kanalına ayarlanan istemci uygulamaları, bağlanmak istediğiniz her kanal için yeni bir Red Hat OpenShift Rotası oluşturulmasını gerektirir. Doğru kuyruk yöneticisine yönelmeye izin vermek için Red Hat OpenShift Container Platform kümenizden benzersiz kanal adları da kullanmanız gerekir.

IBM MQ kanal adlarını SNI üstbilgileriyle eşlediği için MQ kanal adlarının küçük harfle bitmemesi önemlidir.

Yeni Red Hat OpenShift Rotalarınızın her biri için gerekli anasistem adını belirlemek üzere her bir kanal adını bir SNI adresiyle eşlemeniz gerekir. Daha fazla bilgi için bkz. [Nasıl IBM MQ birden çok sertifika yeteneği sağlar](#) .

Daha sonra, kümenizde aşağıdakileri uygulayarak yam1 her kanal için yeni bir Red Hat OpenShift Rotası oluşturmanız gerekir:

```
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: <provide a unique name for the Route>
  namespace: <the namespace of your MQ deployment>
spec:
  host: <SNI address mapping for the channel>
  to:
    kind: Service
    name: <the name of the Kubernetes Service for your MQ deployment (for example "Queue Manager Name)-ibm-mq">
  port:
    targetPort: 1414
  tls:
    termination: passthrough
```

İstemci uygulaması bağlantı ayrıntılarının yapılandırılması

İstemci bağlantınız için kullanılacak anasistem adını saptamak için aşağıdaki komutu çalıştırın:

```
oc get route <Name of hostname based Route (for example "Queue Manager Name)-ibm-mq-qm">
-n <namespace of your MQ deployment> -o jsonpath="{.spec.host}"
```

İstemci bağlantınızın kapısı, Red Hat OpenShift Container Platform Yöneticisi tarafından kullanılan kapıya (normalde 443) ayarlanmalıdır.

İlgili görevler

“Red Hat OpenShift kümesinde konuşlandırılmış IBM MQ Console ile bağlantı kurulması” sayfa 113
Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

CP4I IBM Cloud Pak for Integration Operations Dashboard ile bütünleştirme

The ability to trace transactions through IBM Cloud Pak for Integration is provided by the Operations Dashboard.

Bu görev hakkında

Operations Dashboard ile tümleştirmenin etkinleştirilmesi, kuyruk yöneticinize bir MQ API çıkışı kurar. API çıkışı, kuyruk yöneticisiyle akan iletiler hakkında izleme verilerini Operations Dashboard veri deposuna gönderecektir.

Yalnızca MQ istemci bağlamaları kullanılarak gönderilen iletilerin izlendiğini unutmayın.

Ayrıca, izleme etkinleştirildiğinde, IBM MQ Operator 'un 1.5 öncesi sürümleri için, kuyruk yöneticisinin yanında konuşlandırılan izleme aracı ve toplayıcı görüntülerinin her zaman en son sürümler olduğunu da unutmayın; bu görüntüler, IBM Cloud Pak for Integration' un en son sürümünü kullanmayacaksa bir uyumsuzluk getirebilirler.

Yordam

1. İzlemenin etkinleştirildiği bir kuyruk yöneticisini konuşlandır

Varsayılan olarak izleme özelliği devre dışı bırakılır.

IBM Cloud Pak for Integration Platform Navigator kullanarak konuşlandırıyorsanız, konuşlandırma sırasında izlemeyi etkinleştirebilir; konuşlandırma sırasında **İzlemeyi Etkinleştir** ayarını **Açık** olarak ayarlayarak ve **İzleme Ad Alanı** ayarını, Operations Dashboard olanağının kurulu olduğu ad alanına yerleştirebilirsiniz. Kuyruk yöneticisini konuşlandırma hakkında daha fazla bilgi için bkz. [“Deploying a queue manager using the IBM Cloud Pak for Integration Platform Navigator” sayfa 81](#)

If you are deploying using the [Red Hat OpenShift CLI](#) or [Red Hat OpenShift Web konsolu](#), then you can enable tracing with the following YAML snippet:

```
spec:
  tracing:
    enabled: true
    namespace: <Operations_Dashboard_Namespace
```

Önemli: Kuyruk yöneticisi, MQ Operations Dashboard 'a kaydedilinceye kadar başlatılmaz (sonraki adıma bakın).

Bu özellik etkinleştirildiğinde, kuyruk yöneticisi taşıyıcısına ek olarak iki sidecar taşıyıcısı ("Agent" ve "Collector") çalıştırılır. Bu sidecar taşıyıcılara ilişkin resimler, ana MQ görüntüleriyle aynı kayıta kullanılabilir ve aynı çekme ilkesini kullanacak ve güvenlik dizgisini çekecektir. CPU ve bellek sınırlarını yapılandırmak için kullanılacak ek ayarlar da vardır.

2. Bu, Operations Dashboard bütünleşmesi içeren bir kuyruk yöneticisinin ilk kez bu ad alanında konuşlandırıldıysa, Operations Dashboard ile [Kaydol](#) ' e gereksinim duyarsınız.

Kayıt işlemi, kuyruk yöneticisi Pod 'unun başarıyla başlaması gereken bir Gizli nesne yaratır.

CP4I > CD IBM Cloud Pak for Integration 2021.4 içinde Operations Dashboard bütünleşmesi ile IBM MQ 9.2.2 ya da 9.2.3 ürününün konuşlandırılması ya da büyütülmesi

Her IBM MQ sürümü, bir kuyruk yöneticisiyle birlikte konuşlandırılan Operations Dashboard aracısı ve toplayıcı bileşenlerinin belirli bir sürümüyle ilişkilendirilir. IBM Cloud Pak for Integration 2021.4.1 , eski aracı ve toplayıcı bileşenlerinin Operations Dashboard ile çalışmamasına neden olan bir değişiklik sunar. Bunu düzeltmek için, IBM MQ 9.2.2 ya da 9.2.3 seçeneğini kullandığınızda, kullandığınız Operations Dashboard aracısı ve toplayıcı görüntülerinin sürümünü geçersiz kılmanız gerekir.

Yeni bir IBM MQ 9.2.2 ya da 9.2.3 kuyruk yöneticisinin konuşlandırılması

IBM Cloud Pak for Integration 2021.4.1 'i IBM MQ 9.2.2 ya da 9.2.3 ile kullanırken, Operations Dashboard aracısı ve toplayıcı görüntülerini QueueManager YAML' nizdeki 2.4 sürümlerine geçersiz kılmanız gerekir. Örneğin:

```
spec:
  tracing:
    agent:
      image: cp.icr.io/cp/icp4i/od/icp4i-od-agent@sha256:27a211f0f78eff765d1f9520e0f9841f902600bb556827477b206e209cb44d20
    collector:
      image: cp.icr.io/cp/icp4i/od/icp4i-od-collector@sha256:dc70b1341b23dc72642ce68809811f9db0e8a0c46bda2508e8eb3d4035e04f4b
```

Bunu yapmazsanız, QueueManager Pod 'unuz Pending durumunda sıkışık kalmış olur. IBM MQ 9.2.4' a yükselttiğinizde, bu geçersiz kılmaları kaldırabilirsiniz.

IBM Cloud Pak for Integration 2021.4.1 ürününe büyütme

Not: IBM MQ 9.2.2 ya da 9.2.3 kuyruk yöneticinizi alıyorsanız, adım 3 'i tamamlamayın.

1. Aracı ve toplayıcı görüntülerini geçersiz kılmak için QueueManager ' u (önceden açıklandığı gibi) güncelleyin.
2. Upgrade your IBM Cloud Pak for Integration operators, including the Operations Dashboard and the IBM MQ Operator, as described in [“IBM MQ Operator ve kuyruk yöneticilerinin yükseltilmesi” sayfa 70.](#)
3. (optional) To upgrade to IBM MQ 9.2.4 or later, update your QueueManager to use `.spec.version` for your version of IBM MQ, then remove the override of the agent and collector images.

Red Hat OpenShift CLI kullanılarak özel MQSC ve INI dosyaları içeren bir resim oluşturma

Use an Red Hat OpenShift Container Platform Pipeline to create a new IBM MQ container image, with MQSC and INI files you want to be applied to queue managers using this image. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır

Başlamadan önce

Red Hat OpenShift Container Platform komut satırı arabirimi' yi kurmanız gerekir.

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

Red Hat OpenShift projenizdeki IBM Yetkili Kaydı için bir Red Hat OpenShift Sıranız yoksa, [“Preparing your Red Hat OpenShift project for IBM MQ” sayfa 78](#)adımlarına ilişkin adımları izleyin.

Yordam

1. Yarat ImageStream

Bir görüntü akışı ve ilişkili etiketleri, Red Hat OpenShift Container Platform'ından taşıyıcı görüntülerine gönderme yapmak için bir soyutlama sağlar. Görüntü akışı ve etiketleri, hangi resimlerin kullanılabilir olduğunu görmenizi ve havuzdaki resim değişse de gereksinim duyduğunuz belirli görüntüyü kullanmanızı sağlar.

```
oc create imagestream mymq
```

2. Yeni görüntünüz için bir BuildConfig yaratın

BuildConfig, yeni görüntünüz için oluşturmaları izin verir; bu, IBM resmi görüntüleri temel alınarak oluşturulur, ancak kapsayıcı başlatma sırasında çalıştırılmasını istediğiniz herhangi bir MQSC ya da INI dosyasını ekler.

a) BuildConfig kaynağını tanımlayan bir YAML dosyası yarat

Örneğin, aşağıdaki içerikle "mq-build-config.yaml" adlı bir dosya yaratın:

```
apiVersion: build.openshift.io/v1
kind: BuildConfig
metadata:
  name: mymq
spec:
  source:
    dockerfile: |-
      FROM cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.5.0-r3
      RUN printf "DEFINE QLOCAL(foo) REPLACE\n" > /etc/mqm/my.mqsc \
        && printf "Channels:\n\tMQIBindType=FASTPATH\n" > /etc/mqm/my.ini
      LABEL summary "My custom MQ image"
  strategy:
    type: Docker
    dockerStrategy:
      from:
        kind: "DockerImage"
        name: "cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.5.0-r3"
      pullSecret:
        name: ibm-entitlement-key
  output:
    to:
      kind: ImageStreamTag
      name: 'mymq:latest-amd64'
```

Kullanmak istediğiniz sürüm ve düzeltmeye ilişkin doğru temel görüntüyü göstermek için, IBM MQ tabanının belirtildiği iki yer yerine geçmeniz gerekir (ayrıntılar için [“IBM MQ Operator için yayın geçmişi” sayfa 19](#) ' e bakın). Düzeltmeler uygulandığında, imajını yeniden oluşturmak için bu adımları yinelemeniz gerekir.

Bu örnek, IBM resmi görüntüsünü temel alan yeni bir görüntü yaratır ve "my.mqsc" ve "my.ini" adlı dosyaları /etc/mqm dizinine ekler. Bu dizinde bulunan herhangi bir MQSC ya da INI dosyası, başlatma sırasında taşıyıcı tarafından uygulanacaktır. INI dosyaları, **crtmqm -ii** seçeneği

kullanılarak uygulanır ve var olan INI dosyalarıyla birleştirilir. MQSC dosyaları alfabetik sırayla uygulanır.

Kuyruk yöneticisi başlatılırken *her zaman* çalıştırılacağı için, MQSC komutlarınızın yinelenebilir olması önemlidir. This typically means adding the REPLACE parameter on any DEFINE commands, and adding the IGNSSTATE (YES) parameter to any START or STOP commands.

b) Apply the BuildConfig to the server.

```
oc apply -f mq-build-config.yaml
```

3. Görüntünüzü oluşturmak için bir oluşturma çalıştırın

a) Oluşturmayı başlat

```
oc start-build mymq
```

Çıktıyı aşağıdakine benzer bir şekilde görmeniz gerekir:

```
build.build.openshift.io/mymq-1 started
```

b) Oluşturmanın durumunu denetleyin

Örneğin, önceki adımda döndürülen oluşturma tanıtıcısını kullanarak aşağıdaki komutu çalıştırabilirsiniz:

```
oc describe build mymq-1
```

4. Yeni görüntünüzü kullanarak bir kuyruk yöneticisini konuşlandırın

[“Kuyruk yöneticisinin bir Red Hat OpenShift Container Platform kümesine konuşlandırılması”](#) sayfa 80'ta açıklanan adımları izleyin, yeni özel görüntünüzü YAML' ye ekleyin.

YAML 'in aşağıdaki parçasını normal QueueManager YAML' nize ekleyebilirsiniz; burada *ad alanı* , kullandığınız Red Hat OpenShift project/ad alanı ve *resim* , daha önce oluşturduğunuz resmin adıdır (örneğin, "mymq:latest-amd64"):

```
spec:
  queueManager:
    image: image-registry.openshift-image-registry.svc:5000/my-namespace/my-image
```

İlgili görevler

[“Kuyruk yöneticisinin bir Red Hat OpenShift Container Platform kümesine konuşlandırılması”](#) sayfa 80 Bir kuyruk yöneticisini bir Red Hat OpenShift Container Platform kümesine konuşlandırmak için QueueManager özel kaynağını kullanın.

Kuyruk yöneticisi kaynaklarına özel ek açıklamalar ve etiketler eklenmesi

QueueManager meta verisine özel ek açıklamalar ve etiketler ekleyebilirsiniz.

Bu görev hakkında

Özel ek açıklamalar ve etiketler, PVC ' ler dışındaki tüm kaynaklara eklenir. Özel bir ek açıklama ya da etiket, var olan bir anahtarla eşleşirse, IBM MQ Operator değeri ile ayarlanan değer kullanılır.

Yordam

- Özel ek açıklamalar eklemenizi sağlar.

To add custom annotations to queue manager resources, including the pod, add the annotations under metadata. Örneğin:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: quickstart-cp4i
```



```
annotations:
  annotationKey: "value"
```

- Özel etiketler ekleyin.

Pod da içinde olmak üzere kuyruk yöneticisi kaynaklarına özel etiketler eklemek için etiketleri metadataaaltına ekleyin. Örneğin:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: quickstart-cp4i
  labels:
    labelKey: "value"
```

OpenShift CP4I Runtime webhook denetimlerinin geçersiz kılınması

Çalıştırma zamanı webhook denetimleri, depolama sınıflarının kuyruk yöneticiniz için uygun olup olmadığını denetler. Performansı artırmak için ya da ortamınız için geçerli olmadıkları için bunları devre dışı bırakırsınız.

Bu görev hakkında

Çalıştırma zamanı webhook denetimleri kuyruk yöneticisi yapılanışında yapılır. Depolama sınıflarının seçilen kuyruk yöneticisi tipinize uygun olup olmadığını denetler.

Bu denetimleri, kuyruk yöneticisi oluşturma için alınan süreyi azaltmak için ya da denetlerin özel ortamınız için geçerli olmadığı için geçersiz kılmalarını seçebilirsiniz.

Not: Çalıştırma zamanı webhook denetimlerini devre dışı bıraktıktan sonra, herhangi bir depolama sınıfı değerine izin verilir. Bu, bozuk bir kuyruk yöneticisinde sonuçlanabilir.

Çalıştırma zamanı denetimleri desteği IBM MQ Operator 1.2 içinde kullanıma sunulmuştur.

Yordam

- Runtime webhook denetimlerini geçersiz kıl

Aşağıdaki ek açıklamayı metadataaaltına ekleyin. Örneğin:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: quickstart-cp4i
  annotations:
    "com.ibm.cp4i/disable-webhook-runtime-checks" : "true"
```

OpenShift CP4I Operating IBM MQ using the IBM MQ Operator

Bu görev hakkında

Yordam

- [“Preparing your Red Hat OpenShift project for IBM MQ” sayfa 78.](#)
- [“Kuyruk yöneticisinin bir Red Hat OpenShift Container Platform kümesine konuşlandırılması” sayfa 80.](#)

OpenShift CP4I Red Hat OpenShift kümesinde konuşlandırılmış IBM MQ Console ile bağlantı kurulması

Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

Bu görev hakkında

IBM MQ Console URL 'si, Red Hat OpenShift web konsolunun QueueManager ayrıntılar sayfasında ya da IBM Cloud Pak for Integration Platform Navigator' de bulunabilir. Diğer bir seçenek olarak, aşağıdaki komutu çalıştırarak Red Hat OpenShift CLI 'sından bulunabilir:

```
oc get queuemanager <QueueManager Name> -n <namespace of your MQ deployment> --output jsonpath='{.status.adminUiUrl}'
```

IBM Cloud Pak for Integration lisansı kullanıyorsanız, IBM MQ Web konsolu, IBM Cloud Pak Identity and Access Manager (IAM) olanağını kullanacak şekilde yapılandırılır. IAM bileşeni, küme denetimciniz tarafından önceden ayarlanmış olabilir. Ancak, IAM ilk kez Red Hat OpenShift kümeniz üzerinde kullanıldıysa, ilk yönetici parolasını almanız gerekir. Ek bilgi için [İlk yönetici parolasının alınması](#) başlıklı konuya bakın.

Bir IBM MQ lisansı kullanıyorsanız, MQ web konsolu önceden yapılandırılmamışsa ve bunu kendiniz yapılandırmanız gerekir. Daha fazla bilgi için [Kullanıcıları ve rolleri yapılandırma](#) başlıklı konuya bakın.

İlgili görevler

“Red Hat OpenShift kümesi dışından bir kuyruk yöneticisine bağlanmak için bir Rotanın yapılandırılması” [sayfa 107](#)

Bir uygulamayı Red Hat OpenShift kümesi dışından IBM MQ kuyruk yöneticisine bağlamak için bir Red Hat OpenShift Rotası gerekir. SNI yalnızca TLS 1.2 ya da daha yüksek bir iletişim kuralı kullanıldığında TLS iletişim kuralında kullanılabilir olduğundan, IBM MQ kuyruk yöneticisi ve istemci uygulamanızda TLS 'yi etkinleştirmeniz gerekir. Red Hat OpenShift Container Platform Router , istekleri IBM MQ kuyruk yöneticisine yönlendirmek için SNI kullanır.

IBM Cloud Pak IAM olanağını kullanarak IBM MQ Console için izin verme

Permissions for the IBM MQ Console are managed through the IBM Cloud Pak Administration Hub, and not the IBM Cloud Pak for Integration Platform Navigator. IBM MQ does not use the "Automation" permissions provided by the IBM Cloud Pak for Integration, but instead uses the basic permissions enabled by the IBM Cloud Pak Identity and Access Manager (IAM).

Yordam

1. IBM Cloud Pak Administration Console olanağını açın.

IBM Cloud Pak for Integration Platform UI' den, araç çubuğunun sağ üst köşesindeki Cloud Pak switcher (9-nokta simgesi) simgesini tıklatın, daha sonra **IBM Cloud Pak Administration** (IBM Cloud-Pak Administration) panosunu tıklatın.

2. Üst sol köşedeki gezinme menüsünde **Kimlik ve erişim** seçeneğini belirleyin, ardından **Ekipler ve hizmetler tanıtıcıları** seçeneğini belirleyin.

3. Bir ekip oluşturun, ardından kullanıcıları bu ekibe ekleyin.

- a) **Ekip oluşturma** seçeneğini belirleyin.

- b) Bir ekip adı girin ve yönetmek istediğiniz kullanıcılar için güvenlik etki alanını seçin.

- c) Kullanıcılar için arama yapın.

Bu kullanıcıların önceden kimlik sağlayıcınızda var olması gerekir.

- d) Her bir kullanıcıyı bulduğunuzda, onlara bir rol verin. This must be "Administrator" or "Cluster Administrator", to administer IBM MQ using the IBM MQ Console.

4. Her kullanıcıyı bir ad alanına ekleyin.

- a) Düzenlemek için ekibi seçin.

- b) **Kaynaklar > Kaynakları Yönet** seçeneklerini belirleyin.

- c) Bu ekibin yönetmesini istediğiniz ad alanlarını seçin. Bunlar, kuyruk yöneticisi içeren herhangi bir ad alanı olabilir.

IBM MQ Operator tarafından yönetilen kuyruk yöneticileri, Prometheus ile uyumlu metrikler üretebilir.

Bu metrikleri, Red Hat OpenShift Container Platform (OCP) izleme yığını kullanarak görüntüleyebilirsiniz. OCP' ta **Metrikler** sekmesini açın ve daha sonra, **Gözetim > Metrikler** seçeneklerini tıklayın. Kuyruk yöneticisi metrikleri varsayılan olarak etkindir, ancak **.spec.metrics.enabled** ayarı false değerine ayarlanarak devre dışı bırakılabilir.

Prometheus, metrikler için bir zaman serisi veritabanı ve kural değerlendirme motorudur. IBM MQ kapsayıcıları, Prometheus tarafından sorgulanabilen bir metrik uç noktasını gösterir. Ölçümler, izleme ve etkinlik izleme için MQ sistem konularından oluşturulur.

Red Hat OpenShift Container Platform, bir Prometheus sunucusu kullanan önceden yapılandırılmış, önceden kurulmuş ve kendi kendini güncelleyen bir izleme yığını içerir. Kullanıcı tanımlı projeleri izlemek için Red Hat OpenShift Container Platform izleme yığınının yapılandırılması gerekir. Daha fazla bilgi için [Kullanıcı tanımlı projeler için izleme özelliğinin etkinleştirilmes](#) başlıklı konuya bakın. The IBM MQ Operator creates a ServiceMonitor when you create a QueueManager with metrics enabled, which the Prometheus Operator can then discover.

In older versions of IBM Cloud Pak for Integration, you could also use the [IBM Cloud Platform İzleme](#) service to provide a Prometheus server instead.

Kuyruk yöneticisi taşıyıcıları, Red Hat OpenShift Monitoring ile uyumlu metrikleri yayınlatabilir.

Metrik	Tip	Tanım
ibmmq_qmgr_commit_total	counter	Kesinleştirme sayısı
ibmmq_qmgr_cpu_load_fifteen_minute_average_percentage	gauge	CPU yükü-ortalama on beş dakika
ibmmq_qmgr_cpu_load_five_minute_average_percentage	gauge	CPU yükü-ortalama beş dakika
ibmmq_qmgr_cpu_load_one_minute_average_percentage	gauge	CPU yükü-bir dakika ortalama
ibmmq_qmgr_destructive_get_bytes_total	counter	Aralık toplam yıkıcı get-byte sayısı
ibmmq_qmgr_destructive_get_total	counter	Aralık toplam yıkıcı get-count
ibmmq_qmgr_durable_subscription_alter_total	counter	Kalıcı abonelik sayısını değiştir
ibmmq_qmgr_durable_subscription_create_total	counter	Sürekli abonelik sayısı yarat
ibmmq_qmgr_durable_subscription_delete_total	counter	Kalıcı abonelik sayısını sil

Metrik	Tip	Tanım
ibmmq_qmgr_durable_subscription_resume_total	counter	Sürekli abonelik sayısını sürdür
ibmmq_qmgr_errors_file_system_free_space_percentage	gauge	MQ hata dosyası sistemi-boş alan
ibmmq_qmgr_errors_file_system_in_use_bytes	gauge	MQ errors file system - bytes in use
ibmmq_qmgr_expired_message_total	counter	Süresi dolan ileti sayısı
ibmmq_qmgr_failed_browse_total	counter	Başarısız göz atma sayısı
ibmmq_qmgr_failed_mqcb_total	counter	Başarısız olan MQCB sayısı
ibmmq_qmgr_failed_mqclose_total	counter	MQCLOSE sayısı başarısız oldu
ibmmq_qmgr_failed_mqconn_mqconnx_total	counter	Başarısız olan MQCONN/MQCONNX sayısı
ibmmq_qmgr_failed_mqget_total	counter	MQGET-count başarısız oldu
ibmmq_qmgr_failed_mqinq_total	counter	Başarısız olan MQINQ sayısı
ibmmq_qmgr_failed_mqopen_total	counter	Başarısız MQOPEN sayısı
ibmmq_qmgr_failed_mqput1_total	counter	Başarısız MQPUT1 sayısı
ibmmq_qmgr_failed_mqput_total	counter	Başarısız MQPUT sayısı
ibmmq_qmgr_failed_mqset_total	counter	Başarısız MQSET sayısı
ibmmq_qmgr_failed_mqsubrq_total	counter	Başarısız olan MQSUBRQ sayısı
ibmmq_qmgr_failed_subscription_create_alter_resume_total	counter	Başarısız yaratma/değiştirme/sürdürme abonelik sayısı
ibmmq_qmgr_failed_subscription_delete_total	counter	Abonelik silme başarısız sayısı
ibmmq_qmgr_failed_topic_mqput_mqput1_total	counter	Başarısız olan MQPUT/MQPUT1 sayısı

Metrik	Tip	Tanım
ibmmq_qmgr_fdc_files	gauge	MQ FDC dosya sayısı
ibmmq_qmgr_log_file_system_in_use_bytes	gauge	Günlük kütüğü sistem byte'larını kullanımda
ibmmq_qmgr_log_file_system_max_bytes	gauge	Günlük dosyası sistemi-byte sayısı üst sınırı
ibmmq_qmgr_log_in_use_bytes	gauge	Günlük-bayt kullanımda
ibmmq_qmgr_log_logical_written_bytes_total	counter	Günlük-mantıksal baytlar yazıldı
ibmmq_qmgr_log_max_bytes	gauge	Günlük-byte sayısı üst sınırı
ibmmq_qmgr_log_physical_written_bytes_total	counter	Günlük-fiziksel baytlar yazıldı
ibmmq_qmgr_log_primary_space_in_use_percentage	gauge	Günlük-yürürlükteki birincil alan kullanımda
ibmmq_qmgr_log_workload_primary_space_utilization_percentage	gauge	Günlük iş yükü birincil alan kullanımı
ibmmq_qmgr_log_write_latency_seconds	gauge	Günlük yazma gecikme süresi
ibmmq_qmgr_log_write_size_bytes	gauge	Günlük yazma büyüklüğü
ibmmq_qmgr_mqcb_total	counter	MQCB sayısı
ibmmq_qmgr_mqclose_total	counter	MQCLOSE sayısı
ibmmq_qmgr_mqconn_mqconnx_total	counter	MQCONN/MQCONNX sayısı
ibmmq_qmgr_mqctl_total	counter	MQCTL sayısı
ibmmq_qmgr_mqdisc_total	counter	MQDISC sayısı
ibmmq_qmgr_mqinq_total	counter	MQINQ sayısı
ibmmq_qmgr_mqopen_total	counter	MQOPER sayısı
ibmmq_qmgr_mqput_mqput1_bytes_total	counter	Aralık toplam MQPUT/MQPUT1 byte sayısı

Metrik	Tip	Tanım
ibmmq_qmgr_mqput_mqput1_total	counter	Aralık toplamı MQPUT/MQPUT1 sayısı
ibmmq_qmgr_mqset_total	counter	MQSET sayısı
ibmmq_qmgr_mqstat_total	counter	MQSTAT sayısı
ibmmq_qmgr_mqsubrq_total	counter	MQSUBRQ sayısı
ibmmq_qmgr_non_durable_subscription_create_total	counter	Kalıcı olmayan abonelik sayısı yarat
ibmmq_qmgr_non_durable_subscription_delete_total	counter	Kalıcı olmayan abonelik sayısını sil
ibmmq_qmgr_non_persistent_message_browse_bytes_total	counter	Kalıcı olmayan ileti göz atma-bayt sayısı
ibmmq_qmgr_non_persistent_message_browse_total	counter	Kalıcı olmayan ileti göz atma sayısı
ibmmq_qmgr_non_persistent_message_destructive_get_total	counter	Kalıcı olmayan ileti yıkıcı get-count
ibmmq_qmgr_non_persistent_message_get_bytes_total	counter	Kalıcı olmayan iletiler alındı-bayt sayısı
ibmmq_qmgr_non_persistent_message_mqput1_total	counter	Kalıcı olmayan ileti MQPUT1 sayısı
ibmmq_qmgr_non_persistent_message_mqput_total	counter	Kalıcı olmayan ileti MQPUT sayısı
ibmmq_qmgr_non_persistent_message_put_bytes_total	counter	Kalıcı olmayan iletileri koy-byte sayısı
ibmmq_qmgr_non_persistent_topic_mqput_mqput1_total	counter	Kalıcı olmayan konu MQPUT/MQPUT1 sayısı
ibmmq_qmgr_persistent_message_browse_bytes_total	counter	Kalıcı ileti göz atma-bayt sayısı
ibmmq_qmgr_persistent_message_browse_total	counter	Kalıcı ileti göz atma sayısı

Metrik	Tip	Tanım
ibmmq_qmgr_persistent_message_destructive_get_total	counter	Kalıcı ileti yıkıcı get-count
ibmmq_qmgr_persistent_message_get_bytes_total	counter	Kalıcı iletiler-byte sayısı alındı
ibmmq_qmgr_persistent_message_mqput1_total	counter	Kalıcı ileti MQPUT1 sayısı
ibmmq_qmgr_persistent_message_mqput_total	counter	Kalıcı ileti MQPUT sayısı
ibmmq_qmgr_persistent_message_put_bytes_total	counter	Kalıcı iletileri koy-byte sayısı
ibmmq_qmgr_persistent_topic_mqput_mqput1_total	counter	Kalıcı-konu MQPUT/MQPUT1 sayısı
ibmmq_qmgr_published_to_subscribers_bytes_total	counter	Abonelere yayınlandı-bayt sayısı
ibmmq_qmgr_published_to_subscribers_message_total	counter	Abonelere yayınlandı-ileti sayısı
ibmmq_qmgr_purged_queue_total	counter	Temizlenen kuyruk sayısı
ibmmq_qmgr_queue_manager_file_system_free_space_percentage	gauge	Kuyruk Yöneticisi dosya sistemi-boş alan
ibmmq_qmgr_queue_manager_file_system_in_use_bytes	gauge	Kuyruk Yöneticisi dosya sistemi-kullanımdaki bayt sayısı
ibmmq_qmgr_ram_free_percentage	gauge	RAM serbest yüzdesi
ibmmq_qmgr_ram_usage_estimate_for_queue_manager_bytes	gauge	RAM toplam bayt sayısı-kuyruk yöneticisi için tahmin
ibmmq_qmgr_rollback_total	counter	Geriye işleme sayısı
ibmmq_qmgr_system_cpu_time_estimate_for_queue_manager_percentage	gauge	Sistem CPU süresi-kuyruk yöneticisi için yüzde tahmini
ibmmq_qmgr_system_cpu_time_percentage	gauge	Sistem CPU süresi yüzdesi

Metrik	Tip	Tanım
ibmmq_qmgr_topic_mqput_mqput1_total	counter	Konu MQPUT/MQPUT1 aralık toplamı
ibmmq_qmgr_topic_put_bytes_total	counter	Aralık toplam konu byte sayısı
ibmmq_qmgr_trace_file_system_free_space_percentage	gauge	MQ izleme kütüğü sistemi-boş alan
ibmmq_qmgr_trace_file_system_in_use_bytes	gauge	MQ trace file system - bytes in use
ibmmq_qmgr_user_cpu_time_estimate_for_queue_manager_percentage	gauge	Kullanıcı CPU süresi-kuyruk yöneticisi için yüzde tahmini
ibmmq_qmgr_user_cpu_time_percentage	gauge	Kullanıcı CPU süresi yüzdesi

CP4I V9.2.2 CD IBM MQ sertifikalı kapsayıcılar için Yerel HA kuyruk yöneticilerinin durumunu görüntüleme

IBM MQ sertifikalı kapsayıcılar için, çalışmakta olan Pds 'lerden birinde **dspmq** komutunu çalıştırarak Yerel HA eşgörünümünün durumunu görüntüleyebilirsiniz.

Bu görev hakkında

Önemli:

Bir kuyruk yöneticisi eşgörünümünün çalışma durumunu görüntülemek için, çalışmakta olan PDS 'lerden birinde **dspmq** komutunu kullanabilirsiniz. Döndürülen bilgiler, eşgörünümün etkin mi, yoksa eşleme mi olduğuna bağlıdır. Etkin örnek tarafından sağlanan bilgiler kesin, eşleme düğümlerinden alınan bilgiler güncel olmayabilir.

Aşağıdaki işlemleri gerçekleştirebilirsiniz:

- Geçerli düğümdeki kuyruk yöneticisi örneğinin etkin mi, yoksa eşleme mi olduğunu görüntüler.
- Yürürlükteki düğümdeki yönetim ortamının Yerel HA çalışma durumunu görüntüler.
- Yerel HA yapılandırmasındaki tüm üç örneğin çalışma durumunu görüntüleyin.

Yerel HA yapılanış durumunu raporlamak için aşağıdaki durum alanları kullanılır:

Görev

Eşgörünümün geçerli rolünü belirtir ve Active, Replicaya da Unknown' den biridir.

yönetim ortamı

The name provided for this instance of the queue manager when it was created using the **-lr** option of the **crtmqm** command.

İNSYNC

Yönetim ortamının gerektiğinde etkin yönetim ortamı olarak kabul edilip edilemeyeceğini belirtir.

ÇEKİRDEK

Yetersayı durumunu *number_of_instances_in-sync/number_of_instances_confitcale* biçiminde raporlar.

YANITLA

Kuyruk yöneticisi örneğinin eşleme adresi.

CONNECTV

Düğümün etkin eşgörünümle bağlı olup olmadığını belirtir.

BACKLOG

Eşgörünümün arkasında olduğu KB sayısını gösterir.

CONNINST

Belirtilen yönetim ortamının bu yönetim ortamına bağlı olup olmadığını gösterir.

TARİH DEĞERİ

Bu bilgilerin en son güncellendiği tarihi gösterir (hiçbir zaman güncellenmediyse boş değer).

ALTTIME

Bu bilgilerin en son güncellendiği zamanı belirtir (hiçbir zaman güncellenmediyse boş değer).

Yordam

- Kuyruk yöneticinizin bir parçası olan bölmeleri bulun.

```
oc get pod --selector app.kubernetes.io/instance=nativeha-qm
```

- Kapsüllerden birinde dspm q komutunu çalıştırın.

```
oc exec -t Pod dspm q
```

```
oc rsh Pod
```

dspm q ' u doğrudan çalıştırabileceğiniz etkileşimli bir kabuk için.

- Bir kuyruk yöneticisi örneğinin etkin eşgörünüm olarak mı, yoksa eşleme olarak mı çalıştırılacağını belirlemek için:

```
oc exec -t Pod dspm q -o status -m QMgrName
```

BOB adlı bir kuyruk yöneticisinin etkin bir eşgörünümü aşağıdaki durumu bildirecektir:

```
QMNAME(BOB)          STATUS(Running)
```

BOB adlı bir kuyruk yöneticisinin eşleme eşgörünümü aşağıdaki durumu bildirecektir:

```
QMNAME(BOB)          STATUS(Replica)
```

Etkin olmayan bir yönetim ortamı aşağıdaki durumu bildirecektir:

```
QMNAME(BOB)          STATUS(Ended Immediately)
```

- Belirtilen bölmede eşgörünümün yerel HA çalışma durumunu belirlemek için:

```
oc exec -t Pod dspm q -o nativeha -m QMgrName
```

BOB adlı bir kuyruk yöneticisinin etkin örneği, aşağıdaki durumu bildirebilir:

```
QMNAME(BOB)          ROLE(Active) INSTANCE(inst1) INSYNC(Yes) QUORUM(3/3)
```

BOB adlı bir kuyruk yöneticisinin eşleme örneği aşağıdaki durumu bildirebilir:

```
QMNAME(BOB)          ROLE(Replica) INSTANCE(inst2) INSYNC(Yes) QUORUM(2/3)
```

BOB adlı bir kuyruk yöneticisinin etkin olmayan bir eşgörünümü aşağıdaki durumu bildirebilir:

```
QMNAME(BOB)          ROLE(Unknown) INSTANCE(inst3) INSYNC(no) QUORUM(0/3)
```

- Yerel HA yapılandırmasındaki tüm eşgörünümlerin Yerel HA işletim durumunu belirlemek için:

```
oc exec -t Pod dspm q -o nativeha -x -m QMgrName
```

Bu komutu, kuyruk yöneticisi BOB ' un etkin örneğini çalıştıran düğümde yayıyorsunuz, aşağıdaki durumu alabilirsiniz:

```
QMNAME(BOB)          ROLE(Active) INSTANCE(inst1) INSYNC(Yes) QUORUM(3/3)
INSTANCE(inst1) ROLE(Active) REPLADDR(9.20.123.45) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
```

Bu komutu, kuyruk yöneticisi BOB ' un eşleme örneğini çalıştıran bir düğümde yayıyorsunuz, eşlemelerden birinin geride kaldığını belirten aşağıdaki durumu alabilirsiniz:

```
QMNAME(BOB)          ROLE(Replica) INSTANCE(inst2) INSYNC(Yes) QUORUM(2/3)
INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst1) ROLE(Active) REPLADDR(9.20.123.45) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(No) BACKLOG(435)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
```

Bu komutu, kuyruk yöneticisi BOB ' un etkin olmayan bir örneğini çalıştıran bir düğümde yayıyorsunuz, aşağıdaki durumu alabilirsiniz:

```
QMNAME(BOB)          ROLE(Unknown) INSTANCE(inst3) INSYNC(no) QUORUM(0/3)
INSTANCE(inst1) ROLE(Unknown) REPLADDR(9.20.123.45) CONNACTV(Unknown) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATA() ALTTIME()
INSTANCE(inst2) ROLE(Unknown) REPLADDR(9.20.123.46) CONNACTV(Unknown) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATA() ALTTIME()
INSTANCE(inst3) ROLE(Unknown) REPLADDR(9.20.123.47) CONNACTV(No) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATA() ALTTIME()
```

Yönetim ortamları hala etkin olan ve eşlemeler olan bir yönetim ortamı üzerinde çalışmaya devam ederken komutu yayınlıyorsanız, aşağıdaki durumu alırsınız:

```
QMNAME(BOB)          STATUS(Negotiating)
```

İlgili başvurular

[dspmq \(kuyruk yöneticilerini görüntüle\) komutu](#)

“Örnek: Yerel HA kuyruk yöneticisinin yapılandırılması” sayfa 93

This example shows how you deploy a queue manager using the native high availability feature into the Red Hat OpenShift Container Platform (OCP) using the IBM MQ Operator.

Backing up and restoring queue manager configuration using the Red Hat OpenShift CLI

Kuyruk yöneticisi yapılandırmalarının yedeklenmesi, kuyruk yöneticisi yapılandırmalarının kaybolması durumunda, kuyruk yöneticisini tanımlarından yeniden oluşturmanıza yardımcı olabilir. Bu yordam kuyruk yöneticisi günlük verilerini yedeklemez. İletilerin geçici niteliği nedeniyle, geçmiş günlük verilerinin geri yükleme sırasında ilgisiz olması olasıdır.

Başlamadan önce

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

Yordam

- Kuyruk yöneticisi yapılandırması yedekle.

Bir IBM MQ kuyruk yöneticisinin yapılandırmasını dökümünü almak için **dmpmqcfig** komutunu kullanabilirsiniz.

- a) Kuyruk yöneticinizin kapsülünün adını alın.

Örneğin, aşağıdaki komutu çalıştırabilirsiniz; burada *queue_manager_name* , QueueManager kaynağınız adıdır:

```
oc get pods --selector app.kubernetes.io/name=ibm-mq,app.kubernetes.io/instance=queue_manager_name
```

b) Kapsülde **dmpmqcfig** komutunu çalıştırın, çıkışı yerel makinenizdeki bir dosyaya yönlendirin.

dmpmqcfig , kuyruk yöneticisinin MQSC yapılandırmasını çıkarır.

```
oc exec -it pod_name -- dmpmqcfig > backup.mqsc
```

- Kuyruk yöneticisi yapılandırmasını geri yükleyin.

Önceki adımda belirtilen yedekleme yordamını izlemiş olması, kuyruk yöneticisi yapılandırmasını içeren bir `backup.mqsc` kütüğüne sahip olmalıdır. Bu dosyayı yeni bir kuyruk yöneticisine uygulayarak yapılandırmayı geri yükleyebilirsiniz.

a) Kuyruk yöneticinizin kapsülünün adını alın.

Örneğin, aşağıdaki komutu çalıştırabilirsiniz; burada *queue_manager_name* , QueueManager kaynağınız adıdır:

```
oc get pods --selector app.kubernetes.io/name=ibm-mq,app.kubernetes.io/instance=queue_manager_name
```

b) Modülde **runmqsc** komutunu çalıştırın, `backup.mqsc` dosyasının içeriğini yönlendirin.

```
oc exec -i pod_name -- runmqsc < backup.mqsc
```

OpenShift

CP4I

IBM MQ Operatorile ilgili sorunların giderilmesi

IBM MQ Operatorile ilgili bir sorun yaşıyorsanız, tanılamaya ve çözümenize yardımcı olması için açıklanan teknikleri kullanın.

Yordam

- [“Sorun giderme: Kuyruk yöneticisi verilerine erişim elde edilmesi” sayfa 123](#)

OpenShift

CP4I

Sorun giderme: Kuyruk yöneticisi verilerine erişim elde edilmesi

Kuyruk yöneticisi bölmesine uzak bir kabuğun oluşturulmadığı bir kuyruk yöneticisi PVC 'deki dosyalara erişim elde etmek için PVC denetçi aracını kullanın. Bunun nedeni bölmesinin **Error** ya da **CrashLoopBackOff** durumunda olması olabilir. Bu araç, IBM MQ Operatortarafından konuşlandırılan kuyruk yöneticileriyle kullanılmak üzere tasarlanmıştır.

Başlamadan önce

PVC denetçi aracını kullanmak için. Kuyruk yöneticisi ad alanınıza erişiminiz olmalıdır.

Bu görev hakkında

Sorun gidermeye yardımcı olmak için, belirli bir kuyruk yöneticisiyle ilişkili Kalıcı Birim Talepleri (PVC) üzerinde saklanan verilere erişebilirsiniz. Bunu yapmak için, PVC 'leri bir dizi denetçi bölmesine monte etmek için bir araç kullanırsınız. Daha sonra dosyaları okumak için herhangi bir denetçi bölmesine uzak bir kabuk alabilirsiniz.

Devreye alma tipine bağlı olarak, bir ile üç denetçi bölmesi arasında oluşturulur. Bir Yerel HA ya da Çok Eşgörünümlü kuyruk yöneticisinin belirli bir bölmesine özgü birimler, ilişkili PVC denetçi kapsülünde kullanılabilir. Paylaşılan birimler tüm inceleyicilerden kullanılabilir. Denetçi bölmesinin adı, ilişkili kuyruk yöneticisi bölmesinin adını içerir.

Yordam

1. MQ PVC Inspector aracını karşıdan yükleyin.

Araç burada bulunur: <https://github.com/ibm-messaging/mq-pvc-tool>.

2. Kümenizde oturum açtığınızdan emin olun.

3. Kuyruk yöneticisinin adını ve kuyruk yöneticisinin çalıştığı ad alanını bulun.

4. Denetçi aracını kuyruk yöneticiniz için çalıştırın.

a) Kuyruk yöneticisi adınızı ve ad alanı adını belirterek aşağıdaki komutu çalıştırın.

```
./pvc-tool.sh queue_manager_name queue_manager_namespace_name
```

b) Araç tamamlandıktan sonra, yaratılmakta olan denetçi bölmesini görüntülemek için aşağıdaki komutu çalıştırın.

```
oc get pods
```

5. Denetçi bölmesine takılan dosyaları görüntüleyin.

a) Her PVC denetçi bölmesi bir kuyruk yöneticisi bölmesiyle ilişkilendirilmiştir, bu nedenle birden çok denetçi bölmesi olabilir. Aşağıdaki komutu çalıştırarak bu bölmlerden birine erişin:

```
oc rsh pvc-inspector-pod-name
```

Bağlı PVC dizinlerini içeren dizine yerleştirildiniz.

b) Aşağıdaki komutu çalıştırarak bölmeye uzak bir kabuk açın:

```
ls
```

c) Takılan PVC 'lerle aynı ada sahip dizinleri görebilirsiniz. Bu dizinlere göz atarak kuyruk yöneticisi PVC 'lerindeki dosyalara erişin. PVC 'lerin listesini görmek için, uzak kabuk oturumunun dışında aşağıdaki komutu çalıştırın:

```
oc get pvc
```

d) Aşağıdaki komutu çalıştırarak araç tarafından oluşturulan bölkları temizleyin:

```
'oc delete pods -l tool=mq-pvc-inspector
```

OpenShift CP4I IBM MQ Operator için API başvurusu

IBM MQ , Red Hat OpenShift Container Platform ile yerel bütünleştirme sağlayan bir Kubernetes Operator sağlar.

OpenShift CP4I mq.ibm.com/v1beta1 için API başvurusu

v1beta1 API 'si, QueueManager kaynaklarını yaratmak ve yönetmek için kullanılabilir.

OpenShift CP4I CD EUS mq.ibm.com/v1beta1 için lisans başvurusu

Geçerli lisans sürümleri

spec.license.license alanı, kabul etmekte olduğunuz lisansa ilişkin lisans tanıtıcısını içermelidir. Geçerli değerler şunlardır:

Değer: spec.license.1 icense	Değer: spec.license.u se	Lisans bilgisi	Geçerli IBM MQ sürümleri
L-RJON-C7QG3S	Production ya da NonProduction	IBM Cloud Pak for Integration 2021.4.1	9.2.4 ya da 9.2.5
L-RJON-C7QFZX	Production ya da NonProduction	IBM Cloud Pak for Integration Sınırlı Sürüm 2021.4.1	9.2.4 ya da 9.2.5
L-RJON-C5CSNH	Production ya da NonProduction	IBM Cloud Pak for Integration 2021.3.1	9.2.3 ya da 9.2.4
L-RJON-C5CSM2	Production ya da NonProduction	IBM Cloud Pak for Integration Limited Edition 2021.3.1	9.2.3 ya da 9.2.4
L-RJON-BZFQU2	Production ya da NonProduction	IBM Cloud Pak for Integration 2021.2.1	9.2.3
L-RJON-BZFQSB	Production ya da NonProduction	IBM Cloud Pak for Integration Limited Edition 2021.2.1	9.2.3
L-RJON-BUVMQX	Production ya da NonProduction	IBM Cloud Pak for Integration 2020.4.1	9.2.0 EUS ya da 9.2.1
L-RJON-BUVMYB	Production ya da NonProduction	IBM Cloud Pak for Integration Sınırlı Sürüm 2020.4.1	9.2.0 EUS ya da 9.2.1
L-APIG-BZDDDY	Production	IBM MQ Advanced and IBM MQ Advanced for Non-Production Environment 9.2 - 07/2021	9.2.3, 9.2.4 ya da 9.2.5
L-APIG-BYHCL7	Development	IBM MQ Advanced for Developers (Non- Warranted) V9.2 -07/2021	9.2.3, 9.2.4 ya da 9.2.5
L-APIG-BVJJB3	Production	IBM MQ Advanced and IBM MQ Advanced for Non-Production Environment 9.2 - 03/2021	9.2.2
L-APIG-BMJJBM	Production	IBM MQ Advanced V9.2	9.2.0 CD ya da 9.2.1
L-APIG-BMKG5H	Development	IBM MQ Advanced for Developers (Non- Warranted) V9.2	9.2.0 CD, 9.2.1 ya da 9.2.2

Note that the license *sürüm* is specified, which is not always the same as the version of IBM MQ.

Eski lisans sürümleri

spec.license.license alanı, kabul etmekte olduğunuz lisansa ilişkin lisans tanıtıcısını içermelidir. Geçerli değerler şunlardır:

Değer: spec.license.1 icense	Değer: spec.license.u se	Lisans bilgisi	Geçerli IBM MQ sürümleri
L-RJON-BXUPZ2	Production ya da NonProduction	IBM Cloud Pak for Integration 2021.1.1	9.2.2
L-RJON-BXUQ34	Production ya da NonProduction	IBM Cloud Pak for Integration Limited Edition 2021.1.1	9.2.2

Değer: spec.license.1 license	Değer: spec.license.uses	Lisans bilgisi	Geçerli IBM MQ sürümleri
L-RJON-BYRMYW	NonProduction	IBM Cloud Pak for Integration Eval-Gösterim 2021.1.1. Yalnızca IBM MQ Operator 1.5 ile Native HA ile kullanım için erken yayın düzeyi.	9.2.2
L-RJON-BQPGWD	Production ya da NonProduction	IBM Cloud Pak for Integration 2020.3.1	9.2.0 CD 'si
L-RJON-BN7PN3	Production ya da NonProduction	IBM Cloud Pak for Integration 2020.2.1	9.1.5 ya da 9.2.0 CD 'si
L-RJON-BPHL2Y	Production ya da NonProduction	IBM Cloud Pak for Integration Limited Edition 2020.2.1	9.1.5
L-APIG-BJAKBF	Production	IBM MQ Advanced V9.1 -04/2020	9.1.5
L-APIG-BM7GDH	Development	IBM MQ Advanced for Developers (Non-Warranted) V9.1 -04/2020	9.1.5

Note that the license *sürüm* is specified, which is not always the same as the version of IBM MQ.

  **QueueManager için API başvurusu (mq.ibm.com/v1beta1)**

QueueManager

QueueManager , uygulamalara kuyruğa alma ve yayınlama/abone olma hizmetleri sağlayan bir IBM MQ sunucudur.

Alan	Tanım
apiVersion dizgi	APIVersion, bir nesnenin bu gösterimine ilişkin sürümü oluşturulan şemayı tanımlar. Sunucular, tanınan şemaları en son iç değere dönüştürmeli ve tanınmayan değerleri reddedebilir. Daha fazla bilgi: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources .
kind dizgi	Tür, bu nesnenin temsil ettiği REST kaynağını temsil eden bir dizgi değeridir. Sunucular, bu durumu istemci göndereceği uç noktadan çıkartabilir. Güncellenemez. CamelCase içinde. Daha fazla bilgi: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-çeşit .
metadata	
spec QueueManagerBelirtimi	QueueManager' un istenen durumu.
status QueueManagerDurumu	The observed state of the QueueManager.

.spec

QueueManager' un istenen durumu.

Burada görünür:

- [“QueueManager” sayfa 126](#)

Alan	Tanım
affinity	Standart Kubernetes benzeşim kuralları. Daha fazla bilgi için bkz. https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.17/#affinity-v1-core .
annotations <u>Ek Açıklamalar</u>	Ek açıklamalar alanı, Pod ek açıklamaları için bir düzgeçiş işlevi olarak işlev görür. Kullanıcılar, bu alana herhangi bir ek açıklama ekleyebilir ve bu ek açıklamayı Pod 'a uygulayabilir. Burada ek açıklamalar, varsa, varsayılan ek açıklamaların üzerine yazılır. MQ Operator 1.3.0 sürümünü ya da üstünü gerektirir.
imagePullSecrets <u>LocalObjectBaşvuru dizisi</u>	Bu QueueManagertarafından kullanılan görüntülerin herhangi birini çekmek için kullanılacak aynı ad alanındaki sırlara yapılan başvuruların isteğe bağlı bir listesi. Belirtirse, bu sırların kullanması için tek tek puller uygulamalarına geçilecek. Örneğin, docker durumunda, yalnızca DockerConfig tipi sırlar onurlandırılır. Daha fazla bilgi için bkz. https://kubernetes.io/docs/concepts/containers/images#specifying-imagepullsırlar-on-a-pod .
labels <u>Etiketler</u>	Etiketler alanı, Pod etiketleri için düzgeçiş olarak işlev görür. Kullanıcılar bu alana herhangi bir etiket ekleyebilir ve bu etiketi Pod 'a uygulayabilir. Burada etiketler, varsa varsayılan etiketlerin üzerine yazılır. MQ Operator 1.3.0 sürümünü ya da üstünü gerektirir.
license <u>Lisans</u>	Lisansı kabul ettiğinizi ve hangi lisans ölçümlerinin kullanılacağını denetleyen ayarlar.
pki <u>PKI</u>	Anahtarları ve sertifikaları İletim Arabirim Katmanı Güvenliği (TLS) ya da MQ Advanced Message Security (AMS) ile kullanmak üzere tanımlamak için genel anahtar altyapısı ayarları.
queueManager <u>QueueManagerYapılandırması</u>	Kuyruk Yöneticisi taşıyıcısı ve temeldeki kuyruk yöneticisine ilişkin ayarlar.
securityContext <u>SecurityContext</u>	Kuyruk Yöneticisi Pod 'sına eklenecek güvenlik ayarları securityContext.
template <u>Şablon</u>	Kubernetes kaynakları için gelişmiş şablon oluşturma. The template allows users to override how IBM MQ generates the underlying Kubernetes resources, such as StatefulSet, Pods and Services. Bu, yalnızca ileri düzey kullanıcılar içindir; yanlış kullanıldığında MQ ' nun olağan çalışmasını kesintiye uğratabilecek bir potansiyeldir. QueueManager kaynağındaki başka bir yerde belirlenen değerler, şablondaki ayarlarla geçersiz kılınır.
terminationGracePeriod <u>Seconds tamsayı</u>	İsteğe bağlı olarak, Pod 'un zarafetle sonlandırılması gereken saniye cinsinden süre. Değer eksi olmayan bir tamsayı olmalıdır. Sıfır değeri, silmeyi hemen gösterir. Kuyruk yöneticisinin sona erdirilmesinin denendiği hedef zaman, uygulama bağlantısını kesme aşamaları üst seviyeye iletmeye çalışılır. Gerekirse, önemli kuyruk yöneticisi bakım görevleri kesintiye uğratılır. Varsayılan değer 30 saniyedir.
tracing <u>TracingConfig</u>	Cloud Pak for Integration Operations Dashboard ile bütünleştirme bütünleştirmesine ilişkin ayarlar.
version <u>dizgi</u>	Kullanılacak MQ sürümünü denetleyen ayar (gerekli). Örneğin: 9.1.5.0-r2 , taşıyıcı görüntüsünün ikinci düzeltmesini kullanarak MQ sürüm 9.1.5.0değerini belirtirdi. Taşıyıcıya özgü düzeltmeler, temel görüntüye ilişkin düzeltmeler gibi, sık sık düzeltmelere uygulanır.
web <u>WebServerYapılandırması</u>	MQ web sunucusuna ilişkin ayarlar.

.spec.annotations

Ek açıklamalar alanı, Pod ek açıklamaları için bir düzgeçiş işlevi olarak işlev görür. Kullanıcılar, bu alana herhangi bir ek açıklama ekleyebilir ve bu ek açıklamayı Pod 'a uygulayabilir. Burada ek açıklamalar, varsa, varsayılan ek açıklamaların üzerine yazılır. MQ Operator 1.3.0 sürümünü ya da üstünü gerektirir.

Burada görünür:

- [“.spec” sayfa 126](#)

.spec.imagePullSecrets

LocalObjectReferansı, aynı ad alanı içinde başvuru nesneyi bulmanıza olanak verecek yeterli bilgiyi içerir.

Burada görünür:

- [“.spec” sayfa 126](#)

Alan	Tanım
name dizgi	Başvurunun adı. Daha fazla bilgi: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names TODO: Diğer yararlı alanları ekleyin. apiVersion, tür, uid?

.spec.labels

Etiketler alanı, Pod etiketleri için düzgeçiş olarak işlev görür. Kullanıcılar bu alana herhangi bir etiket ekleyebilir ve bu etiketi Pod 'a uygulayabilir. Burada etiketler, varsa varsayılan etiketlerin üzerine yazılır. MQ Operator 1.3.0 sürümünü ya da üstünü gerektirir.

Burada görünür:

- [“.spec” sayfa 126](#)

.spec.license

Lisansı kabul ettiğinizi ve hangi lisans ölçümlerinin kullanılacağını denetleyen ayarlar.

Burada görünür:

- [“.spec” sayfa 126](#)

Alan	Tanım
accept boole	Bu yazılımla ilişkili lisansı kabul edip etmediğiniz (gerekli).
license dizgi	Kabul ettiğiniz lisansın tanıtıcısı. Bu, kullanmakta olduğunuz MQ sürümüne ilişkin doğru lisans tanıtıcısı olmalıdır. Geçerli değerler için http://ibm.biz/BdqvCF adresine bakın.
metric dizgi	Hangi lisans metriğinin kullanılacağını belirten ayar. Örneğin, ProcessorValueUnit, VirtualProcessorCore ya da ManagedVirtualServer. Defaults to ProcessorValueUnit when using an MQ license and VirtualProcessorCore when using a Cloud Pak for Integration license.
use dizgi	Lisansın birden çok kullanımla desteklendiği durumlarda yazılımın nasıl kullanılacağına ilişkin denetimler ayarlanıyor. Geçerli değerler için http://ibm.biz/BdqvCF adresine bakın.

.spec.pki

Anahtarları ve sertifikaları İletim Arabirim Katmanı Güvenliği (TLS) ya da MQ Advanced Message Security (AMS) ile kullanmak üzere tanımlamak için genel anahtar altyapısı ayarları.

Burada görünür:

- [“.spec” sayfa 126](#)

Alan	Tanım
keys PKISource dizisi	Kuyruk Yöneticisi 'nin anahtar havuzuna eklemek için özel anahtarlar.
trust PKISource dizisi	Kuyruk Yöneticisi 'nin anahtar havuzuna eklenecek sertifikalar.

.spec.pki.keys

PKISource, anahtarlar ya da sertifikalar gibi bir Genel Anahtar Altyapısı bilgileri kaynağı tanımlar.

Burada görünür:

- [“.spec.pki” sayfa 129](#)

Alan	Tanım
name dizgi	Ad, anahtar ya da sertifika için etiket olarak kullanılır. Küçük harfli alfasayısal bir dizgi olmalıdır.
secret Gizli	Kubernetes Güvenlik Dizgisini kullanarak bir anahtar sağlayın.

.spec.pki.keys.secret

Kubernetes Güvenlik Dizgisini kullanarak bir anahtar sağlayın.

Burada görünür:

- [“.spec.pki.keys” sayfa 129](#)

Alan	Tanım
items dizi	Keys inside the Kubernetes secret which should be added to the Queue Manager container.
secretName dizgi	Kubernetes gizli adının adı.

.spec.pki.trust

PKISource, anahtarlar ya da sertifikalar gibi bir Genel Anahtar Altyapısı bilgileri kaynağı tanımlar.

Burada görünür:

- [“.spec.pki” sayfa 129](#)

Alan	Tanım
name dizgi	Ad, anahtar ya da sertifika için etiket olarak kullanılır. Küçük harfli alfasayısal bir dizgi olmalıdır.
secret Gizli	Kubernetes Güvenlik Dizgisini kullanarak bir anahtar sağlayın.

.spec.pki.trust.secret

Kubernetes Güvenlik Dizgisini kullanarak bir anahtar sağlayın.

Burada görünür:

- [“.spec.pki.trust” sayfa 129](#)

Alan	Tanım
items dizi	Keys inside the Kubernetes secret which should be added to the Queue Manager container.
secretName dizgi	Kubernetes gizli adının adı.

.spec.queueManager

Kuyruk Yöneticisi taşıyıcısı ve temeldeki kuyruk yöneticisine ilişkin ayarlar.

Burada görünür:

- [“.spec” sayfa 126](#)

Alan	Tanım
availability Kullanılabilirlik	Kuyruk yöneticisine ilişkin kullanılabilirlik ayarları; örneğin, etkin-bekleme çiftinin ya da yerel yüksek kullanılabilirliğin kullanılması gibi.
debug boole	Hata ayıklama iletilerinin taşıyıcıya özel koddan, taşıyıcı günlüğüne kaydedilip kaydedilmeyeceğini belirleyin. Varsayılan değer yanlış (false) olarak ayarlanır.
image dizgi	Kullanılacak taşıyıcı görüntüsü.
imagePullPolicy dizgi	kubelet belirtilen görüntüyü çekmeye çalışıldığında bu kontrollerin ayarlanması. Varsayılan olarak IfNotPresent değerine ayarlanır.
ini INISource dizisi	Kuyruk Yöneticisi için INI sağlamaya ilişkin ayarlar. MQ Operator 1.1.0 ya da üstünü gerektirir.
livenessProbe QueueManagerLivenessProbe	Hızlılık sondasını denetleyen ayarlar.
logFormat dizgi	Bu taşıyıcı için kullanılacak günlük biçimi. Taşıyıcıdaki JSON biçimli günlükler için JSON simgesini kullanın. Metin biçimli iletiler için Basic 'ı kullanın. Varsayılan olarak Basic değerine ayarlanır.
metrics QueueManagerMetrikleri	Prometheusstili metriklerle ilişkin ayarlar.
mqsc MQSCSSource dizisi	Kuyruk Yöneticisi için MQSC sağlamaya ilişkin ayarlar. MQ Operator 1.1.0 ya da üstünü gerektirir.
name dizgi	Temeldeki MQ Kuyruk Yöneticisi 'nin adı (metadata.name ögesinden farklıysa). Ad için Kubernetes kurallarına uymayan bir Kuyruk Yöneticisi adı istiyorsanız bu alanı kullanın (örneğin, temel harfler içeren bir ad).
readinessProbe QueueManagerReadinessProbe	Hazır olma durumunu denetleyen ayarlar.
resources Kaynaklar	Kaynak gereksinimlerini denetleyen ayarlar.
route Rota	Kuyruk Yöneticisi rotasına ilişkin ayarlar. MQ Operator 1.4.0 ya da üstünü gerektirir.
startupProbe StartupProbe	Başlatma araştırmasını denetleyen ayarlar. Yalnızca MultiInstance ve NativeHA konuşlandırmaları için geçerlidir. MQ Operator 1.5.0 sürümünü ya da üstünü gerektirir.
storage QueueManagerSaklama Alanı	Kuyruk Yöneticisi 'nin kalıcı hacimleri ve depolama sınıflarını kullanımını denetlemek için kullanılan depolama ayarları.

.spec.queueManager.availability

Kuyruk yöneticisine ilişkin kullanılabilirlik ayarları; örneğin, etkin-bekleme çiftinin ya da yerel yüksek kullanılabilirliğin kullanılması gibi.

Burada görünür:

- [“.spec.queueManager” sayfa 130](#)

Alan	Tanım
<code>tls Tls</code>	NativeHA eşlemeleri arasında güvenli iletişimi yapılandırmak için kullanılan isteğe bağlı TLS ayarları. MQ Operator 1.5.0 sürümünü ya da üstünü gerektirir.
<code>type</code> dizgi	Kullanılacak kullanılabilirliğin tipi. Otomatik olarak (bazı durumlarda) Kubernetes tarafından yeniden başlatılacak tek bir Pod için <code>SingleInstance</code> 'yi kullanın. Use <code>MultiInstance</code> for a pair of Pods, one of which is the active Queue Manager, and the other of which is a standby. Yerel yüksek kullanılabilirlik eşlemesi için <code>NativeHA</code> kullanın (MQ Operator 1.5.0 ya da sonraki bir sürümü gerektirir). Varsayılan olarak <code>SingleInstance</code> değerine ayarlanır. Daha ayrıntılı bilgi için http://ibm.biz/BdQAQa adresine bakın.
<code>updateStrategy</code> dizgi	<code>MultiInstance</code> ve <code>NativeHA</code> Kuyruk Yöneticileri için kullanılacak güncelleme stratejisi. Kuyruk Yöneticisi yapılandırması her değiştiğinde otomatik olarak kaybolan güncellemeleri etkinleştirmek için <code>RollingUpdate</code> seçeneğini kullanın. Otomatik yuvarlama güncellemelerini devre dışı bırakmak için <code>OnDelete</code> 'u kullanın, Kuyruk Yöneticisi değişiklikleri yalnızca Pods silindiğinde uygulanacaktır (dış etkenler tarafından tetiklenen Pod silmeleri dahil). Varsayılan olarak <code>RollingUpdate</code> değerine ayarlanır. MQ Operator 1.6.0 ya da üstünü gerektirir.

.spec.queueManager.availability.tls

NativeHA eşlemeleri arasında güvenli iletişimi yapılandırmak için kullanılan isteğe bağlı TLS ayarları. MQ Operator 1.5.0 sürümünü ya da üstünü gerektirir.

Burada görünür:

- [“.spec.queueManager.availability” sayfa 131](#)

Alan	Tanım
<code>cipherSpec</code> dizgi	NativeHA TLS için <code>CipherSpec</code> 'in adı.
<code>secretName</code> dizgi	Kubernetes gizli adının adı.

.spec.queueManager.ini

INI yapılandırma dosyalarının kaynağı.

Burada görünür:

- [“.spec.queueManager” sayfa 130](#)

Alan	Tanım
<code>configMap</code> <code>ConfigMapINISource</code>	<code>ConfigMap</code> , INI bilgilerini içeren bir Kubernetes <code>ConfigMap</code> 'i temsil eder.
<code>secret</code> <code>SecretINISource</code>	Güvenlik dizgisi, INI bilgilerini içeren bir Kubernetes <code>Secret</code> 'ını temsil eder.

.spec.queueManager.ini.configMap

`ConfigMap` , INI bilgilerini içeren bir Kubernetes `ConfigMap` 'i temsil eder.

Burada görünür:

- [“.spec.queueManager.ini” sayfa 131](#)

Alan	Tanım
items dizi	Uygulanması gereken Kubernetes kaynağının içindeki tuşlar.
name dizgi	Kubernetes kaynağının adı.

.spec.queueManager.ini.secret

Güvenlik dizgisi, INI bilgilerini içeren bir Kubernetes Secret 'ını temsil eder.

Burada görünür:

- [“.spec.queueManager.ini” sayfa 131](#)

Alan	Tanım
items dizi	Uygulanması gereken Kubernetes kaynağının içindeki tuşlar.
name dizgi	Kubernetes kaynağının adı.

.spec.queueManager.livenessProbe

Hızlılık sondasını denetleyen ayarlar.

Burada görünür:

- [“.spec.queueManager” sayfa 130](#)

Alan	Tanım
failureThreshold tamsayı	Araştırmacı başarılı olduktan sonra başarısız olarak değerlendirilecek art arda başarısızlıklar alt sınırı. Varsayılan değer 1 'e ayarlanır.
initialDelaySeconds tamsayı	Konteyner başlatılmadan önce başlatıldıktan sonra geçecek süre (saniye). Defaults to 90 seconds for SingleInstance. MultiInstance ve NativeHA konuşlandırmaları için varsayılan değer 0 saniye olarak ayarlanır. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .
periodSeconds tamsayı	Sondayı (saniye cinsinden) ne sıklıkta (saniye cinsinden) gerçekleştirmek için? Varsayılan değer 10 saniyedir.
successThreshold tamsayı	Sondanın başarısız olduktan sonra başarılı sayılması için arka arkaya en az başarı elde edilmesi gerekir. Varsayılan değer 1 'e ayarlanır.
timeoutSeconds tamsayı	Araştırmacı zamanaşımına geçtikten sonra geçecek süre (saniye). Varsayılan değer 5 saniyedir. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .

.spec.queueManager.metrics

Prometheusstili metriklerle ilişkin ayarlar.

Burada görünür:

- [“.spec.queueManager” sayfa 130](#)

Alan	Tanım
enabled boole	Prometheus-uyumlu metrikler için bir uç noktenin etkinleştirilip etkinleştirilmeyeceğini. Varsayılan değer doğru (true) olarak ayarlanır.

.spec.queueManager.mqsc

MQSC yapılanış kütüklerinin kaynağı.

Burada görünür:

- [“.spec.queueManager” sayfa 130](#)

Alan	Tanım
configMap ConfigMapMQSCSource	ConfigMap , MQSC bilgilerini içeren bir Kubernetes ConfigMap ' i gösterir.
secret SecretMQSCSource	Güvenlik dizgisi, MQSC bilgilerini içeren bir Kubernetes Secret ögesini temsil eder.

.spec.queueManager.mqsc.configMap

ConfigMap , MQSC bilgilerini içeren bir Kubernetes ConfigMap ' i gösterir.

Burada görünür:

- [“.spec.queueManager.mqsc” sayfa 133](#)

Alan	Tanım
items dizi	Uygulanması gereken Kubernetes kaynağının içindeki tuşlar.
name dizgi	Kubernetes kaynağının adı.

.spec.queueManager.mqsc.secret

Güvenlik dizgisi, MQSC bilgilerini içeren bir Kubernetes Secret ögesini temsil eder.

Burada görünür:

- [“.spec.queueManager.mqsc” sayfa 133](#)

Alan	Tanım
items dizi	Uygulanması gereken Kubernetes kaynağının içindeki tuşlar.
name dizgi	Kubernetes kaynağının adı.

.spec.queueManager.readinessProbe

Hazır olma durumunu denetleyen ayarlar.

Burada görünür:

- [“.spec.queueManager” sayfa 130](#)

Alan	Tanım
failureThreshold tamsayı	Araştırıcı başarılı olduktan sonra başarısız olarak değerlendirilecek art arda başarısızlıklar alt sınırı. Varsayılan değer 1 'e ayarlanır.
initialDelaySeconds tamsayı	Konteyner başlatılmadan önce başlatıldıktan sonra geçecek süre (saniye). Defaults to 10 seconds for SingleInstance. Defaults to 0 for MultiInstance and NativeHA deployments. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .
periodSeconds tamsayı	Sondayı (saniye cinsinden) ne sıklıkta (saniye cinsinden) gerçekleştirmek için? Varsayılan değer 5 saniyedir.

Alan	Tanım
successThreshold tamsayı	Sondanın başarısız olduktan sonra başarılı sayılması için arka arkaya en az başarı elde edilmesi gerekir. Varsayılan değer 1 'e ayarlanır.
timeoutSeconds tamsayı	Araştırmacı zamanaşımına geçtikten sonra geçecek süre (saniye). Varsayılan değer 3 saniyedir. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .

.spec.queueManager.resources

Kaynak gereksinimlerini denetleyen ayarlar.

Burada görünür:

- [“.spec.queueManager” sayfa 130](#)

Alan	Tanım
limits Sınırlar	CPU & bellek ayarları.
requests İstekler	CPU & bellek ayarları.

.spec.queueManager.resources.limits

CPU & bellek ayarları.

Burada görünür:

- [“.spec.queueManager.resources” sayfa 134](#)

Alan	Tanım
cpu	
memory	

.spec.queueManager.resources.requests

CPU & bellek ayarları.

Burada görünür:

- [“.spec.queueManager.resources” sayfa 134](#)

Alan	Tanım
cpu	
memory	

.spec.queueManager.route

Kuyruk Yöneticisi rotasına ilişkin ayarlar. MQ Operator 1.4.0 ya da üstünü gerektirir.

Burada görünür:

- [“.spec.queueManager” sayfa 130](#)

Alan	Tanım
enabled boole	Rotanın etkinleştirilip etkinleştirilmeyeceğini. Varsayılan değer doğru (true) olarak ayarlanır.

.spec.queueManager.startupProbe

Başlatma araştırmasını denetleyen ayarlar. Yalnızca MultiInstance ve NativeHA konuşlandırmaları için geçerlidir. MQ Operator 1.5.0 sürümünü ya da üstünü gerektirir.

Burada görünür:

- [“.spec.queueManager” sayfa 130](#)

Alan	Tanım
failureThreshold tamsayı	Sondanın başarısız olarak kabul edilmesi için art arda yapılan arızalar alt sınırı. Varsayılan değer 60 'a ayarlanır.
initialDelaySeconds tamsayı	Konteyner başlatılmadan önce başlatıldıktan sonra geçecek süre (saniye). Varsayılan değer 0 saniye olarak ayarlanır. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .
periodSeconds tamsayı	Sondayı (saniye cinsinden) ne sıklıkta (saniye cinsinden) gerçekleştirmek için? Varsayılan değer 5 saniyedir.
successThreshold tamsayı	Sondanın başarılı sayılması için ardışık art arda başarı sayısı alt sınırı. Varsayılan değer 1 'e ayarlanır.
timeoutSeconds tamsayı	Araştırıcı zamanaşımına geçtikten sonra geçecek süre (saniye). Varsayılan değer 5 saniyedir. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .

.spec.queueManager.storage

Kuyruk Yöneticisi 'nin kalıcı hacimleri ve depolama sınıflarını kullanımını denetlemek için kullanılan depolama ayarları.

Burada görünür:

- [“.spec.queueManager” sayfa 130](#)

Alan	Tanım
defaultClass dizgi	Varsayılan olarak, bu Kuyruk Yöneticisinin tüm kalıcı birimlerine uygulanacak depolama sınıfı. Belirli kalıcı birimler, bu varsayılan depolama sınıfı ayarını geçersiz kılacak kendi depolama sınıfını tanımlayabilir. type of availability SingleInstance ya da NativeHAise, depolama sınıfı ReadWrite(Okuma Yazma) ya da ReadWrite(Okuma Yazma) türlerinden biri olabilir. type of availability MultiInstanceise, depolama sınıfı çok sayıda ReadWritetipinde olmalıdır.
defaultDeleteClaim boole	Kuyruk yöneticisi silindiğinde tüm birimlerin silinip silinmeyeceğini belirleyin. Belirli kalıcı birimler, deleteClaim için kendi değerini tanımlayabilir ve bu değer, bu defaultDeletetalep ayarını geçersiz kılar. Varsayılan değer yanlış (false) olarak ayarlanır.
persistedData QueueManagerOptionalVolume	Yapılandırma, kuyruklar ve iletiler de içinde olmak üzere MQ kalıcı verileri için PersistentVolume ayrıntıları. Çok eşgörünümlü kuyruk yöneticisi kullanılırken gereklidir.
queueManager QueueManagerBirimi	/var/mqmtındaki normalde herhangi bir veri için varsayılan PersistentVolume . Başka bir birim belirtilmediyse, tüm kalıcı verileri ve kurtarma günlüklerini içerir.
recoveryLogs QueueManagerOptionalVolume	MQ kurtarma günlüklerine ilişkin kalıcı birim ayrıntıları. Çok eşgörünümlü kuyruk yöneticisi kullanılırken gereklidir.

.spec.queueManager.storage.persistedData

Yapılandırma, kuyruklar ve iletiler de içinde olmak üzere MQ kalıcı verileri için PersistentVolume ayrıntıları. Çok eşgözümlü kuyruk yöneticisi kullanılırken gereklidir.

Burada görünür:

- [“.spec.queueManager.storage” sayfa 135](#)

Alan	Tanım
class dizgi	Bu birim için kullanılacak depolama sınıfı. Yalnızca type persistent-claimise geçerlidir. type of availability SingleInstance ya da NativeHAise, depolama sınıfı ReadWrite(Okuma Yazma) ya da ReadWrite(Okuma Yazma) türlerinden biri olabilir. type of availability MultiInstanceise, depolama sınıfı çok sayıda ReadWritetipinde olmalıdır.
deleteClaim boole	Kuyruk yöneticisi silindiğinde bu birimin silinip silinmeyeceğini belirleyin.
enabled boole	Bu birimin ayrı bir birim olarak etkinleştirilip etkinleştirilmeyeceğini ya da varsayılan queueManager birimine yerleştirilip yerleştirilmeyeceğini belirleyin. Varsayılan değer yanlış (false) olarak ayarlanır.
size dizgi	SI birimleri de içinde olmak üzere, Kubernetes' e geçmek için PersistentVolume (PersistentVolume) büyüklüğü. Yalnızca type persistent-claimise geçerlidir. Örneğin, 2Gi. Varsayılan olarak 2Gi değerine ayarlanır.
sizeLimit dizgi	ephemeral birimi kullanırken boyut sınırı. Dosyalar hala geçici bir dizine yazılıyor, bu nedenle boyutu sınırlandırmak için bu seçeneği kullanabilirsiniz. Yalnızca type ephemeralise geçerlidir.
type dizgi	Kullanılacak birim tipi. Kalıcı olmayan depolama alanını kullanmak için ephemeral , kalıcı bir birim kullanmak için persistent-claim seçeneğini belirleyin. Varsayılan olarak persistent-claimdeğerine ayarlanır.

.spec.queueManager.storage.queueManager

/var/mqmtındaki normalde herhangi bir veri için varsayılan PersistentVolume . Başka bir birim belirtilmediyse, tüm kalıcı verileri ve kurtarma günlüklerini içerir.

Burada görünür:

- [“.spec.queueManager.storage” sayfa 135](#)

Alan	Tanım
class dizgi	Bu birim için kullanılacak depolama sınıfı. Yalnızca type persistent-claimise geçerlidir. type of availability SingleInstance ya da NativeHAise, depolama sınıfı ReadWrite(Okuma Yazma) ya da ReadWrite(Okuma Yazma) türlerinden biri olabilir. type of availability MultiInstanceise, depolama sınıfı çok sayıda ReadWritetipinde olmalıdır.
deleteClaim boole	Kuyruk yöneticisi silindiğinde bu birimin silinip silinmeyeceğini belirleyin.
size dizgi	SI birimleri de içinde olmak üzere, Kubernetes' e geçmek için PersistentVolume (PersistentVolume) büyüklüğü. Yalnızca type persistent-claimise geçerlidir. Örneğin, 2Gi. Varsayılan olarak 2Gi değerine ayarlanır.
sizeLimit dizgi	ephemeral birimi kullanırken boyut sınırı. Dosyalar hala geçici bir dizine yazılıyor, bu nedenle boyutu sınırlandırmak için bu seçeneği kullanabilirsiniz. Yalnızca type ephemeralise geçerlidir.

Alan	Tanım
type dizgi	Kullanılacak birim tipi. Kalıcı olmayan depolama alanını kullanmak için ephemeral , kalıcı bir birim kullanmak için persistent-claim seçeneğini belirleyin. Varsayılan olarak persistent-claimdeğerine ayarlanır.

.spec.queueManager.storage.recoveryLogs

MQ kurtarma günlüklerine ilişkin kalıcı birim ayrıntıları. Çok eşgörünümli kuyruk yöneticisi kullanılırken gereklidir.

Burada görünür:

- [“.spec.queueManager.storage” sayfa 135](#)

Alan	Tanım
class dizgi	Bu birim için kullanılacak depolama sınıfı. Yalnızca type persistent-claimise geçerlidir. type of availability SingleInstance ya da NativeHAise, depolama sınıfı ReadWrite(Okuma Yazma) ya da ReadWrite(Okuma Yazma) türlerinden biri olabilir. type of availability MultiInstanceise, depolama sınıfı çok sayıda ReadWritetipinde olmalıdır.
deleteClaim boole	Kuyruk yöneticisi silindiğinde bu birimin silinip silinmeyeceğini belirleyin.
enabled boole	Bu birimin ayrı bir birim olarak etkinleştirilip etkinleştirilmeyeceğini ya da varsayılan queueManager birimine yerleştirilip yerleştirilmeyeceğini belirleyin. Varsayılan değer yanlış (false) olarak ayarlanır.
size dizgi	SI birimleri de içinde olmak üzere, Kubernetes' e geçmek için PersistentVolume (PersistentVolume) büyüklüğü. Yalnızca type persistent-claimise geçerlidir. Örneğin, 2Gi. Varsayılan olarak 2Gideğerine ayarlanır.
sizeLimit dizgi	ephemeral birimi kullanırken boyut sınırı. Dosyalar hala geçici bir dizine yazılıyor, bu nedenle boyutu sınırlandırmak için bu seçeneği kullanabilirsiniz. Yalnızca type ephemeralise geçerlidir.
type dizgi	Kullanılacak birim tipi. Kalıcı olmayan depolama alanını kullanmak için ephemeral , kalıcı bir birim kullanmak için persistent-claim seçeneğini belirleyin. Varsayılan olarak persistent-claimdeğerine ayarlanır.

.spec.securityContext

Kuyruk Yöneticisi Pod 'sına eklenecek güvenlik ayarları securityContext.

Burada görünür:

- [“.spec” sayfa 126](#)

Alan	Tanım
fsGroup tamsayı	Bir kaptaki tüm kaplar için geçerli olan özel bir tamamlayıcı grup. Bazı birim tipleri, Kubelet 'in bu birimin sahipliğini kapsül sahibi olacak şekilde değiştirmesine olanak tanır: 1. İye GID, FSGroup 2 olarak kabul edilebilir. setgid biti ayarlanır (birimde oluşturulan yeni dosyalar FSGroup 'a aittir) 3. İzin bitleri ya da rw-rw ile 'd' dir ---- Eğer ayarlanmazsa, Kubelet herhangi bir birimin sahipliğini ve izinlerini değiştirmeyecektir.

Alan	Tanım
initVolumeAsRoot boole	Bu, PersistentVolume'ı kullanıma hazırlayan taşıyıcı tarafından kullanılan securityContext ' ı etkiler. Yeni yetkilendirilen birimlere erişmek için kök kullanıcı olmanız gereken bir depolama sağlayıcısı kullanıyorsanız, bu ayarı true olarak ayarlayın. Bunu true olarak ayarlamak, hangi Güvenlik Bağlamı Kısıtlamaları (SCC) nesnesini kullanabileceğini etkiler ve kök kullanıcıya izin veren SCC ' yi kullanma yetkiniz yoksa, Kuyruk Yöneticisi başlatılamayabilir. Varsayılan değer yanlış (false) olarak ayarlanır. Daha fazla bilgi için bkz. https://docs.openshift.com/container-platform/latest/authentication/managing-security-context-constraints.html .
supplementalGroups dizi	Taşıyıcının birincil GID ' ye ek olarak, her taşıyıcıda çalıştırılan ilk işlem için uygulanan grupların listesi. Belirtilmemişse, herhangi bir taşıyıcıya grup eklenmez.

.spec.template

Kubernetes kaynakları için gelişmiş şablon oluşturma. The template allows users to override how IBM MQ generates the underlying Kubernetes resources, such as StatefulSet, Pods and Services. Bu, yalnızca ileri düzey kullanıcılar içindir; yanlış kullanıldığında MQ ' nun olağan çalışmasını kesintiye uğratabilecek bir potansiyeldir. QueueManager kaynağındaki başka bir yerde belirlenen değerler, şablondaki ayarlarla geçersiz kılınır.

Burada görünür:

- [“.spec” sayfa 126](#)

Alan	Tanım
pod	Pod için kullanılan şablona ilişkin geçersiz kılmalar. Bkz. https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.17/#podspec-v1-core .

.spec.tracing

Cloud Pak for Integration Operations Dashboard ile bütünleştirme bütünleştirmesine ilişkin ayarlar.

Burada görünür:

- [“.spec” sayfa 126](#)

Alan	Tanım
agent TracingAgent	Yalnızca Cloud Pak for Integration ' da isteğe bağlı İzleme Aracısı ayarlarını yapılandırabilirsiniz.
collector TracingCollector	Yalnızca Cloud Pak for Integration içinde isteğe bağlı Tracing Collector ile ilgili ayarları yapılandırabilirsiniz.
enabled boole	İzleme aracılığıyla Cloud Pak for Integration Operations Dashboard ile bütünleştirmenin etkinleştirilip etkinleştirilmeyeceğini belirleyin. Varsayılan değer yanlış (false) olarak ayarlanır.
namespace dizgi	Cloud Pak for Integration Operations Dashboard olanağının kurulu olduğu ad alanı.

.spec.tracing.agent

Yalnızca Cloud Pak for Integration ' da isteğe bağlı İzleme Aracısı ayarlarını yapılandırabilirsiniz.

Burada görünür:

- [“.spec.tracing” sayfa 138](#)

Alan	Tanım
image dizgi	Kullanılacak taşıyıcı görüntüsü.
imagePullPolicy dizgi	kubelet belirtilen görüntüyü çekmeye çalıştığında bu kontrollerin ayarlanması. Varsayılan olarak IfNotPresent değerine ayarlanır.
livenessProbe TracingProbe	Hızlılık sondasını denetleyen ayarlar.
readinessProbe TracingProbe	Hazır olma durumunu denetleyen ayarlar.

.spec.tracing.agent.livenessProbe

Hızlılık sondasını denetleyen ayarlar.

Burada görünür:

- [“.spec.tracing.agent” sayfa 138](#)

Alan	Tanım
failureThreshold tamsayı	Araştırmacı başarılı olduktan sonra başarısız olarak değerlendirilecek art arda başarısızlıklar alt sınırı. Varsayılan değer 1 'e ayarlanır.
initialDelaySeconds tamsayı	Hızlılık sondaları başlatılmadan önce başlatıldıktan sonra geçecek süre (saniye). Varsayılan değer 10 saniyedir. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .
periodSeconds tamsayı	Sondayı (saniye cinsinden) ne sıklıkta (saniye cinsinden) gerçekleştirmek için? Varsayılan değer 10 saniyedir.
successThreshold tamsayı	Sondanın başarısız olduktan sonra başarılı sayılması için arka arkaya en az başarı elde edilmesi gerekir. Varsayılan değer 1 'e ayarlanır.
timeoutSeconds tamsayı	Araştırmacı zamanaşımına geçtikten sonra geçecek süre (saniye). Varsayılan değer 2 saniye olarak ayarlanır. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .

.spec.tracing.agent.readinessProbe

Hazır olma durumunu denetleyen ayarlar.

Burada görünür:

- [“.spec.tracing.agent” sayfa 138](#)

Alan	Tanım
failureThreshold tamsayı	Araştırmacı başarılı olduktan sonra başarısız olarak değerlendirilecek art arda başarısızlıklar alt sınırı. Varsayılan değer 1 'e ayarlanır.
initialDelaySeconds tamsayı	Hızlılık sondaları başlatılmadan önce başlatıldıktan sonra geçecek süre (saniye). Varsayılan değer 10 saniyedir. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .
periodSeconds tamsayı	Sondayı (saniye cinsinden) ne sıklıkta (saniye cinsinden) gerçekleştirmek için? Varsayılan değer 10 saniyedir.
successThreshold tamsayı	Sondanın başarısız olduktan sonra başarılı sayılması için arka arkaya en az başarı elde edilmesi gerekir. Varsayılan değer 1 'e ayarlanır.

Alan	Tanım
timeoutSeconds tamsayı	Araştırmacı zamanaşımına geçtikten sonra geçecek süre (saniye). Varsayılan değer 2 saniye olarak ayarlanır. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .

.spec.tracing.collector

Yalnızca Cloud Pak for Integration içinde isteğe bağlı Tracing Collector ile ilgili ayarları yapılandırabilirsiniz.

Burada görünür:

- [“.spec.tracing” sayfa 138](#)

Alan	Tanım
image dizgi	Kullanılacak taşıyıcı görüntüsü.
imagePullPolicy dizgi	kubelet belirtilen görüntüyü çekmeye çalıştığında bu kontrollerin ayarlanması. Varsayılan olarak IfNotPresent değerine ayarlanır.
livenessProbe TracingProbe	Hızlılık sondasını denetleyen ayarlar.
readinessProbe TracingProbe	Hazır olma durumunu denetleyen ayarlar.

.spec.tracing.collector.livenessProbe

Hızlılık sondasını denetleyen ayarlar.

Burada görünür:

- [“.spec.tracing.collector” sayfa 140](#)

Alan	Tanım
failureThreshold tamsayı	Araştırmacı başarılı olduktan sonra başarısız olarak değerlendirilecek art arda başarısızlıklar alt sınırı. Varsayılan değer 1 'e ayarlanır.
initialDelaySeconds tamsayı	Hızlılık sondaları başlatılmadan önce başlatıldıktan sonra geçecek süre (saniye). Varsayılan değer 10 saniyedir. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .
periodSeconds tamsayı	Sondayı (saniye cinsinden) ne sıklıkta (saniye cinsinden) gerçekleştirmek için? Varsayılan değer 10 saniyedir.
successThreshold tamsayı	Sondanın başarısız olduktan sonra başarılı sayılması için arka arkaya en az başarı elde edilmesi gerekir. Varsayılan değer 1 'e ayarlanır.
timeoutSeconds tamsayı	Araştırmacı zamanaşımına geçtikten sonra geçecek süre (saniye). Varsayılan değer 2 saniye olarak ayarlanır. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .

.spec.tracing.collector.readinessProbe

Hazır olma durumunu denetleyen ayarlar.

Burada görünür:

- [“.spec.tracing.collector” sayfa 140](#)

Alan	Tanım
failureThreshold tamsayı	Araştırmacı başarılı olduktan sonra başarısız olarak değerlendirilecek art arda başarısızlıklar alt sınırı. Varsayılan değer 1 'e ayarlanır.
initialDelaySeconds tamsayı	Hızlılık sondaları başlatılmadan önce başlatıldıktan sonra geçecek süre (saniye). Varsayılan değer 10 saniyedir. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .
periodSeconds tamsayı	Sondayı (saniye cinsinden) ne sıklıkta (saniye cinsinden) gerçekleştirmek için? Varsayılan değer 10 saniyedir.
successThreshold tamsayı	Sondanın başarısız olduktan sonra başarılı sayılması için arka arkaya en az başarı elde edilmesi gerekir. Varsayılan değer 1 'e ayarlanır.
timeoutSeconds tamsayı	Araştırmacı zamaşımına geçtikten sonra geçecek süre (saniye). Varsayılan değer 2 saniye olarak ayarlanır. Daha fazla bilgi: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar .

.spec.web

MQ web sunucusuna ilişkin ayarlar.

Burada görünür:

- [“.spec” sayfa 126](#)

Alan	Tanım
enabled boole	Web sunucusunun etkinleştirilip etkinleştirilmeyeceği. Varsayılan değer yanlış (false) olarak ayarlanır.

.status

The observed state of the QueueManager.

Burada görünür:

- [“QueueManager” sayfa 126](#)

Alan	Tanım
adminUiUrl dizgi	Yönetici UI 'si için URL.
availability Kullanılabilirlik	Kuyruk Yöneticisi için uygunluk durumu.
conditions QueueManagerStatusConditio n dizisi	Koşullar, Kuyruk Yöneticisi 'nin durumuna ilişkin en son kullanılabilir gözlemlenmeleri gösterir.
endpoints QueueManagerStatusEndpoin t dizisi	Bu Kuyruk Yöneticinin gösterileceği uç noktalarla ilgili bilgiler (API ya da UI uç noktaları gibi).
name dizgi	Kuyruk Yöneticisi 'nin adı.
phase dizgi	Kuyruk Yöneticisi 'nin durumu aşaması.
versions QueueManagerStatusVersion	Kullanılmakta olan MQ sürümü ve IBM Yetkili Kayıt Defterinden sağlanan diğer sürümler.

.status.availability

Kuyruk Yöneticisi için uygunluk durumu.

Burada görünür:

- [“.status” sayfa 141](#)

Alan	Tanım
initialQuorumEstablished boole	NativeHA için bir ilk yetersayı oluşturulmuş olup olmadığı.

.status.conditions

QueueManagerStatusCondition , Kuyruk Yöneticisinin koşullarını tanımlar.

Burada görünür:

- [“.status” sayfa 141](#)

Alan	Tanım
lastTransitionTime dizgi	Koşul bir durumdan diğerine geçişte son kez.
message dizgi	Son geçişle ilgili ayrıntıları belirten insan tarafından okunabilir ileti.
reason dizgi	Bu durumun son geçişinin nedeni.
status dizgi	Koşulun durumu.
type dizgi	Koşul tipi.

.status.endpoints

QueueManagerStatusEndpoint , QueueManager' a ilişkin uç noktaları tanımlar.

Burada görünür:

- [“.status” sayfa 141](#)

Alan	Tanım
name dizgi	Uç noktenin adı.
type dizgi	Bir UI uç noktası için 'UI' gibi, bir API uç noktası için 'API', API belgeleri için 'OpenAPI' gibi bir uç noktana ilişkin tip.
uri dizgi	Uç noktana ilişkin URI.

.status.versions

Kullanılmakta olan MQ sürümü ve IBM Yetkili Kayıt Defterinden sağlanan diğer sürümler.

Burada görünür:

- [“.status” sayfa 141](#)

Alan	Tanım
available QueueManagerStatusVersionKullanılabilir	Diğer MQ sürümleri IBM Yetkili Kayıt Defterinden kullanılabilir.
reconciled dizgi	Kullanılmakta olan IBM MQ sürümü. Özel bir resim belirtilirse, bu, gerçekte kullanılmakta olan MQ sürümüyle eşleşmeyebilir.

.status.versions.available

Diğer MQ sürümleri IBM Yetkili Kayıt Defterinden kullanılabilir.

Burada görünür:

- [“.status.versions” sayfa 142](#)

Alan	Tanım
channels dizi	MQ sürümünü otomatik olarak güncelleştirmek için kullanılacak kanallar.
versions Sürümler dizisi	MQ ' ın belirli sürümleri kullanılabilir.

.status.versions.available.versions

QueueManagerStatusVersion , MQ' un bir sürümünü tanımlar.

Burada görünür:

- [“.status.versions.available” sayfa 143](#)

Alan	Tanım
name dizgi	QueueManager' ın bu sürümüne ilişkin sürüm name . Bunlar, spec . version alanı için geçerli değerlerdir.

Status conditions for QueueManager (mq.ibm.com/v1beta1)

status.conditions alanları, QueueManager kaynağının koşulunu yansıtacak şekilde güncellenir.

Genel olarak, koşullar anormal durumları açıklar. Sağlıklı ve hazır durumdaki bir kuyruk yöneticisi **Error** ya da **Pending** koşuluna sahip değil. Bazı danışmanlık **Warning** koşulları olabilir.

Koşullar için destek IBM MQ Operator 1.2 içinde kullanıma sunulmuştur.

Bir QueueManager kaynağı için aşağıdaki koşullar tanımlanır:

Çizelge 1. Kuyruk yöneticisi durum koşulları

Bileşen	Koşul Tipi	Neden kodu	İleti uyarısı
QueueManager ⁷	Bekliyor	Oluşturma	MQ kuyruk yöneticisi konuşlandırılıyor
	Bekliyor	OidcPending	MQ kuyruk yöneticisi OIDC istemci kaydını bekliyor
	Hata	Başarısız	MQ kuyruk yöneticisi konuşlandırılmadı
	Uyarı	UnsupportedVersion	⁸ Bir işleç, OCP sürüm < ocp_version> ¹ de desteklenmeyen bir işleç tarafından kuruldu. Bu işlenen desteklenmiyor.
	Uyarı	EUSSupport	⁹ EUS işleneni < mq_version> kuruldu, ancak genişletilmiş destek süresi için uygun olmayan bir işleç tarafından yönetiliyor. Bu işlenen, genişletilmiş destek süresi için uygun değil.
	Uyarı	EUSSupport	¹⁰ EUS işleneni < mq_version> kuruldu, ancak OCP sürüm < ocp_version> , genişletilmiş destek süresi için uygun değil. Bu işlenen, genişletilmiş destek süresi için uygun değil.
Uyarı	EUSSupport	¹¹ EUS işleneni < mq_version> kuruldu, ancak OCP sürümü < ocp_version> genişletilmiş destek süresi için uygun değil. Bu işlenen, olağan bir CD yayınına göre desteklenir.	

⁷ The conditions Creating and Failed monitor the overall progress of the deployment of the queue manager. Bir IBM Cloud Pak for Integration lisansı ve MQ Web konsolu kullanıyorsanız, OidcPending koşulu, OIDC istemci kaydının IAM ile tamamlanmak üzere beklerken kuyruk yöneticisinin durumunu günlüğe kaydeder.

⁸ Operator 1.4.0 and later

⁹ Operator 1.4.0 and later

¹⁰ Operator 1.4.0 and later

¹¹ Yalnızca 1.3.0 işleci

Çizelge 1. Kuyruk yöneticisi durum koşulları (devamı var)

Bileşen	Koşul Tipi	Neden kodu	İleti uyarısı
Bölme ¹²	Bekliyor	PodPending	MQ kuyruk yöneticisi için pod dağıtılıyor
	Hata	PodFailed	MQ kuyruk yöneticisi için pod dağıtılıyor
Depolama ¹³	Bekliyor	StoragePending	MQ kuyruk yöneticisine ilişkin saklama alanı sağlanıyor
	Uyarı	StorageEphemeral	Üretim MQ kuyruk yöneticisi için ephemeral depolama alanı kullanma
	Hata	StorageFailed	MQ kuyruk yöneticisi için saklama alanı sağlanamadı

Multi Kendi IBM MQ taşıyıcınız ve konuşlandırma kodunuzu oluşturma

Kendi kendine yapılı bir taşıyıcı geliştirin. Bu, en esnek taşıyıcı çözümdür, ancak kapsayıcıların yapılandırılmasına ilişkin güçlü becerilere ve sonuçta elde edilen kapsayıcıya "Kendi" ' e gereksinim duymanızı gerektirir.

Başlamadan önce

Kendi kapsayıcınız geliştirmeden önce, bunun yerine IBM tarafından sağlanan önceden paketlenmiş kaplardan birini kullanıp kullanamayacağınızı göz önünde bulundurun. Bkz. [IBM MQ kapsayıcılarında](#)

Bu görev hakkında

When you package IBM MQ as a container image, changes to your application can be deployed to test and staging systems quickly and easily. Bu, işletmeniz içinde sürekli teslimat için önemli bir avantaj olabilir.

Yordam

- [“Bir taşıyıcıyı kullanarak kendi IBM MQ kuyruk yöneticisi görüntüsünün planlanması” sayfa 146](#)
- [“Örnek bir IBM MQ kuyruk yöneticisi taşıyıcı görüntüsü oluşturulması” sayfa 146](#)
- [“Yerel bağ tanımlama uygulamalarının ayrı taşıyıcılarda çalıştırılması” sayfa 149](#)

İlgili kavramlar

[Taşıyıcılarda IBM MQ](#)

¹² Pod koşulları, bir kuyruk yöneticisinin konuşlandırılırken kapsüllerin durumunu izler. Herhangi bir PodFailed koşulu görüyorsanız, genel kuyruk yöneticisi koşulu da Failed olarak ayarlanır.

¹³ Depolama koşulları, kalıcı depolama için birimler oluşturmak için isteklerin ilerleyişini (StoragePending koşulu) izler ve bağlama hatalarını ve diğer hataları geri bildirir. If any error occurs during storage provisioning, the StorageFailed condition will be added to the conditions list, and the overall queue manager condition will also be set to Failed.

Bir taşıyıcıyı kullanarak kendi IBM MQ kuyruk yöneticisi görüntüsünün planlanması

Bir taşıyıcıda bir IBM MQ kuyruk yöneticisi çalıştırırken dikkate alınması gereken bazı gereksinimler vardır. Örnek taşıyıcı görüntüsü, bu gereksinimleri ele almak için bir yol sağlar; ancak, kendi görüntünüzü kullanmak istiyorsanız, bu gereksinimlerin nasıl işlendiğini göz önünde bulundurmanız gerekir.

Süreç denetimi

Bir taşıyıcıyı çalıştırdığınızda, temel olarak, daha sonra alt süreçleri yumurtlayabilen tek bir süreç (taşıyıcının içinde 1 PID) çalıştırıyorsunuz demektir.

Ana işlem sona ererse, taşıyıcı yürütme ortamı taşıyıcıyı durdurur. Bir IBM MQ kuyruk yöneticisi, arka planda birden çok işlem çalıştırılmasını gerektirir.

Bu nedenle, kuyruk yöneticisi çalışır durumda olduğundan ana işleminizin etkin kalmasını sağlamak gerekir. Örneğin, kuyruk yöneticisinin bu süreçten (örneğin, denetim sorguları gerçekleştirerek) etkin olup olmadığını kontrol etmek iyi bir uygulamadır.

/var/mqmveri yerleştiriliyor

Kapsayıcılar, bir birim olarak /var/mqm ile yapılandırılmalıdır.

Bunu yaparken, taşıyıcı ilk başlatıldığında birimin dizini boş olur. Bu izin genellikle kuruluş sırasında doldurulur, ancak kuruluş ve yürütme ortamı bir taşıyıcı kullanılırken ayrı ortamlardır.

To solve this, when your container starts, you can use the `crtmqdir` command to populate /var/mqm when it runs for the first time.

Konteyner güvenliği

Yürütme ortamı güvenlik gereksinimlerini en aza indirmek için, örnek taşıyıcı görüntüleri IBM MQ tarafından kaldırılabilir kuruluş kullanılarak kurulur. Bu, hiçbir setuid bitinin ayarlanmamasını ve taşıyıcının ayrıcalık yükseltme olanağını kullanmayabilmesini sağlar. Bazı kapsayıcı sistemler, hangi kullanıcı kimliklerinin kullanabileceğini tanımlar ve sıkıştırılmaz kuruluş, kullanılabilir işletim sistemi kullanıcıları hakkında herhangi bir varsayımda bulunmaz.

Örnek bir IBM MQ kuyruk yöneticisi taşıyıcı görüntüsü oluşturulması

Bir taşıyıcıda IBM MQ kuyruk yöneticisini çalıştırmak için örnek bir taşıyıcı görüntüsü oluşturmak için bu bilgileri kullanın.

Bu görev hakkında

Firstly, you build a base image containing an Red Hat Universal Base Image file system and a clean installation of IBM MQ.

İkinci olarak, temel kullanıcı kimliği ve parola güvenliği sağlamak için bazı IBM MQ yapılandırması ekleyen temel tabanın üzerine başka bir taşıyıcı görüntü katmanı oluşturursunuz.

Son olarak, bu görüntüyü dosya sistemi olarak kullanarak, anasistem dosya sistemindeki bir taşıyıcıya özgü bir birim tarafından sağlanan /var/mqm içeriğiyle birlikte bir kapsayıcı çalıştırıyorsunuz.

Yordam

- Bir taşıyıcıda IBM MQ kuyruk yöneticisini çalıştırmak için örnek bir taşıyıcı görüntüsünün oluşturulmasına ilişkin bilgi edinmek için aşağıdaki alt konulara bakın:
 - [“Örnek temel IBM MQ kuyruk yöneticisi görüntüsü oluşturulması” sayfa 147](#)

Multi Örnek temel IBM MQ kuyruk yöneticisi görüntüsü oluşturulması

Kendi kap resminizde IBM MQ ' yi kullanmak için, öncelikle temiz bir IBM MQ kuruluşuna sahip bir temel görüntü oluşturmada gerekir. Aşağıdaki adımlar, GitHub' da barındırılan örnek kodu kullanarak örnek temel bir resmin nasıl oluşturulacağı gösterilmektedir.

Yordam

- Üretim kabınız görüntüsünü oluşturmak için [mq-container GitHub havuzu](#) ' ta sağlanan make dosyalarını kullanın.

GitHub' da [Taşıyıcı görüntüsü oluşturma](#) konusundaki yönergeleri izleyin. If you plan to configure secure access using the Red Hat OpenShift Container Platform "sınırlı" Security Context Constraint (SCC), you must use the 'No-Install' IBM MQ package.

Sonuçlar

Şimdi IBM MQ kurulu bir temel taşıyıcı görüntüsünüz var.

Artık [build a sample configured IBM MQ queue manager image](#)' a hazırsınız.

Multi Örnek yapılandırılmış bir IBM MQ kuyruk yöneticisi görüntüsü oluşturulması

Soyal temel IBM MQ taşıyıcı görüntüsünü oluşturduktan sonra, güvenli erişime izin vermek için kendi yapılandırmanızı uygulamanız gerekir. Bunu yapmak için, genel görüntüyü bir üst öge olarak kullanarak kendi taşıyıcı görüntü katmanınızı oluşturursunuz.

Başlamadan önce

V 9.2.0 This task assumes that, when you [örnek temel IBM MQ kuyruk yöneticisi görüntüsünü oluşturma](#), you used the "Hayır-Kuruluş" IBM MQ package. Tersine durumda, Red Hat OpenShift Container Platform "restricted" Security Constraint Constraint (SCC) olanağını kullanarak güvenli erişimi yapılandıramazsınız. The "sınırlı" SCC, which is used by default, uses random user IDs, and prevents privilege escalation by changing to a different user. IBM MQ geleneksel RPM tabanlı kuruluş programı bir mqm kullanıcı ve grubuna dayanır ve yürütülebilir programlarda setuid bitlerini kullanır. IBM MQ 9.2' ta, "Hayır-Kuruluş" IBM MQ paketini kullandığınızda, başka bir mqm kullanıcısı ya da bir mqm grubu yoktur.

Yordam

1. Yeni bir dizin yaratın ve aşağıdaki içerikle birlikte `config.mqsc` adlı bir dosya ekleyin:

```
DEFINE QLOCAL(EXAMPLE.QUEUE.1) REPLACE
```

Önceki örneğe basit kullanıcı kimliği ve parola doğrulaması kullandığına dikkat edin. Ancak, teşebbüsünüzün gerektirdiği herhangi bir güvenlik yapılandırmasını uygulayabilirsiniz.

2. Aşağıdaki içerikle birlikte, `Dockerfile` adlı bir dosya oluşturun:

```
FROM mq
COPY config.mqsc /etc/mqm/
```

3. Aşağıdaki komutu kullanarak özel taşıyıcı görüntünüzü oluşturun:

```
docker build -t mymq .
```

Burada ". ", az önce yarattığınız iki dosyayı içeren dizindir.

Docker daha sonra bu görüntüyü kullanarak geçici bir taşıyıcı yaratır ve geri kalan komutları çalıştırır.

Not: Red Hat Enterprise Linux (RHEL) üzerinde, **docker** (RHEL V7) ya da **podman** (RHEL V7 ya da RHEL V8) komutunu kullanıyorsunuz. On Linux, you will need to run **docker** commands with **sudo** at the beginning of the command, to gain extra privileges.

4. Yeni oluşturduğunuz disk görüntüyle yeni bir kapsayıcı oluşturmak için yeni özelleştirilmiş görüntüyü çalıştırın.

Yeni görüntü katmanınız, çalıştırılacak belirli bir komutu belirtmemiş, bu nedenle üst görüntüden devralınmıştır. Üst öğenin giriş noktası (kod GitHub' ta kullanılabilir):

- Kuyruk yöneticisi yaratır
- Kuyruk yöneticisini başlatır
- Varsayılan bir dinleyici oluşturur
- Bundan sonra, /etc/mqm/config.mqsc.' den herhangi bir MQSC komutu çalıştırır

Yeni uyarlanmış görüntüyü çalıştırmak için aşağıdaki komutları verin:

```
docker run \
  --env LICENSE=accept \
  --env MQ_QMGR_NAME=QM1 \
  --volume /var/example:/var/mqm \
  --publish 1414:1414 \
  --detach \
  mymq
```

Burada:

İlk env parametresi

Bir ortam değişkenini kapsayıcıya geçirir. Bu, IBM IBM WebSphere MQ için lisansı kabul ettiğinizi kabul eder. Lisans değişkenini, lisansı görüntülemek için görüntülemek üzere de ayarlayabilirsiniz.

IBM MQ lisanslarına ilişkin ek ayrıntılar için [IBM MQ lisans bilgileri](#) başlıklı konuya bakın.

İkinci env parametresi

Kullanmakta olduğunuz kuyruk yöneticisi adını belirler.

Birim parametresi

Tells the container that whatever MQ writes to /var/mqm should actually be written to /var/example on the host.

Bu seçenek, taşıyıcıyı daha sonra kolayca silebileceğiniz ve kalıcı verileri sakladığınız anlamına gelir. Bu seçenek, günlük dosyalarının görüntülenmesini kolaylaştırır.

Değiştirge yayınla

Anasistemdeki kapıları, taşıyıcıdaki kapılarla eşler. Taşıyıcı, varsayılan değer olarak kendi iç IP adresiyle çalıştırılır; bu da, göstermek istediğiniz herhangi bir kapıyı özellikle eşlememeniz gerektiğini gösterir.

Bu örnekte, bu, anasistemdeki 1414 numaralı kapının taşıyıcıdaki 1414 numaralı bağlantı noktasına eşlenmesini anlamına gelir.

Parametreyi ayır

Taşıyıcıyı arka plandaki çalıştırır.

Sonuçlar

Yapılandırılmış bir taşıyıcı görüntüsü oluşturdu ve **docker ps** komutunu kullanarak çalışan taşıyıcıları görüntüleyebiliyorsunuz. You can view the IBM MQ processes running in your container using the **docker top** command.



Uyarı:

docker logs \${CONTAINER_ID} komutunu kullanarak bir taşıyıcının günlüklerini görüntüleyebilirsiniz.

Sonraki adım

- **docker ps** komutunu kullandığınızda taşıyıcınız gösterilmiyorsa, taşıyıcının başarısız olması gerekebilir. **docker ps -a** komutunu kullanarak başarısız olan taşıyıcıları görebilirsiniz.
- **docker ps -a** komutunu kullandığınızda, taşıyıcı tanıtıcısı görüntülenir. Bu tanıtıcı, **docker run** komutunu verdiğinizde de yazdırılır.
- Bir taşıyıcının günlüklerini **docker logs \${CONTAINER_ID}** komutunu kullanarak görüntüleyebilirsiniz.

Multi Yerel bağ tanımlama uygulamalarının ayrı taşıyıcılarda çalıştırılması

With process namespace sharing between containers in Docker, you can run applications that require a local binding connection to IBM MQ in separate containers from the IBM MQ queue manager.

Bu görev hakkında

Bu işlevsellik IBM MQ 9.0.3 ve sonraki kuyruk yöneticilerinde desteklenir.

Aşağıdaki kısıtlamalara uymanız gerekir:

- `--pid` bağımsız değişkenini kullanarak, taşıyıcıların PID ad alanını paylaşmalısınız.
- You must share the containers IPC namespace using the `--ipc` argument.
- Aşağıdakilerden birini yapmak gerekir:
 1. Taşıyıcıların UTS ad alanını `--uts` bağımsız değişkenini kullanarak anasistemle paylaşır ya da
 2. Taşıyıcıların, `-h` ya da `--hostname` bağımsız değişkenini kullanarak aynı anasistem adına sahip olduğundan emin olun.
- IBM MQ veri dizinini, `/var/mqm` dizini altındaki tüm kapsayıcılar için kullanılabilir bir birime bağlamanız gerekir.

You can try this functionality out, by completing the following steps on a Linux system that already has Docker installed.

Aşağıdaki örnek, örnek IBM MQ kapsayıcı görüntüsünü kullanır. Bu resmin ayrıntılarını [Github](#)'te bulabilirsiniz.

Yordam

1. Aşağıdaki komutu girerek, biriminiz olarak işlev görmeniz için geçici bir dizin oluşturun:

```
mkdir /tmp/dockerVolume
```

2. Create a queue manager (QM1) in a container, with the name `sharedNamespace`, by issuing the following command:

```
docker run -d -e LICENSE=accept -e MQ_QMGR_NAME=QM1 --volume /tmp/dockerVol:/mnt/mqm --uts host --name sharedNamespace ibmcom/mq
```

3. Start a second container called `secondaryContainer`, based off `ibmcom/mq`, but do not create a queue manager, by issuing the following command:

```
docker run --entrypoint /bin/bash --volumes-from sharedNamespace --pid container:sharedNamespace --ipc container:sharedNamespace --uts host --name secondaryContainer -it --detach ibmcom/mq
```

4. Her iki kuyruk yöneticisinin durumunu görmek için, ikinci taşıyıcıda **dspsmq** komutunu çalıştırın ve aşağıdaki komutu verin:

```
docker exec secondaryContainer dspsmq
```

5. Diğer taşıyıcıda çalışan kuyruk yöneticisine yönelik MQSC komutlarını işlemek için aşağıdaki komutu çalıştırın:

```
docker exec -it secondaryContainer runmqsc QM1
```

Sonuçlar

You now have local applications running in separate containers, and you can now successfully run commands like **dspmq**, **amqspmt**, **amqsget**, and **runmqsc** as local bindings to the QM1 queue manager from the secondary container.

Beklediğiniz sonucu görmezseniz, daha fazla bilgi için bkz. [“Ad alanı uygulamalarınızda sorun giderme” sayfa 150](#).

Multi Ad alanı uygulamalarınızda sorun giderme

Paylaşılan ad alanlarını kullanırken, tüm ad alanlarını (IPC, PID ve UTS/hostname) ve bağlı birimleri paylaştığınızdan emin olmanız gerekir, aksi takdirde uygulamalarınız çalışmaz.

İzlemeniz gereken kısıtlamaların listesi için bkz. [“Yerel bağ tanımlama uygulamalarının ayrı taşıyıcılarda çalıştırılması” sayfa 149](#).

Uygulamanız listelenen tüm kısıtlamaları karşılamıyorsa, taşıyıcının başladığı yerde sorunlarla karşılaşabilirsiniz, ancak beklediğiniz işlev çalışmaz.

Aşağıdaki listede bazı sık rastlanan nedenler ve kısıtlamalardan birini karşılamayı unutup unutmadığınızı göremediğiniz davranışlar yer alır.

- Ad alanını (UTS/PID/IPC) ya da taşıyıcıların anasistem adını paylaşmayı unutursanız ve birimi sisteme bağlarsanız, taşıyıcınız kuyruk yöneticisini görebilir, ancak kuyruk yöneticisiyle etkileşimde bulunamaz.
 - **dspmq** komutları için aşağıdakileri görürsünüz:

```
docker exec container dspmq
QMNAME(QM1)                STATUS(Status not available)
```

- **runmqsc** komutları ya da kuyruk yöneticisine bağlanmayı deneyen diğer komutlar için bir AMQ8146 hata iletisi alabilirsiniz:

```
docker exec -it container runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024.
Starting MQSC for queue manager QM1.
AMQ8146: IBM MQ queue manager not available
```

- Gerekli tüm ad alanlarını paylaşır, ancak `/var/mqm` dizinine bir paylaşılan birim bağlamazsanız ve geçerli bir IBM MQ veri yolunuz varsa, komutlarınız AMQ8146 hata iletilerini de alır.

Ancak **dspmq**, kuyruk yöneticinizi hiç göremez ve bunun yerine boş bir yanıt döndürür:

```
docker exec container dspmq
```

- Gerekli tüm ad alanlarını paylaşır, ancak `/var/mqm` dizinine paylaşılan bir birim bağlamazsanız ve geçerli bir IBM MQ veri yolunuz yoksa (ya da IBM MQ veri yolu yoksa), veri yolu bir IBM MQ kuruluşunun temel bileşeni olduğundan çeşitli hatalar görürsünüz. Veri yolu olmadan IBM MQ çalışmaz.

Aşağıdaki komutlardan herhangi birini çalıştırırsanız ve bu örneklerde gösterilenlere benzer yanıtlar görürseniz, dizini sisteme yerleştirdiğinizi ya da bir IBM MQ veri dizini oluşturduğunu doğrulamanız gerekir:

```
docker exec container dspmq
'No such file or directory' from /var/mqm/mqs.ini
AMQ6090: IBM MQ was unable to display an error message FFFFFFFF.
AMQffff
```

```
docker exec container dspmqver
AMQ7047: An unexpected error was encountered by a command. Reason code is 0.

docker exec container mqrc
<file path>/mqrc.c[1152]
lpiObtainQMDetails --> 545261715

docker exec container crtmqm QM1
AMQ8101: IBM MQ error (893) has occurred.

docker exec container strmqm QM1
AMQ6239: Permission denied attempting to access filesystem location '/var/mqm'.
AMQ7002: An error occurred manipulating a file.

docker exec container endmqm QM1
AMQ8101: IBM MQ error (893) has occurred.

docker exec container dlrmqm QM1
AMQ7002: An error occurred manipulating a file.

docker exec container strmqweb
<file path>/mqrc.c[1152]
lpiObtainQMDetails --> 545261715
```

CP4I Kendi taşıyıcılarınız yaratılıyorsa, Yerel HA grubunun yaratılması

Yerel HA grubunu yaratmak için üç kuyruk yöneticisi yaratmalı, yapılandırılmalı ve başlatmalısınız.

Bu görev hakkında

Yerel HA çözümü oluşturmak için önerilen yöntem, IBM MQ işlecini kullanarak (bkz. [Yerel HA](#)). Diğer bir seçenek olarak, kendi kapsayıcılarınızı yaratıyorsanız, bu yönergeleri izleyebilirsiniz.

Bir Yerel HA grubu oluşturmak için, günlük tipi log replication olarak ayarlanmış üç düğüm üzerinde üç kuyruk yöneticisi oluşturursun. Daha sonra her bir kuyruk yöneticisi için `qm.ini` dosyasını düzenleyerek, her üç düğümün her biri için günlük verilerini eşleyebilmeleri için bağlantı ayrıntılarını ekleyin.

Daha sonra, üç kuyruk yöneticisinin tüm üç eşgörünümünün birbiriyle iletişim kurabileceğini ve hangilerinin etkin eşgörünüm olacağını ve hangi eşlemeler olacağını denetleyebilmeleri için üç kuyruk yöneticisini başlatmalısınız.

Yordam

1. Üç düğümün her birinde, bir kuyruk yöneticisi yaratın ve her bir günlük eşlemesi için benzersiz bir ad belirterek bir kuyruk yöneticisi yaratın. Her kuyruk yöneticisinin aynı adı vardır:

```
crtmqm -lr instance_name qmname
```

Örneğin:

```
node 1> crtmqm -lr qm1_inst1 qm1
node 2> crtmqm -lr qm1_inst2 qm1
node 3> crtmqm -lr qm1_inst3 qm1
```

2. On successful creation of each queue manager, an additional stanza named `NativeHALocalInstance` is added to the queue manager configuration file, `qm.ini`. Sağlanan eşgörünüm adını belirterek, kısmına bir Name özneliği eklenir.

İsteğe bağlı olarak, aşağıdaki öznelikleri `qm.ini` dosyasındaki `NativeHALocalInstance` stanza 'ya ekleyebilirsiniz:

KeyRepository

Günlük eşleme trafiğinin korunması için kullanılacak dijital sertifikayı tutan anahtar havuzunun yeri. Konum kök biçiminde verilir; yani, tam yolu ve uzantısız dosya adını içerir. KeyRepository stanza özneliği atlanırsa, günlük eşleme verileri düz metindeki eşgörünüm arasında değış tokuş edilir.

CertificateLabel

Günlük eşleme trafiğinin korunması için kullanılacak sayısal sertifikayı tanımlayan sertifika etiketi. KeyRepository sağlandıysa, ancak CertificateLabel atlanırsa, varsayılan bir `ibmwebspheremqueue_manager` deęeri kullanılır.

CipherSpec

Günlük eşleme trafiğini korumak için kullanılacak MQ CipherSpec . Bu stanza özneliği sağlandıysa, KeyRepository de sağlanmalıdır. KeyRepository sağlandıysa, ancak CipherSpec atlanırsa, varsayılan bir ANY deęeri kullanılır.

LocalAddress

Günlük eşleme trafiğini kabul eden yerel ağ arabirimi adresi. Bu stanza özneliği sağlanırsa, yerel ağ arabirimini ve/veya kapıyı "[addr] [(kapı)]" biçimini kullanarak tanımlıyor. Ağ adresi, anasistem adı, IPv4 noktalı onlu ya da IPv6 onaltılı biçimi olarak belirlenebilir. Bu öznelik atlanırsa, kuyruk yöneticisi tüm ağ arabirimlerine bağlanmayı dener; bu deęer, yerel yönetim ortamı adıyla eşleşen NativeHAInstances 'ta ReplicationAddress ' da belirtilen kapıyı kullanır.

HeartbeatInterval

Saęlıklı işletim bildirim aralığı, bir Native HA kuyruk yöneticisinin etkin bir yönetim ortamının milisaniye olarak ne sıklıkta ağ sinyali gönderdiğini tanımlar. Saęlıklı işletim bildirim aralığı deęeri 500 (0.5 saniye) ile 60000 (1 dakika) arasında bir deęer, bu aralığın dışındaki bir deęer kuyruk yöneticisinin başlamasına neden olur. Bu öznelik atılırsa, varsayılan deęer 5000 (5 saniye) kullanılır. Her eşgörünüm aynı saęlıklı işletim bildirim aralığını kullanmalıdır.

HeartbeatTimeout

Saęlıklı işletim bildirim zamanasını, etkin örneęe yanıt vermemeye karar vermeden önce Yerel HA kuyruk yöneticisinin eşleme eşgörünümünün ne kadar süreyle bekleyeceğini tanımlar. Saęlıklı işletim bildirim aralığı zamanasını deęerinin geçerli aralığı 500 (0.5 saniye) ile 120000 (2 dakika) arasındadır. Saęlıklı işletim bildirim zaman aşısının deęeri, saęlıklı işletim bildirim aralığından büyük ya da ona eşit olmalıdır.

Geçersiz bir deęer, kuyruk yöneticisinin başlatılamamasına neden olur. Bu öznelik atlanırsa, yeni bir etkin yönetim ortamı seçmek için işlemi başlatmadan önce 2 x HeartbeatInterval için bekleyen bir bekleme işlemi bekler. Her eşgörünüm aynı saęlıklı işletim bildirim zamanasını kullanmalıdır.

RetryInterval

Yeniden deneme aralığı, Native HA kuyruk yöneticisinin ne sıklıkta başarısız bir eşleme bağlantısını yeniden denemesi gerektiğini milisaniye olarak tanımlar. Yeniden deneme aralığı için geçerli aralık 500 (0.5 saniye) ile 120000 (2 dakika) arasındadır. Bu öznelik atlanırsa, başarısız bir eşleme bağlantısını yeniden denemeden önce 2 x HeartbeatInterval için bekleyen bir bekleme işlemi bekler.

- Her kuyruk yöneticisi için `qm.ini` dosyasını düzenleyin ve bağlantı ayrıntılarını ekleyin. Yerel HA grubunda (yerel yönetim ortamı da içinde olmak üzere) her bir kuyruk yöneticisi yönetim ortamı için bir tane olmak üzere üç NativeHAInstance stanzası ekleyebilirsiniz. Aşağıdaki öznelikleri ekleyin:

Ad

Kuyruk yöneticisi yönetim ortamını yaratırken kullandığınız yönetim ortamı adını belirtin.

ReplicationAddress

Eşgörünümün anasistem adını, IPv4 noktalı onlu ya da IPv6 onaltılı biçim adresini belirtin. Adresi anasistem adı, IPv4 noktalı onlu ya da IPv6 onaltılı biçim adresi olarak belirleyebilirsiniz. Eşleme adresi, gruptaki her bir eşgörünümünden çözülebilir ve yönlendirilebilir olmalıdır. Günlük eşlemesi için kullanılacak kapı numarasının köşeli parantez içinde olması gerekir, örneğin:

```
ReplicationAddress=host1.example.com(4444)
```


Not: NativeHAInstance stanzaları her yönetim ortamında aynıdır ve otomatik yapılandırma (**crtmqm -ii**) kullanılarak sağlanabilir.

4. Üç eşgörünümünden her birini başlatın:

```
strmqm QMgrName
```

Yönetim ortamları başlatıldığında, tüm üç eşgörünümünün çalışmakta olduğunu denetlemek için iletişim kurar, daha sonra, diğer iki eşgörünüm eşlemeler olarak çalışmaya devam ederken, üç eşgörünümünün hangisinin etkin eşgörünüm olduğuna karar verir.

Örnek

Aşağıdaki örnekte, üç eşgörünümünden biri için gerekli Native HA ayrıntılarını belirten bir `qm.ini` dosyasının bölümü gösterilmektedir:

```
NativeHALocalInstance:
  LocalName=node-1

NativeHAInstance:
  Name=node-1
  ReplicationAddress=host1.example.com(4444)
NativeHAInstance:
  Name=node-2
  ReplicationAddress=host2.example.com(4444)
NativeHAInstance:
  Name=node-3
  ReplicationAddress=host3.example.com(4444)
```

Kubernetes Yerel HA kuyruk yöneticisine ilişkin kendi hareketli güncelleme gerçeştirirken dikkat edilecek noktalar

Yerli HA kuyruk yöneticisi için IBM MQ sürümüne ya da Pod belirtimine ilişkin herhangi bir güncelleme, kuyruk yöneticisi eşgörünümünü hareketli bir şekilde güncelleme gerektirir. IBM MQ Operator bunu sizin için otomatik olarak işler, ancak kendi konuşlandırma kodunuzu oluşturursanız, bazı önemli noktalar vardır.

Not: örnek Helm grafiği, hareketli güncelleme gerçeştirmek için bir kabuk komut dosyası içerir, ancak komut dosyası bu konudaki önemli noktaları ele almadığı için üretim kullanımı için **uygun** değildir.

Kubernetesinde StatefulSet kaynakları, sıralı başlangıç ve toplama güncellemelerini yönetmek için kullanılır. Başlangıç prosedürünün bir parçası, her bir Bölme'yi tek tek başlatmak, onun hazır olmasını beklemek ve daha sonra bir sonraki Bölme'ye geçmektir. Bu Yerli HA için işe yaramayacak, çünkü tüm Pod'ların bir lider seçimlerini yönetebilmeleri için başlatılması gerekiyor. Bu nedenle, StatefulSet üzerindeki `.spec.podManagementPolicy` alanının Parallellolarak ayarlanması gerekir. Bu aynı zamanda tüm Podların da özellikle istenmeyen paralel olarak güncelleneceği anlamına gelir. Bu nedenle StatefulSet, OnDelete güncelleme stratejisini de kullanmalıdır.

StatefulSet hareketli güncelleme kodu sürücülerini kullanmakta yetersizlik, aşağıdakileri göz önünde bulunduran özel bir hareketli güncelleme koduna gereksinim duymanızı sağlar:

- Genel hareketli güncelleme yordamı
- Bölme'ler en iyi sırada güncellenerek kapalı süreyi en aza indirme
- Küme durumundaki değişiklikler işlenecek
- Hataların işlenmesi
- Zamanlama sorunlarının işlenmesi

Genel hareketli güncelleme yordamı

Hareketli güncelleme kodu, her yönetim ortamının REPLICa from dspmqdurumunu göstermesini beklemelidir. Başka bir deyişle, yönetim ortamı bir başlatma düzeyi gerçeştirdi (örneğin, taşıyıcı başlatıldı ve MQ işlemleri çalışıyor), ancak henüz diğer yönetim ortamlarıyla konuşmayı başaramadı. Örneğin: Pod A yeniden başlatılır ve REPLICa durumunda olduğu anda Pod B yeniden başlatılır. Pod B

yeni yapılandırmayla başladıktan sonra, Pod A ile konuşabilmeli ve çekirdek oluşturabilmeli ve A veya B yeni aktif örnek haline gelecek.

Bunun bir parçası olarak, her Bölme REPLICA durumuna ulaştıktan sonra, eşlerine bağlanmasına ve yetersayı oluşturmasına izin vermek için bir gecikme olması yararlıdır.

Bölmeler en iyi sırada güncellenerek kapalı süreyi en aza indirme

Hareketli güncelleme kodu, bilinen bir hata durumundaki Pod'lardan başlayarak her defasında bir podları silmeli ve bunu başarılı bir şekilde başlanmamış herhangi bir Pod 'lar izlemelidir. Etkin kuyruk yöneticisi Pod genellikle en son güncellenmelidir.

Son güncelleme bilinen bir hata durumuna giren bir Pod ile sonuçlandıysa, Pod'ların silinmesini duraklatmak da önemlidir. Bu, tüm Podlarda bozuk bir güncellenmenin devreye alınmasını önler. Örneğin, Pod erişilebilir olmayan (ya da yazım hatası içeren) yeni bir kapsayıcı görüntüsü kullanacak şekilde güncellenirse bu oluşabilir.

Küme durumundaki değişiklikler işlenecek

Hareketli güncelleme kodunun, küme durumundaki gerçek zamanlı değişikliklere uygun şekilde yanıt vermesi gerekir. Örneğin, bir Düğüm yeniden yüklemesi ya da Düğüm basıncı nedeniyle kuyruk yöneticisinin Pod'larından biri çıkarılabilir. Küme meşgulse, tahliye edilmiş bir Pod hemen yeniden zamanlanmayabilir. Bu durumda, yuvarlanan güncelleme kodunun başka bir Pod 'u yeniden başlatmadan önce uygun şekilde beklemesi gerekir.

Hataların işlenmesi

Kubernetes API 'si ve diğer beklenmeyen küme davranışı çağrılırken oluşan güncelleme kodunun güçlü olması gerekir.

Ayrıca, hareketli güncelleme kodunun kendisinin yeniden başlatılmasına tolerans göstermesi gerekir. Hareketli bir güncelleme uzun süreli olabilir ve kodun yeniden başlatılması gerekebilir.

Zamanlama sorunlarının işlenmesi

Hareketli güncelleme kodunun, Pod 'un yeniden başlatıldığından emin olabilmesi için Pod 'un güncelleme düzeltmelerini denetlemesi gerekir. Bu, bir Pod 'un "Başlatıldı" olduğunu göstermesi durumunda zamanlama sorunlarını önler, ancak aslında henüz sonlandırılmamıştır.

İlgili kavramlar

[“Kapsayıcılarda IBM MQ 'ı nasıl kullanmak istediğinizi seçme” sayfa 5](#)

There are multiple options for using IBM MQ in containers: you can choose to use the IBM MQ Operator, which uses pre-packaged container images, or you can build your own images and deployment code.

CP4I Özel yapı taşıyıcılar için Yerel HA kuyruk yöneticilerinin durumunu görüntüleme

For custom-built containers, you can view the status of the Native HA instances by using the **dspmq** command.

Bu görev hakkında

Bir düğümdeki kuyruk yöneticisi örneğinin çalışma durumunu görüntülemek için **dspmq** komutunu kullanabilirsiniz. Döndürülen bilgiler, eşgörünümün etkin mi, yoksa eşleme mi olduğuna bağlıdır. Etkin örnek tarafından sağlanan bilgiler kesin, eşleme düğümlerinden alınan bilgiler güncel olmayabilir.

Aşağıdaki işlemleri gerçekleştirebilirsiniz:

- Geçerli düğümdeki kuyruk yöneticisi örneğinin etkin mi, yoksa eşleme mi olduğunu görüntüler.
- Yürürlükteki düğümdeki yönetim ortamının Yerel HA çalışma durumunu görüntüler.

- Yerel HA yapılandırmasındaki tüm üç örneğin çalışma durumunu görüntüleyin.

Yerel HA yapılanış durumunu raporlamak için aşağıdaki durum alanları kullanılır:

Görev

Eşgörünümün geçerli rolünü belirtir ve Active, Replica ya da Unknown' den biridir.

yönetim ortamı

The name provided for this instance of the queue manager when it was created using the **-lr** option of the **crtmqm** command.

İNSYNC

Yönetim ortamının gerektiğinde etkin yönetim ortamı olarak kabul edilip edilemeyeceğini belirtir.

ÇEKİRDEK

Yetersayı durumunu *number_of_instances_in-sync/number_of_instances_config* biçiminde raporlar.

YANITLA

Kuyruk yöneticisi örneğinin eşleme adresi.

CONNECTV

Düğümün etkin eşgörünümle bağlı olup olmadığını belirtir.

BACKLOG

Eşgörünümün arkasında olduğu KB sayısını gösterir.

CONNINST

Belirtilen yönetim ortamının bu yönetim ortamına bağlı olup olmadığını gösterir.

TARİH DEĞERİ

Bu bilgilerin en son güncellendiği tarihi gösterir (hiçbir zaman güncellenmediyse boş değer).

ALTTIME

Bu bilgilerin en son güncellendiği zamanı belirtir (hiçbir zaman güncellenmediyse boş değer).

Yordam

- Bir kuyruk yöneticisi örneğinin etkin eşgörünüm olarak mı, yoksa eşleme olarak mı çalıştırılacağını belirlemek için:

```
dspmqr -o status -m QMgrName
```

BOB adlı bir kuyruk yöneticisinin etkin bir eşgörünümü aşağıdaki durumu bildirecektir:

```
QMNAME(BOB)          STATUS(Running)
```

BOB adlı bir kuyruk yöneticisinin eşleme eşgörünümü aşağıdaki durumu bildirecektir:

```
QMNAME(BOB)          STATUS(Replica)
```

Etkin olmayan bir yönetim ortamı aşağıdaki durumu bildirecektir:

```
QMNAME(BOB)          STATUS(Ended Immediately)
```

- Yürürlükteki düğümde yönetim ortamının yerel HA işletim durumunu saptamak için:

```
dspmqr -o nativeha -m QMgrName
```

BOB adlı bir kuyruk yöneticisinin etkin örneği, aşağıdaki durumu bildirebilir:

```
QMNAME(BOB)          ROLE(Active) INSTANCE(inst1) INSYNC(Yes) QUORUM(3/3)
```

BOB adlı bir kuyruk yöneticisinin eşleme örneği aşağıdaki durumu bildirebilir:

```
QMNAME(BOB)          ROLE(Replica) INSTANCE(inst2) INSYNC(Yes) QUORUM(2/3)
```

BOB adlı bir kuyruk yöneticisinin etkin olmayan bir eşgörünümü aşağıdaki durumu bildirebilir:

```
QMNAME(BOB)                ROLE(Unknown) INSTANCE(inst3) INSYNC(no) QUORUM(0/3)
```

- Yerel HA yapılandırmasındaki tüm eşgörünümlerin Yerel HA işletim durumunu belirlemek için:

```
dspmqr -o nativeha -x -m QMgrName
```

Bu komutu, kuyruk yöneticisi BOB ' un etkin örneğini çalıştıran düğümde yayınyorsanız, aşağıdaki durumu alabilirsiniz:

```
QMNAME(BOB)                ROLE(Active) INSTANCE(inst1) INSYNC(Yes) QUORUM(3/3)
INSTANCE(inst1) ROLE(Active) REPLADDR(9.20.123.45) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
```

Bu komutu, kuyruk yöneticisi BOB ' un eşleme örneğini çalıştıran bir düğümde yayınyorsanız, eşlemelerden birinin geride kaldığını belirten aşağıdaki durumu alabilirsiniz:

```
QMNAME(BOB)                ROLE(Replica) INSTANCE(inst2) INSYNC(Yes) QUORUM(2/3)
INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst1) ROLE(Active) REPLADDR(9.20.123.45) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(No) BACKLOG(435)
CONNINST(Yes) ALTDATA(2022-01-12) ALTTIME(12.03.44)
```

Bu komutu, kuyruk yöneticisi BOB ' un etkin olmayan bir örneğini çalıştıran bir düğümde yayınyorsanız, aşağıdaki durumu alabilirsiniz:

```
QMNAME(BOB)                ROLE(Unknown) INSTANCE(inst3) INSYNC(no) QUORUM(0/3)
INSTANCE(inst1) ROLE(Unknown) REPLADDR(9.20.123.45) CONNACTV(Unknown) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATA() ALTTIME()
INSTANCE(inst2) ROLE(Unknown) REPLADDR(9.20.123.46) CONNACTV(Unknown) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATA() ALTTIME()
INSTANCE(inst3) ROLE(Unknown) REPLADDR(9.20.123.47) CONNACTV(No) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATA() ALTTIME()
```

Yönetim ortamları hala etkin olan ve eşlemeler olan bir yönetim ortamı üzerinde çalışmaya devam ederken komutu yayınlıyorsanız, aşağıdaki durumu alırsınız:

```
QMNAME(BOB)                STATUS(Negotiating)
```

İlgili başvurular

dspmqr

CP4I Yerel HA kuyruk yöneticilerinin sona erdiriliyor

Yerel bir HA grubunun bir parçası olan etkin ya da eşleme kuyruk yöneticisini sona erdirmek için **endmqm** komutunu kullanabilirsiniz.

Yordam

- Bir kuyruk yöneticisinin etkin yönetim ortamını sona erdirmek için, bu belgelerin Configuring (Yapılandırma) bölümünde [Finding Native HA queue manager](#) (Yerel HA kuyruk yöneticilerinin sona erdirilmesi)

Özel notlar

Bu belge, ABD'de kullanıma sunulan ürünler ve hizmetler için hazırlanmıştır.

IBM, bu belgede sözü edilen ürün, hizmet ya da özellikleri diğer ülkelerde kullanıma sunmayabilir. Bulduğunuz yerde kullanıma sunulan ürün ve hizmetleri yerel IBM müşteri temsilcisinden ya da çözüm ortağınızdan öğrenebilirsiniz. Bir IBM ürün, program ya da hizmetine gönderme yapılması, açık ya da örtük olarak yalnızca o IBM ürünü, programı ya da hizmetinin kullanılabilirliğini göstermez. Aynı işlevi gören ve IBM'in fikri mülkiyet haklarına zarar vermeyen herhangi bir ürün, program ya da hizmet de kullanılabilir. Ancak, IBM dışı ürün, program ya da hizmetlerle gerçekleştirilen işlemlerin değerlendirilmesi ve doğrulanması kullanıcının sorumluluğundadır.

IBM'in, bu belgedeki konularla ilgili patentleri ya da patent başvuruları olabilir. Bu belgenin size verilmiş olması, patentlerin izinsiz kullanım hakkının da verildiği anlamına gelmez. Lisansla ilgili sorularınızı aşağıdaki adrese yazabilirsiniz:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Çift byte (DBCS) bilgilerle ilgili lisans soruları için, ülkenizdeki IBM'in Fikri Haklar (Intellectual Property) bölümüyle bağlantı kurun ya da sorularınızı aşağıda adrese yazın:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japonya

Aşağıdaki paragraf, İngiltere ya da bu tür hükümlerin yerel yasalarla uyuşmadığı diğer ülkelerde geçerli değildir: INTERNATIONAL BUSINESS MACHINES CORPORATION BU YAYINI, HAK İHLALİ YAPILMAYACAĞINA DAİR GARANTİLERLE TİCARİLİK VEYA BELİRLİ BİR AMACA UYGUNLUK İÇİN ZİMNİ GARANTİLER DE DAHİL OLMAK VE FAKS BUNLARLA SINIRLI OLMAMAK ÜZERE AÇIK YA DA ZİMNİ HİÇBİR GARANTİ VERMEKSİZİN "OLDUĞU GİBİ" ESASIYLA SAĞLAMAKTADIR. Bazı ülkeler bazı işlemlerde garantinin açık ya da örtük olarak reddedilmesine izin vermez; dolayısıyla, bu bildirim sizin için geçerli olmayabilir.

Bu yayın teknik yanlışlar ya da yazım hataları içerebilir. Buradaki bilgiler üzerinde düzenli olarak değişiklik yapılmaktadır; söz konusu değişiklikler sonraki basımlara yansıtılacaktır. IBM, önceden bildirimde bulunmaksızın, bu yayında açıklanan ürünler ve/ya da programlar üzerinde iyileştirmeler ve/ya da değişiklikler yapabilir.

Bu belgede IBM dışı Web sitelerine yapılan göndermeler kullanıcıya kolaylık sağlamak içindir ve bu Web sitelerinin onaylanması anlamına gelmez. Bu Web sitelerinin içerdiği malzeme, bu IBM ürününe ilişkin malzemenin bir parçası değildir ve bu tür Web sitelerinin kullanılmasının sorumluluğu size aittir.

IBM'e bilgi ilettiğinizde, IBM bu bilgileri size karşı hiçbir yükümlülük almaksızın uygun gördüğü yöntemlerle kullanabilir ya da dağıtabilir.

(i) Bağımsız olarak yaratılan programlarla, bu program da içinde olmak üzere diğer programlar arasında bilgi değiş tokuşuna ve (ii) değiş tokuş edilen bilginin karşılıklı kullanımına olanak sağlamak amacıyla bu program hakkında bilgi sahibi olmak isteyen lisans sahipleri şu adrese yazabilirler:

IBM Corporation
Yazılım Birlikte Çalışabilirlik Koordinatörü, Bölüm 49XA
3605 Highway 52 N

Rochester, MN 55901
U.S.A.

Bu tür bilgiler, ilgili kayıt ve koşullar altında ve bazı durumlarda bedelli olarak edinilebilir.

Bu belgede açıklanan lisanslı program ve bu programla birlikte kullanılacak tüm lisanslı malzeme, IBM tarafından, IBM Müşteri Sözleşmesi, IBM Uluslararası Program Lisansı Sözleşmesi ya da eşdeğer herhangi bir sözleşmenin kayıt ve koşulları altında sağlanır.

Burada belirtilen performans verileri denetimli bir ortamda elde edilmiştir. Bu nedenle, başka işletim ortamlarında çok farklı sonuçlar alınabilir. Bazı ölçümler geliştirilme düzeyindeki sistemlerde yapılmıştır ve bu ölçümlerin genel kullanıma sunulan sistemlerde de aynı olacağı garanti edilemez. Ayrıca, bazı sonuçlar öngörü yöntemiyle elde edilmiş olabilir. Dolayısıyla, gerçek sonuçlar farklı olabilir. Bu belgenin kullanıcıları, kendi ortamları için geçerli verileri kendileri doğrulamalıdır.

IBM dışı ürünlerle ilgili bilgiler, bu ürünleri sağlayan firmalardan, bu firmaların yayın ve belgelerinden ve genel kullanıma açık diğer kaynaklardan alınmıştır. IBM bu ürünleri sınınamamıştır ve IBM dışı ürünlerle ilgili performans doğruluğu, uyumluluk gibi iddiaları doğrulayamaz. IBM dışı ürünlerin yeteneklerine ilişkin sorular, bu ürünleri sağlayan firmalara yöneltilmelidir.

IBM'in gelecekteki yönelim ve kararlarına ilişkin tüm bildirimler değişebilir ve herhangi bir duyuruda bulunulmadan bunlardan vazgeçilebilir; bu yönelim ve kararlar yalnızca amaç ve hedefleri gösterir.

Bu belge, günlük iş ortamında kullanılan veri ve raporlara ilişkin örnekler içerir. Örneklerin olabildiğince açıklayıcı olması amacıyla kişi, şirket, marka ve ürün adları belirtilmiş olabilir. Bu adların tümü gerçek dışıdır ve gerçek iş ortamında kullanılan ad ve adreslerle olabilecek herhangi bir benzerlik tümüyle rastlantıdır.

YAYIN HAKKI LİSANSI:

Bu belge, çeşitli işletim platformlarında programlama tekniklerini gösteren, kaynak dilde yazılmış örnek uygulama programları içerir. Bu örnek programları, IBM'e herhangi bir ödemede bulunmadan, örnek programların yazıldığı işletim altyapısına ilişkin uygulama programlama arabirimiyle uyumlu uygulama programlarının geliştirilmesi, kullanılması, pazarlanması ya da dağıtılması amacıyla herhangi bir biçimde kopyalayabilir, değiştirebilir ve dağıtabilirsiniz. Bu örnekler her koşul altında tüm ayrıntılarıyla sınınamamıştır. Dolayısıyla, IBM bu programların güvenilirliği, bakım yapılabilirliği ya da işlevleri konusunda açık ya da örtük güvence veremez.

Bu bilgileri elektronik kopya olarak görüntülediyseniz, fotoğraflar ve renkli resimler görünmeyebilir.

Programlama arabirimi bilgileri

Programlama arabirimi bilgileri (sağlandıysa), bu programla birlikte kullanılmak üzere uygulama yazılımları yaratmanıza yardımcı olmak üzere hazırlanmıştır.

Bu kitap, müşterinin WebSphere MQ hizmetlerini edinmek üzere program yazmasına olanak tanıyan, amaçlanan programlama arabirimlerine ilişkin bilgiler içerir.

Ancak, bu bilgiler tanılama, değiştirme ve ayarlama bilgilerini de içerebilir. Tanılama, değiştirme ve ayarlama bilgileri, uygulama yazılımlarınızda hata ayıklamanıza yardımcı olur.

Önemli: Bu tanılama, değiştirme ve ayarlama bilgilerini bir programlama arabirimi olarak kullanmayın; bu, değişiklik söz konusu olduğunda kullanılır.

Ticari Markalar

IBM, IBM logosu, ibm.com, IBM Corporation 'ın dünya çapında birçok farklı hukuk düzeninde kayıtlı bulunan ticari markalarıdır. IBM ticari markalarının güncel bir listesini Web üzerinde "Telif hakkı ve ticari marka bilgileri" www.ibm.com/legal/copytrade.shtml adresinde bulabilirsiniz. Diğer ürün ve hizmet adları IBM'in veya diğer şirketlerin ticari markaları olabilir.

Microsoft ve Windows, Microsoft Corporation'ın ABD ve/veya diğer ülkelerdeki ticari markalarıdır.

UNIX, The Open Group şirketinin ABD ve diğer ülkelerdeki tescilli ticari markasıdır.

Linux, Linus Torvalds'ın ABD ve/ya da diđer ülkelerdeki tescilli ticari markasıdır.

Bu ürün, Eclipse Project (<https://www.eclipse.org/>) tarafından geliştirilen yazılımları içerir.

Java ve Java tabanlı tüm markalar ve logolar, Oracle firmasının ve/ya da iştiraklerinin markaları ya da tescilli markalarıdır.



Parça numarası:

(1P) P/N: