9.2

IBM MQ v kontejnerech

Poznámka

Než začnete používat tyto informace a produkt, který podporují, přečtěte si informace, které uvádí “Poznámky” na stránce 157.

Toto vydání se vztahuje k verzi 9 vydání 2 produktu IBM® MQ a ke všem následujícím vydáním a modifikacím, dokud nebude v nových vydáních uvedeno jinak.

Když odešlete informace do IBM, udělíte společnosti IBM nevýlučné právo použít nebo distribuovat informace libovolným způsobem, který společnost považuje za odpovídající, bez vzniku jakýchkoliv závazků vůči vám.

Obsah

Multi IBM MQ v kontejnerech a IBM Cloud Pak for Integration

Kontejnery umožňují zabalit správce front IBM MQ nebo aplikaci klienta IBM MQ se všemi závislostmi do standardizované jednotky pro vývoj softwaru.

Můžete spustit IBM MQ pomocí IBM MQ Operator v systému Red Hat® OpenShift®. To lze provést pomocí IBM Cloud Pak for Integration, IBM MQ Advanced nebo IBM MQ Advanced for Developers.

Produkt IBM MQ můžete také spustit v kontejneru, který sami sestavujete.

MQ Adv. CD Další informace o IBM MQ Operator viz následující odkazy.

Multi Plánování pro IBM MQ v kontejnerech

Když plánujete pro produkt IBM MQ v kontejnerech, zvažte podporu, kterou produkt IBM MQ poskytuje pro různé architektonické volby, jako je například způsob správy vysoké dostupnosti a jak zabezpečit správce front.

Informace o této úloze

Před plánováním architektury produktu IBM MQ v kontejnerech byste se měli seznámit se základními koncepty IBM MQ (viz Technický přehled IBM MQ) i základními koncepty Kubernetes/Red Hat OpenShift (viz architektura Red Hat OpenShift Container Platform).

Procedura

- “Zvolení, jak se má produkt IBM MQ používat v kontejnerech” na stránce 5.
- “Podpora pro IBM MQ Operator” na stránce 6.
- “Podpora pro sestavení vlastních obrázků kontejneru správce front IBM MQ” na stránce 12.
- “Aspekty úložiště pro IBM MQ Operator” na stránce 11.
- “Vysoká dostupnost pro IBM MQ v kontejnerech” na stránce 16.
- “Zotavení z havárie pro produkt IBM MQ v kontejnerech” na stránce 18.
- “Ověření uživatele a autorizace pro produkt IBM MQ v kontejnerech” na stránce 18.

Zvolení, jak se má produkt IBM MQ používat v kontejnerech

Existuje více voleb pro použití produktu IBM MQ v kontejnerech: můžete zvolit použití IBM MQ Operator, který používá předem seskupené kontejnerové obrazy nebo můžete sestavit vlastní obrazy a kód implementace.

Použití produktu IBM MQ Operator

OpenShift CP4I

Plánujete-li implementovat v produktu Red Hat OpenShift Container Platform, pravděpodobně budete chtít používat IBM MQ Operator.

Produkt IBM MQ Operator přidá nový vlastní prostředek `QueueManager` do produktu Red Hat OpenShift Container Platform. Operátor sleduje nové definice správce front a poté je změní na nezbytné prostředky s nízkou úrovní, jako například prostředky `StatefulSet` a `Service`. V případě nativní vysoké dostupnosti může operátor také provést komplexní průběžnou aktualizaci instancí správce front. Viz “Faktory ovlivňující provádění vlastní průběžné aktualizace správce front nativní vysoké dostupnosti” na stránce 152

Některé funkce IBM MQ nejsou podporovány při použití IBM MQ Operator. Chcete-li provést některou z následujících akcí, budete muset sestavit vlastní obrazy a grafy:

- Použijte rozhraní REST API pro administraci nebo systém zpráv.
- Použijte kteroukoli z následujících komponent produktu MQ:
  - Managed File Transfer Agents a jeho prostředky. Produkt IBM MQ Operator však můžete použít k poskytnutí jednoho nebo více správců front Coordination, Command nebo Agent.
  - AMQP
  - IBM MQ Bridge to Salesforce
  - IBM MQ Bridge to blockchain (není podporováno v kontejnerech).
  - IBM MQ Telemetry Transport (MQTT).
- Upravte volby použité s příkazy **crtmqm**, **strmqm** a **endmqm**, např. konfigurací stránek souboru protokolu. Většinu voleb lze nakonfigurovat pomocí souboru INI.

Všimněte si, že IBM MQ Operator a kontejnery se vyvíjejí rychle, a proto nejsou podporovány pod vydáními produktu Long Term Support.

IBM MQ Operator obsahuje jak předem sestavené kontejnerové obrazy, tak i kód implementace pro spuštění v produktu Red Hat OpenShift Container Platform. Produkt IBM MQ Operator lze použít k implementaci poskytnutého kontejnerového obrazu IBM MQ nebo kontejnerového obrazu ve vrstvě nad ním, nelze jej ale použít k implementaci vlastních vestavěných kontejnerových obrazů MQ.

Vytváření vlastních obrazů a kódu implementace

Multi

Jedná se o nejflexibilnější řešení kontejneru, které ale od vás vyžaduje značné dovednosti v konfiguraci kontejnerů a abyste "vlastnili" výsledný kontejner. Pokud nemáte v úmyslu používat platformu Red Hat OpenShift Container Platform, budete muset sestavit vlastní obrazy a kód implementace.

K dispozici jsou ukázky pro sestavení vlastních obrazů. Viz “Sestavení vlastního kontejneru IBM MQ a kódu implementace” na stránce 144.

Související pojmy

“Podpora pro IBM MQ Operator” na stránce 6
IBM MQ Operator je podporován pouze při implementaci v Red Hat OpenShift Container Platform.

“Podpora pro sestavení vlastních obrázků kontejneru správce front IBM MQ” na stránce 12
Produkt IBM MQ poskytuje kód pro sestavení kontejneru správce front IBM MQ v GitHub. To je založeno na procesu, který IBM používá k vytvoření svého vlastního podporovaného kontejneru, a můžete použít toto úložiště GitHub ke zjednodušení a urychlení sestavení vašich vlastních obrázků kontejneru.

OpenShift CP4I CD EUS Podpora pro IBM MQ Operator

IBM MQ Operator je podporován pouze při implementaci v Red Hat OpenShift Container Platform.

IBM MQ Operator používá obrazy založené na vydáních IBM MQ Continuous Delivery (CD) přestože vydání EUS (Extended Update Support) je k dispozici s IBM Cloud Pak for Integration. Vydání CD jsou podporovány po dobu až jednoho roku nebo pro dvě vydání CD, podle toho, která doba je delší. Verze Long Term Support produktu IBM MQ nejsou k dispozici prostřednictvím IBM MQ Operator. Produkt IBM Cloud Pak for Integration 2020.4.1 je verze EUS (Extended Update Support), která je podporována po dobu 18 měsíců, používáte-li verzi produktu IBM MQ označenou jako `-eus`. Jinak IBM MQ 9.2 je považován za vydání Continuous Delivery s IBM MQ Operator.

IBM MQ Operator používá kontejnerové obrazy, které poskytují instalaci produktu IBM MQ v Red Hat Universal Base Image (UBI), který obsahuje klíčové knihovny Linux® a obslužné programy použité produktem IBM MQ. UBI je podporován společností Red Hat při spuštění v Red Hat OpenShift.

Produkt IBM MQ Operator je podporován na architekturách `amd64` a `s390x` (z/Linux).

Související pojmy

“Podpora pro sestavení vlastních obrázků kontejneru správce front IBM MQ” na stránce 12
Produkt IBM MQ poskytuje kód pro sestavení kontejneru správce front IBM MQ v GitHub. To je založeno na procesu, který IBM používá k vytvoření svého vlastního podporovaného kontejneru, a můžete použít toto úložiště GitHub ke zjednodušení a urychlení sestavení vašich vlastních obrázků kontejneru.

Podpora verze pro IBM MQ Operator

Mapování mezi podporovanými verzemi IBM MQ, Red Hat OpenShift Container Platform a IBM Cloud Pak for Integration.

- “Dostupné verze produktu IBM MQ” na stránce 7
- “Kompatibilní verze Red Hat OpenShift Container Platform” na stránce 8
- “Verze IBM Cloud Pak for Integration” na stránce 8
- “Dostupné verze produktu IBM MQ ve starších operátorech” na stránce 8
- “Kompatibilní verze Red Hat OpenShift Container Platform pro starší operátory” na stránce 9

Dostupné verze produktu IBM MQ

Kanál operátoru	Verze operátoru	Verze IBM MQ							
		9.1.5	9.2.0 CD	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	⚠	⚠	→	⚠	⚠	●	●	●

Klíč:

●
Dostupná podpora Continuous Delivery

▯
Extended Update Support k dispozici

→
K dispozici pouze během migrace z operandu Extended Update Support na operand Continuous Delivery.

⚠
Zamítnuto. Vzhledem k tomu, že vydání IBM MQ vychází z podpory, mohou být stále konfigurovatelné v operátoru, ale již nejsou vhodné pro podporu a mohou být v budoucích vydáních odebrány.

Úplné podrobnosti o jednotlivých verzích, včetně podrobných funkcí, změn a oprav v jednotlivých verzích viz “Historie vydání pro IBM MQ Operator” na stránce 19.

Kompatibilní verze Red Hat OpenShift Container Platform

Kanál operátoru	Verze operátoru	Verze Red Hat OpenShift Container Platform[1]				
		4.6	4.7[2]	4.8	4.9	4.10
v1.6	1.6	●	●	●	●	●
v1.7	1.7	●	●	●	●	●
v1.8	1.8	●	●	●	●	●

Klíč:

● Dostupná podpora Continuous Delivery

▯ Extended Update Support k dispozici

Verze IBM Cloud Pak for Integration

Produkt IBM MQ Operator 1.8.x je podporován pro použití jako součást produktu IBM Cloud Pak for Integration verze 2021.4.1 nebo nezávisle.

Produkt IBM MQ Operator 1.7.x je podporován pro použití jako součást produktu IBM Cloud Pak for Integration verze 2021.4.1 nebo nezávisle.

Produkt IBM MQ Operator 1.6.x je podporován pro použití jako součást IBM Cloud Pak for Integration verze 2021.2.1, 2021.3.1 nebo nezávisle.

Objekt IBM MQ Operator 1.5.x již není podporován.

Objekt IBM MQ Operator 1.4.x již není podporován.

Objekt IBM MQ Operator 1.3.x již není podporován.

Produkt IBM MQ Operator 1.2.x již není podporován.

Moduly IBM MQ Operators 1.1.x a 1.0.x již nejsou podporovány.

Dostupné verze produktu IBM MQ ve starších operátorech

Následující tabulka se vztahuje na verze produktu IBM MQ Operator, které nyní dosáhly „konce životnosti“.

Kanál operátoru	Verze operátoru	Verze IBM MQ							
		9.1.5	9.2.0 CD	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	⚠	⚠	⚠					

[1] Verze Red Hat OpenShift Container Platform jsou předmětem svých vlastních dat podpory. Další informace viz Red Hat OpenShift Container Platform Life Cycle Policy.

[2] IBM MQ Operator závisí na IBM Cloud Pak foundational services. Chcete-li použít produkt Red Hat OpenShift Container Platform 4.7, měli byste nejprve upgradovat verzi IBM Cloud Pak foundational services.

Kanál operátoru	Verze operátoru	Verze IBM MQ							
		9.1.5	9.2.0 CD	9.2.0 EUS	9.2.1	9.2.2	9.2.3	9.2.4	9.2.5
v1.4	1.4	⚠	⚠	→	⚠				
v1.5	1.5	⚠	⚠	→	⚠	⚠			

Klíč:

→
K dispozici pouze během migrace z operandu Extended Update Support na operand Continuous Delivery.

⚠
Zamítnuto. Vzhledem k tomu, že produkt vydání IBM MQ vychází z podpory, mohou být stále konfigurovatelné v IBM MQ Operator, ale již nejsou vhodné pro podporu.

Úplné podrobnosti o jednotlivých verzích, včetně podrobných funkcí, změn a oprav v jednotlivých verzích viz “Historie vydání pro IBM MQ Operator” na stránce 19.

Kompatibilní verze Red Hat OpenShift Container Platform pro starší operátory

Následující tabulka se vztahuje na verze produktu IBM MQ Operator, které nyní dosáhly „konce životnosti“.

Kanál operátoru	Verze operátoru	Verze Red Hat OpenShift Container Platform[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			⚠	⚠	⚠	⚠	⚠

Klíč:

→
K dispozici pouze během migrace z operandu Extended Update Support na operand Continuous Delivery.

⚠
IBM MQ Operator verze dosáhla „konce životnosti“, ale byla dříve k dispozici na této verzi produktu Red Hat OpenShift Container Platform

[3] Verze Red Hat OpenShift Container Platform jsou předmětem svých vlastních dat podpory. Další informace viz Red Hat OpenShift Container Platform Life Cycle Policy.

[4] Red Hat OpenShift Container Platform 4.4 končí, dosáhlo „konce životnosti“. Další informace viz Red Hat OpenShift Container Platform Life Cycle Policy.

[5] Red Hat OpenShift Container Platform 4.5 dosáhl "konce životnosti". Další informace viz Red Hat OpenShift Container Platform Life Cycle Policy.

[6] IBM MQ Operator závisí na IBM Cloud Pak foundational services. Chcete-li použít produkt Red Hat OpenShift Container Platform 4.7, měli byste nejprve upgradovat verzi IBM Cloud Pak foundational services.

OpenShift CP4I Závislosti pro IBM MQ Operator

IBM MQ Operator závisí na IBM Cloud Pak foundational services Operator, který rovněž instaluje operátor IBM Operand Deployment Lifecycle Manager (ODLM). Tyto operátory budou nainstalovány automaticky při instalaci IBM MQ Operator. Tito závislí operátoři mají malý prostor CPU a paměťový nárok a používají se k implementaci dalších prostředků za určitých okolností.

Když vytváříte `QueueManager`, IBM MQ Operator vytvoří operand `OperandRequest` pro další služby, které potřebuje. Produkt `OperandRequest` je splněn operátorem ODLM a v případě potřeby nainstaluje a vytvoří instanci požadovaných služeb. Které služby jsou vyžadovány, je určeno na základě licenční smlouvy přijaté při implementaci správce front a na základě kterých komponent správce front jsou požadovány.

- Vyberete-li licenci IBM MQ Advanced nebo IBM MQ Advanced for Developers, nebudou vyžadovány žádné další služby. Například v následujícím případě se IBM Cloud Pak foundational services nepoužívá:

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

- Pokud vyberete licenci IBM Cloud Pak for Integration a vyberete povolení webového serveru, IBM MQ Operator také vytvoří instanci operátoru IBM Identity and Access Management (IAM), aby povolil jednotné přihlášení. Operátor IAM bude vždy k dispozici, pokud jste nainstalovali operátor IBM Cloud Pak for Integration. Příklad:

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

  Pokud však zakážete webový server, nevyžaduje se IBM Cloud Pak foundational services. Příklad:

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

Starší verze produktu IBM MQ Operator vždy požadovaly instalaci produktu IBM Licensing Operator (a jeho závislostí) ke sledování použití licencí. Od verze IBM MQ Operator 1.5 není licenční služba požadována a je třeba ji požadovat samostatně.

IBM MQ Operator vyžaduje 1 jádro CPU a paměť 1 GB. Podrobný rozpis požadavků na hardware a software pro závislé operátory viz Požadavky na hardware a doporučení pro základní služby.

Můžete si vybrat množství CPU a paměti, které vaši správci front používají. Další informace naleznete v tématu “.spec.queueManager.resources” na stránce 133.

Související odkazy

“Odkaz na licenci pro mq.ibm.com/v1beta1” na stránce 124

OpenShift CP4I Oprávnění s vymezeným klastrem vyžadovaná produktem IBM MQ Operator

Produkt IBM MQ Operator vyžaduje oprávnění s vymezeným klastrem ke správě webhooků pro příjem a ukázek a ke čtení informací o třídě úložiště a verzi klastru.

IBM MQ Operator vyžaduje následující oprávnění s vymezeným klastrem:

- Oprávnění ke správě webhooků pro příjem. To umožňuje vytvářet, načítat a aktualizovat specifické webhooky, které se používají v procesu vytváření a správy kontejnerů poskytnutých operátorem.

  - Skupiny rozhraní API: **admissionregistration.k8s.io**
  - Prostředky: **validatingwebhookconfigurations**
  - Příkazová slova: **create, get, update**
- Oprávnění k vytváření a správě prostředků, které se používají v konzole Red Hat OpenShift k poskytování ukázek a úseků kódu při vytváření vlastních prostředků.
  - Skupiny rozhraní API: **console.openshift.io**
  - Prostředky: **consoleyamlsamples**
  - Příkazová slova: **create, get, update, delete**
- Oprávnění ke čtení verze klastru. Operator tak může vrátit zpět veškeré problémy s prostředím klastru.
  - Skupiny rozhraní API: **config.openshift.io**
  - Prostředky: **clusterversions**
  - Příkazová slova: **get, list, watch**
- Oprávnění ke čtení paměťových tříd v klastru. Operator tak může vrátit zpět veškeré problémy s vybranými paměťovými třídami úložiště v kontejnerech.
  - Skupiny rozhraní API: **storage.k8s.io**
  - Prostředky: **storageclasses**
  - Příkazová slova: **get, list**

OpenShift CP4I Kubernetes Aspekty úložiště pro IBM MQ Operator

IBM MQ Operator se spouští ve dvou režimech úložiště:

- **Přechodné úložiště** se používá, když se všechny stavové informace pro kontejner mohou vyřadit po restartu kontejneru. Běžně se používá při vytváření předváděcích prostředí nebo při vývoji se samostatnými správci front.
- **Trvalé úložiště** je běžná konfigurace produktu IBM MQ, která zajišťuje, že pokud je kontejner restartován, budou v restartovaném kontejneru existující konfigurace, protokoly a trvalé zprávy k dispozici.

IBM MQ Operator poskytuje schopnost pro přizpůsobení charakteristik úložiště, které se mohou výrazně lišit v závislosti na prostředí a požadovaném režimu úložiště.

Přechodné úložiště

Produkt IBM MQ je stavová aplikace a uchovává tento stav pro úložiště pro zotavení v případě restartování. Pokud používáte dočasné úložiště, všechny informace o stavu správce front se při restartu ztratí. To zahrnuje:

- Všechny zprávy.
- Všichni správci front do stavu komunikace správce front (pořadová čísla zpráv kanálu).
- Identita klastru MQ správce front.
- Stav všech transakcí.
- Konfiguraci všech správců front.
- Všechna lokální diagnostická data.

Z tohoto důvodu byste měli zvážit, zda přechodné úložiště je vhodný přístup pro scénář produkce, testování nebo vývoje. Například u všech zpráv, u nichž je známo, že jsou dočasné a že správce front není členem klastru MQ. Kromě likvidace veškerého stavu systému zpráv při restartu, bude také vyřazena konfigurace správce front. Chcete-li povolit úplně přechodný kontejner, musí být konfigurace produktu IBM MQ přidána do samotného kontejnerového obrazu (další informace viz “Sestavení obrazu pomocí vlastních souborů MQSC a INI s použitím rozhraní CLI Red Hat OpenShift” na stránce 110). Není-li tento proces dokončen, bude muset být při každém restartování kontejneru nakonfigurován produkt IBM MQ.

OpenShift CP4I Chcete-li např. nakonfigurovat produkt IBM MQ s přechodným úložištěm, měl by typ úložiště `QueueManager` obsahovat následující:

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

Trvalé úložiště

OpenShift CP4I

Produkt IBM MQ za normálních okolností pracuje s trvalým úložištěm, aby bylo zajištěno, že správce front zachová své trvalé zprávy a konfiguraci po restartu. Proto se jedná o výchozí chování. Kvůli různým poskytovatelům úložiště a různým schopnostem každé podpory to často znamená, že je nezbytné přizpůsobit konfiguraci. Níže uvedený příklad ukazuje společná pole, které upravují konfiguraci úložiště MQ v rozhraní v1beta1 API:

- spec.queueManager.availability řídí režim dostupnosti. Používáte-li `SingleInstance`, vyžadujete pouze úložiště `ReadWriteOnce`, zatímco `multiInstance` vyžaduje paměťovou třídu, která podporuje `ReadWriteMany` se správnými charakteristikami zamykání souborů. IBM MQ poskytuje prohlášení o podpoře a prohlášení o testování. Režim dostupnosti má také vliv na rozvržení trvalého svazku. Další informace viz “Vysoká dostupnost pro IBM MQ v kontejnerech” na stránce 16.
- spec.queueManager.storage řídí nastavení individuálního úložiště. Správce front lze nakonfigurovat tak, aby používal jeden až čtyři trvalé svazky.

V následujícím příkladu je zobrazen úsek jednoduché konfigurace pomocí správce front s jednou instancí:

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

V následujícím příkladu je zobrazen úsek kódu konfigurace správce front s více instancemi, s jinou než výchozí třídou úložiště a s úložištěm souborů vyžadujícím doplňkové skupiny:

```
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]
```

Poznámka: Doplňkové skupiny můžete také konfigurovat pomocí správců front s jednou instancí.

Poznámka: Pokud používáte nativní vysokou dostupnost (viz “Vysoká dostupnost pro IBM MQ v kontejnerech” na stránce 16), nevyžadujte sdílené systémy souborů. Zejména byste neměli používat systém NFSv3.

Linux Podpora pro sestavení vlastních obrázků kontejneru správce front IBM MQ

Produkt IBM MQ poskytuje kód pro sestavení kontejneru správce front IBM MQ v GitHub. To je založeno na procesu, který IBM používá k vytvoření svého vlastního podporovaného kontejneru, a můžete použít toto úložiště GitHub ke zjednodušení a urychlení sestavení vašich vlastních obrázků kontejneru.

Tento kód je uveden v úložišti mq-container GitHub zde: https://github.com/ibm-messaging/mq-container. To je poskytnuto v rámci licence produktu Apache 2.0 s podporou poskytovanou komunitou.

Úložiště nepoužívá standardní balíky rm Linux; používá komprimovaný balík pro implementace kontejneru. Výhodou tohoto přístupu je to, že můžete pracovat ve více zabezpečeném prostředí kontejneru bez nutnosti eskalovaných oprávnění. Avšak to má vliv na dostupné volby zabezpečení, protože produkt IBM MQ tradičně používá eskalovaná oprávnění pro ověření založené na operačním systému. V případě implementace kontejneru není použití ověření založeného na operačním systému obvykle dobrou praxí; místo toho můžete použít vzájemné ověřování TLS nebo LDAP. Pomocí produktu IBM MQ Advanced for Developersmůžete také použít ověřování založené na souborech, která uživatelům umožní rychle začít.

Správce front replikovaných dat (RDQM) není v kontejnerovém prostředí podporován. Podobné schopnosti můžete získat do RDQM pomocí produktu “Nativní vysoká dostupnost” na stránce 91.

Související pojmy

“Podpora pro IBM MQ Operator” na stránce 6
IBM MQ Operator je podporován pouze při implementaci v Red Hat OpenShift Container Platform.

Neinstalační obrazy produktu IBM MQ

Linux Anotace licencí při sestavování vlastního obrazu kontejneru produktu IBM MQ

Anotace licencí umožňují sledovat využití na základě limitů definovaných v kontejneru, a nikoli na základním počítači. Konfigurujete klienty pro implementaci kontejneru se specifickými anotacemi, které IBM License Service používá ke sledování využití.

Při implementaci obrazu vlastního kontejneru produktu IBM MQ se používají dva běžné přístupy k licencování:

- Udělení licence pro celý počítač, na kterém je spuštěn kontejner.
- Udělení licence pro kontejner na základě přiřazených limitů.

Obě volby jsou k dispozici klientům a další podrobnosti lze nalézt na stránce IBM Container Licenses v programu Passport Advantage.

Pokud má být kontejner produktu IBM MQ licencován na základě limitů kontejnerů, je třeba produkt IBM License Service nainstalovat, aby bylo možné sledovat použití. Další informace týkající se podporovaných prostředí a pokynů k instalaci naleznete na stránce ibm-licensing-operator na GitHub.

IBM License Service je nainstalován v klastru Kubernetes, kde je implementován kontejner IBM MQ a kde se ke sledování využití používají anotace Pod. Klienti tedy musejí implementovat Pod se specifickými anotacemi, které pak produkt IBM License Service používá. Na základě vašich oprávnění a schopností implementovaných v rámci kontejneru použijte jednu nebo více následujících anotací:

- “Kontejner IBM MQ Advanced” na stránce 14
- “Kontejner IBM MQ Advanced High Availability Replica” na stránce 14
- “Kontejner IBM MQ Base” na stránce 14
- “Kontejner IBM MQ Base High Availability Replica” na stránce 14
- “Kontejner IBM MQ Advanced for Developers” na stránce 14
- “Kontejner IBM MQ Advanced s oprávněním CP4I (Production)” na stránce 14
- “Kontejner IBM MQ Advanced High Availability Replica s oprávněním CP4I (Production)” na stránce 14
- “Kontejner IBM MQ Advanced s oprávněním CP4I (Non-Production)” na stránce 15
- “Kontejner IBM MQ Advanced High Availability Replica s oprávněním CP4I (Non-Production)” na stránce 15
- “IBM MQ Base s oprávněním CP4I (Production)” na stránce 15
- “IBM MQ Base High Availability Replica s oprávněním CP4I (Production)” na stránce 15

- “IBM MQ Base s oprávněním CP4I (Non-Production)” na stránce 15
- “IBM MQ Base High Availability Replica s oprávněním CP4I (Non-Production)” na stránce 15

Kontejner IBM MQ Advanced

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

Kontejner IBM MQ Advanced High Availability Replica

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

Kontejner IBM MQ Base

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

Kontejner IBM MQ Base High Availability Replica

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

Kontejner IBM MQ Advanced for Developers

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

Kontejner IBM MQ Advanced s oprávněním CP4I (Production)

```
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"
```

Kontejner IBM MQ Advanced High Availability Replica s oprávněním CP4I (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"
```

Kontejner IBM MQ Advanced s oprávněním CP4I (Non-Production)

```
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"
```

Kontejner IBM MQ Advanced High Availability Replica s oprávněním CP4I (Non-Production)

```
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"
```

IBM MQ Base s oprávněním CP4I (Production)

```
productName: "IBM MQ with CP4I License"
productID: "c661609261d5471fb4ff8970a36bccea"
productChargedContainers: "All" | "NAME_OF_CONTAINER"
productMetric: "VIRTUAL_PROCESSOR_CORE"
productCloudpakRatio: "4:1"
cloudpakName: "IBM Cloud Pak for Integration"
cloudpakId: "c8b82d189e7545f0892db9ef2731b90d"
```

IBM MQ Base High Availability Replica s oprávněním CP4I (Production)

```
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 s oprávněním CP4I (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 Base High Availability Replica s oprávněním CP4I (Non-Production)

```
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 Vysoká dostupnost pro IBM MQ v kontejnerech

K dispozici máte tři volby vysoké dostupnosti s IBM MQ Operator: **Správce front nativní vysoké dostupnosti** (který má aktivní repliku a dvě záložní repliky), **Správce front s více instancemi** (což je dvojice aktivní-pohotovostní, využívající sdílený síťový systém souborů) nebo **Jeden odolný správce front** (který nabízí jednoduchý přístup pro vysokou dostupnost používající síťové úložiště). Druhá z těchto dvou možností závisí na systému souborů, aby se zajistila dostupnost obnovitelných dat, ale nativní vysoká dostupnost nikoli. Proto, když nepoužíváte Nativní vysokou dostupnost, je dostupnost systému souborů kritická pro dostupnost správce front. Tam, kde je obnova dat důležitá, by měl systém souborů zajistit redundanci pomocí replikace.

Měli byste zvážit mít dostupnost pro **zprávy** a **služby** odděleně. S IBM MQ for Multiplatforms je zpráva uložena přesně do jednoho správce front. Takže pokud se tento správce front stane nedostupným, dočasně ztratíte přístup ke zprávám, které obsahuje. Chcete-li dosáhnout vysoké dostupnosti `zprávy`, musíte být schopni obnovit správce front co nejrychleji. Dostupnost `služby` můžete dosáhnout tím, že budete mít více instancí front pro aplikace klienta, které se mají používat, například pomocí uniformního klastru IBM MQ.

Správce front lze považovat za dvě části: data uložená na disku a běžící procesy, které umožňují přístup k datům. Libovolného správce front lze přesunout do jiného uzlu Kubernetes, pokud uchovává stejná data (poskytovaná Trvalými svazky Kubernetes) a je stále síťově adresovatelný v aplikacemi klienta. V Kubernetes je služba použita k poskytnutí konzistentní sítě identity.

IBM MQ spoléhá na dostupnost dat na trvalých svazcích. Dostupnost úložiště poskytujícího trvalé svazky je proto rozhodující pro dostupnost správce front, neboť produkt IBM MQ nemůže být dostupnější než úložiště, které používá. Chcete-li tolerovat výpadek celé zóny dostupnosti, je třeba použít poskytovatele svazků, který replikuje zápisy na disk do jiné zóny.

Správce front nativní vysoké dostupnosti

CP4I V 9.2.3

Správci front nativní vysoké dostupnosti jsou k dispozici v IBM Cloud Pak for Integration 2021.2.1, s použitím produktu IBM MQ Operator 1.6 nebo vyšší, s IBM MQ 9.2.3 nebo vyšší.

Správci front nativní vysoké dostupnosti zahrnují **aktivní** pody a dvě **repliky** Kubernetes, které běží jako součást stavové sady Kubernetes s přesně třemi replikami, každá s vlastní sadou trvalých svazků Kubernetes. Požadavky na IBM MQ pro sdílené systémy souborů také platí pro použití správce front nativní vysoké dostupnosti (s výjimkou zamykání na základě nájmů), u něhož ale nepotřebujete sdílený systém souborů. Úložiště bloků můžete používat s vhodným završujícím systémem souborů. Např. *xfs* nebo *ext4*. Doby obnovy pro správce front nativní vysoké dostupnosti jsou řízeny následujícími faktory:

1. Jak dlouho trvá instancím repliky zjistit, že aktivní instance se nezdařila. Toto je konfigurovatelné.
2. Jak dlouho trvá sondě připravenost podu Kubernetes zjistit, že je kontejner připraven se změnit a přesměrovat síťový provoz. Toto je konfigurovatelné.
3. Jak dlouho trvá, než se klienti IBM MQ znovu připojí.

Další informace naleznete v tématu “Nativní vysoká dostupnost” na stránce 91

Správce front s více instancemi

Multi

Správce front s více instancemi zahrnují **aktivní** a **pohotovostní** Pody Kubernetes, které se spouštějí jako součást stavové sady Kubernetes s přesně dvěma replikami a sadou trvalých svazků Kubernetes. Protokoly a data transakcí správce front jsou drženy ve dvou trvalých svazcích za použití sdíleného systému souborů.

Správci front s více instancemi vyžadují **aktivní** i **pohotovostní** Pody, aby měli souběžný přístup k trvalému svazku. Chcete-li provést konfiguraci, použijte trvalé svazky Kubernetes s parametrem **`access mode`** nastaveným na `ReadWriteMany`. Svazky musí také splňovat IBM MQ požadavky pro sdílené systémy souborů, protože produkt IBM MQ spoléhá na automatické uvolnění zámků souborů k podněcování překonání selhání správce front. IBM MQ produkuje seznam testovaných systémů souborů.

Doby obnovy pro správce front s více instancemi jsou řízeny následujícími faktory:

1. Jak dlouho trvá, než dojde k selhání sdíleného systému souborů, aby uvolnil zámky původně provedené aktivní instancí.
2. Jak dlouho trvá, než pohotovostní instance získá zámky, a pak se spustí.
3. Jak dlouho trvá sondě připravenost podu Kubernetes zjistit, že je kontejner připraven se změnit a přesměrovat síťový provoz. Toto je konfigurovatelné.
4. Jak dlouho trvá, než se klienti IBM MQ znovu připojí.

Jeden odolný správce front

Jeden odolný správce front je jedna instance správce front spuštěná v jednom podu Kubernetes, kde Kubernetes monitoruje správce front a v případě potřeby pod nahradí.

Požadavky IBM MQ pro sdílené systémů souborů také platí pro použití jednoho odolného správce front (s výjimkou zamykání na základě nájmů), u něhož ale nepotřebujete sdílený systém souborů. Úložiště bloků můžete používat s vhodným završujícím systémem souborů. Např. *xfs* nebo *ext4*.

Doby obnovy pro jednoho odolného správce front jsou řízeny následujícími faktory:

1. Jak dlouho trvá spuštění sondy živosti a kolik chyb toleruje. Toto je konfigurovatelné.
2. Jak dlouho trvá plánovači Kubernetes znovu na novém uzlu naplánovat nezdařený Pod.
3. Jak dlouho trvá stažení kontejnerového obrazu do nového uzlu. Použijete-li hodnotu **`imagePullPolicy`** parametru `IfNotPresent`, může tento obraz již v daném uzlu existovat.
4. Jak dlouho trvá, než se nová instance správce front spustí.
5. Jak dlouho trvá sondě připravenosti Podu Kubernetes zjistit, že je kontejner připraven. Toto je konfigurovatelné.
6. Jak dlouho trvá, než se klienti IBM MQ znovu připojí.

Důležité:

Ačkoli vzor jednoho odolného správce front nabízí některé výhody, je třeba porozumět tomu, zda lze dosáhnout cílů dostupnosti s omezeními v souvislosti se selháními uzlu.

V Kubernetes je selhaný Pod obvykle rychle obnoven, ale selhání celého uzlu se zpracovává jinak. Při použití stavové pracovní zátěže jako IBM MQ s Kubernetes StatefulSet, pokud hlavní uzel produktu Kubernetes ztratí kontakt s pracovním uzlem, nemůže určit, zda uzel selhal, nebo zda se jednoduše ztratil připojení k síti. Proto Kubernetes v tomto případě neprovede **žádnou akci**, dokud se nevyskytne jedna z následujících událostí:

1. Uzel se obnoví do stavu, v němž může hlavní uzel Kubernetes s ním komunikovat.
2. Je provedena administrativní akce, která explicitně odstraní Pod v hlavním uzlu Kubernetes. Spuštěný Pod se nemusí nutně zastavit, stačí jej odstranit z úložiště Kubernetes. Tuto administrativní akci je proto třeba velmi pečlivě zvážit.

Související úlohy

“Konfigurace vysoké dostupnosti pro správce front pomocí IBM MQ Operator” na stránce 91

Související odkazy

Konfigurace vysoké dostupnosti

OpenShift CP4I Kubernetes Zotavení z havárie pro produkt IBM MQ v kontejnerech

Musíte zvážit, na jakou havárii se chystáte. V prostředích cloudu poskytují zóny dostupnosti určitou úroveň tolerance k haváriím, a používání je mnohem snazší. Pokud máte lichý počet datových středisek (pro kvorum) a síťový odkaz s nízkou latencí, mohli byste potenciálně spustit jeden klastr Red Hat OpenShift Container Platform nebo Kubernetes s více zónami dostupnosti, každý v odděleném fyzickém umístění. Toto téma probírá aspekty pro zotavení z havárie, kde nelze splnit tato kritéria: to znamená buď sudý počet datových středisek, nebo síťový odkaz s vysokou latencí.

Pro zotavení z havárie je třeba vzít v úvahu následující skutečnosti:

- Replikace dat produktu IBM MQ (držených v jednom nebo více prostředcích `PersistentVolume`) do umístění pro zotavení z havárie
- Opětovné vytvoření správce front s použitím replikovaných dat.
- ID sítě správce front, které je viditelné pro aplikace klienta produktu IBM MQ a další správce front. Toto ID může být např. položkou DNS.

Trvalá data je třeba replikovat, buď synchronně, nebo asynchronně na server pro zotavení z havárie. Tento stav je obvykle specifický pro poskytovatele úložiště, ale lze jej také provést pomocí produktu `VolumeSnapshot`. Další informace o snímcích svazku viz Snímky svazku CSI.

Při zotavování z havárie budete muset znovu vytvořit instanci správce front na novém klastru Kybernetes s použitím replikovaných dat. Pokud používáte produkt IBM MQ Operator, budete potřebovat YAML produktu `QueueManager`, stejně jako YAML pro další podpůrné prostředky, jako například `ConfigMap` nebo `Secret`.

Související informace

ha_for_ctr.dita

Ověření uživatele a autorizace pro produkt IBM MQ v kontejnerech

Produkt IBM MQ může být nakonfigurován pro použití uživatelů a skupin LDAP. Volitelně můžete použít uživatele lokálního operačního systému a skupiny v rámci kontejnerového obrazu. IBM MQ Operator neumožňuje uživatele z uživatelů a skupin operačního systému v důsledku ochrany zabezpečení.

V kontejnerizovaném prostředí s více nájemci jsou obvykle k dispozici omezení zabezpečení, aby se zabránilo možným problémům se zabezpečením, například:

- **Zabránění použití uživatele "root" uvnitř kontejneru.**
- **Vynucení použití náhodného UID**. Například v Red Hat OpenShift Container Platform používá pro každý kontejner výchozí `SecurityContextConstraints` (s názvem `restricted`) náhodné ID uživatele.
- **Zabránění použití eskalace oprávnění**. IBM MQ v Linux používá eskalaci oprávnění ke kontrole hesel uživatelů — používá program "setuid" k tomu, aby se uživatel "root" stal tímto uživatelem.

OpenShift CP4I Chcete-li zajistit shodu s těmito bezpečnostními opatřeními, IBM MQ Operator neumožňuje použití ID, která jsou definována v knihovnách operačního systému v kontejneru. V kontejneru není definováno žádné ID uživatele nebo skupiny `mqm`. Při použití IBM MQ v IBM Cloud Pak for Integration a Red Hat OpenShift musíte nakonfigurovat svého správce front pro použití LDAP pro ověření a autorizaci uživatele. Informace o konfiguraci produktu IBM MQ viz Ověření připojení: Úložiště uživatelů a Autorizace LDAP

Multi Plánování rozšiřitelnosti a výkonu pro produkt IBM MQ v kontejnerech

Ve většině případů je změna měřítka a výkon produktu IBM MQ v kontejnerech stejné jako IBM MQ for Multiplatforms. Existuje však několik dalších limitů, které mohou být uloženy platformou kontejneru.

Informace o této úloze

Když plánujete rozšiřitelnost a výkon pro produkt IBM MQ v kontejnerech, zvažte následující možnosti:

Procedura

- **Omezte počet podprocesů a procesů.**

  Produkt IBM MQ používá podprocesy ke správě souběžnosti. V produktu Linuxjsou podprocesy implementovány jako procesy, takže můžete narazit na limity uložené platformou kontejneru nebo operačním systémem, a to až na maximální počet procesů. V produktu Red Hat OpenShift Container Platformje standardní mezní hodnota 4096 procesů na kontejner (1024 procesů až do OpenShift 4.11). I když je to vhodné pro velkou většinu scénářů, mohou existovat případy, kdy to může mít vliv na počet připojení klienta pro správce front.

  Limit počtu procesů v produktu Kubernetes může být konfigurován administrátorem klastru s použitím nastavení konfigurace kubelet **`podPidsLimit`**. Viz Omezení ID procesu a rezervace v dokumentaci Kubernetes . V produktu Red Hat OpenShift Container Platformmůžete také vytvořit vlastní prostředek produktu **`ContainerRuntimeConfig`** k úpravě parametrů CRI-O.

  Ve své konfiguraci produktu IBM MQ můžete také nastavit maximální počet připojení klienta pro správce front. Viz Limity kanálu připojení serveru pro použití limitů pro jednotlivé kanály připojení serveru a Atribut INI MAXCHANNELS INI pro použití limitů na celého správce front.

- **Omezte počet svazků.**

  V cloudu a v systémech kontejnerů se běžně používají svazky úložišť připojené k síti. Počet svazků, které lze připojit k uzlům Linux , je omezen. Například AWS EC2 omezuje maximální počet na 30 svazků na virtuální počítač. Red Hat OpenShift Container Platform má podobný limit, jako je Microsoft Azure a Google Cloud Platform.

  Nativní správce front HA vyžaduje jeden svazek pro každou ze tří instancí a vynucuje, aby se instance rozloží mezi uzly. Správce front však můžete nakonfigurovat tak, aby používal tři svazky na instanci (data správce front, protokoly obnovení a trvalá data).

- **Použijte techniky škálování IBM MQ .**

  Místo malého počtu velkých správců front může být užitečné použít techniky škálování IBM MQ , jako je například IBM MQ uniformní klastry, aby bylo možné spustit více správců front se stejnou konfigurací. To má přidaný přínos, že se sníží dopad jednoho restartování kontejneru (například jako součást údržby platformy kontejneru).

OpenShift CP4I CD EUS Použití IBM MQ v IBM Cloud Pak for Integration a Red Hat OpenShift

IBM MQ Operator implementuje a spravuje IBM MQ jako součást produktu IBM Cloud Pak for Integration nebo samostatně v produktu Red Hat OpenShift Container Platform

Procedura

- “Historie vydání pro IBM MQ Operator” na stránce 19.
- “Migrace IBM MQ do produktu IBM Cloud Pak for Integration” na stránce 36.
- “Instalace a odinstalace produktu IBM MQ Operator v systému Red Hat OpenShift” na stránce 58.
- “Upgrade produktu IBM MQ Operator a správců front” na stránce 70.
- “Implementace a konfigurace správců front pomocí IBM MQ Operator” na stránce 78.
- “Provozování produktu IBM MQ pomocí IBM MQ Operator” na stránce 113.
- “Odkaz rozhraní API pro IBM MQ Operator” na stránce 124.

OpenShift CP4I CD EUS Historie vydání pro IBM MQ Operator

IBM MQ Operator

IBM MQ Operator 1.8.2

CD

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2021.4.1

Kanál operátoru
`v1.8`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Red Hat OpenShift Container Platform 4.6 a vyšší

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.8 a vyšší (kanál v3)

Novinky

- Aktualizace pouze pro zabezpečení postavená na operátorovi IBM MQ 1.8.0.
- Zranitelnosti, které jsou řešeny, jsou podrobně popsány na této Vývěsce o zabezpečení.

IBM MQ Operator 1.8.1

CD

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2021.4.1

Kanál operátoru
`v1.8`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Red Hat OpenShift Container Platform 4.6 a vyšší

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.8 a vyšší (kanál v3)

Novinky

- Aktualizace pouze pro zabezpečení postavená na operátorovi IBM MQ 1.8.0.
- Zranitelnosti, které jsou řešeny, jsou podrobně popsány na této Vývěsce o zabezpečení.

IBM MQ Operator 1.8.0

CD

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2021.4.1

Kanál operátoru
`v1.8`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Red Hat OpenShift Container Platform 4.6 a vyšší

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.8 a vyšší (kanál v3)

Novinky

- Přidá stavové podmínky pro zamítnuté verze produktu IBM MQ.

Změny

- Obrazy přesunuty ze serveru Docker Hub do IBM Container Registry.
  - Zákazníci s pravidly brány firewall by je mohli potřebovat upravit pro přístup k obrazům v produktu IBM Container Registry.
  - Při upgradu na produkt IBM MQ Operator 1.8.0 se zákazníci se systémem Airgap setkají s restartem uzlu.
- Zamítnuté verze: IBM MQ 9.1.5, 9.2.0 CD, 9.2.1, 9.2.2. Tyto verze nemusí být sladěny s budoucími verzemi produktu IBM MQ Operator.
- Změny v licenční logice: Zákazníci upgradující na produkt IBM MQ 9.2.5 mohou používat pouze licence uvedené pro práci s produktem IBM MQ 9.2.5. Viz téma “Odkaz na licenci pro mq.ibm.com/v1beta1” na stránce 124.
- Zranitelnosti, které jsou řešeny, jsou podrobně popsány na této Vývěsce o zabezpečení.

IBM MQ Operator 1.7.0

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2021.4.1

Kanál operátoru
`v1.7`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Red Hat OpenShift Container Platform 4.6 a vyšší

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.8 a vyšší (kanál v3)

Novinky

- Přidává IBM MQ 9.2.4 jako vydání continuous delivery

IBM MQ Operator 1.6.0

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2021.2.1

Kanál operátoru
`v1.6`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Red Hat OpenShift Container Platform 4.6 a vyšší

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.7 a vyšší (kanál v3)

Novinky

- Přidává produkt IBM MQ 9.2.3 jako souvislé vydání doručení (amd64 pouze pro IBM Cloud Pak for Integration 2021.2.1; amd64 nebo s390x při použití licence IBM MQ)
- Nový typ dostupnosti pro správce front: Nativní vysoká dostupnost. K dispozici pro produkční použití jako součást IBM Cloud Pak for Integration 2021.2.1.

Změny

- IBM MQ Operator 1.6 a vyšší používá IBM Container Registry namísto Docker Hub. To znamená, že je třeba použít produkt `CatalogSource` z produktu `icr.io`. Viz téma "Instalace a odinstalace produktu IBM MQ Operator v systému Red Hat OpenShift" na stránce 58.
- Aktualizace posouvání Nativní vysoké dostupnosti již nečeká na to, aby byla replika synchronizovaná před přechodem na další repliku.
- Opraví problém s afinitou Nativní vysoké dostupnosti na OCP 4.7 a vyšším.
- Opraví problém při použití certifikátů podepsaných CA s Nativním vysokou dostupností.

IBM MQ Operator 1.5.0

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2021.1.1

Kanál operátoru
`v1.5`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Red Hat OpenShift Container Platform 4.6 a vyšší

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.7 a vyšší (kanál v3)

Novinky

- Přidává produkt IBM MQ 9.2.2 jako souvislé vydání doručení (amd64 pouze pro IBM Cloud Pak for Integration 2021.1.1; amd64 nebo s390x při použití licence IBM MQ)
- Nový typ dostupnosti pro správce front: Nativní vysoká dostupnost. K dispozici pouze pro účely hodnocení, jako součást IBM Cloud Pak for Integration 2021.1.1.
- Integrace s metrikami produktu Red Hat OpenShift Container Platform Cluster Monitoring for Prometheus poskytnutím prostředku `ServiceMonitor`

Změny

- Při vytváření správce front již produkt IBM Licensing Operator není vytvořen.
- Aktualizace správců front s více instancemi jsou nyní obsluhovány postupně. V rámci této změny byla představena spouštěcí sonda Kybernetes, která ovlivňuje hodnoty použité při konfiguraci sondy živosti. Spouštěcí sonda se spustí okamžitě, potom čeká na úspěšné spuštění správce front. Když uplyne čas spouštěcí sondy během této doby čekání, spustí se sondy připravenosti a živosti. Pokud jste dříve spustili správce front, jehož spuštění bylo pomalé, mohli jste zvýšit nastavení `initialDelaySeconds` v testu aktivity. Pokud jste to provedli, měli byste nyní vrátit `initialDelaySeconds` na dřívějším nastavením.
- `CustomResourceDefinition` je upgradován z `apiextensions.k8s.io/v1beta1` na `apiextensions.k8s.io/v1`.

Známé problémy a omezení

- Vyžaduje IBM Cloud Pak foundational services 3.7 obsahující nekompatibilní změnu v komponentě Identity and Access Management (IAM). Pokud máte nějaké správce front, kteří používají licenci IBM Cloud Pak for Integration, pak po tomto upgradu bude pro přístup k webové konzole vyžadován restart správce front a také se zobrazí další chyby přihlášení do webové konzoly. Tyto chyby můžete opravit upgradem na nejnovější hodnotu `.spec.version` pro zvolenou verzi produktu IBM MQ po dokončení upgradu operátora.
- Pokud provádíte upgrade verze MQ, nedojde k automatickému spuštění průběžné aktualizace. Musíte ručně odstranit pody.

IBM MQ Operator 1.4.0

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.4.1 (IBM MQ Operator 1.4.0 je vydání CD a není vhodné pro Extended Update Support)

Kanál operátoru
`v1.4`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Red Hat OpenShift Container Platform 4.6 a vyšší

Novinky

- Přidá IBM MQ 9.2.1 jako vydání continuous delivery
- Nyní můžete zabránit vytvoření výchozí trasy správce front nastavením `.spec.queueManager.route.enabled` na `false`.

Známé problémy a omezení

- Když aktualizujete `QueueManager` s typem dostupnosti `MultiInstance`, budou oba pody okamžitě odstraněny. Oba by měli být rychle restartovány Red Hat OpenShift Container Platform.

IBM MQ Operator 1.3.8 (EUS)

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.4.1

Kanál operátoru
`v1.3-eus`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Pouze Red Hat OpenShift Container Platform 4.6

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.6 (kanál `stable-v1`)

Novinky

- Přidá nový operand verze 9.2.0.6-r3-eus.
- Zranitelnosti, které jsou řešeny, jsou podrobně popsány na této Vývěsce o zabezpečení.

IBM MQ Operator 1.3.7 (EUS)

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.4.1

Kanál operátoru
`v1.3-eus`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Pouze Red Hat OpenShift Container Platform 4.6

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.6 (kanál `stable-v1`)

Novinky

- Přidá nový operand verze 9.2.0.6-r2-eus.
- Zranitelnosti, které jsou řešeny, jsou podrobně popsány na této Vývěsce o zabezpečení.

IBM MQ Operator 1.3.6 (EUS)

EUS

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.4.1

Kanál operátoru
`v1.3-eus`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Pouze Red Hat OpenShift Container Platform 4.6

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.6 (kanál `stable-v1`)

Novinky

- Přidá nový operand verze 9.2.0.6-r1-eus.
- Zranitelnosti, které jsou řešeny, jsou podrobně popsány na této Vývěsce o zabezpečení.

IBM MQ Operator 1.3.5 (EUS)

EUS

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.4.1

Kanál operátoru
`v1.3-eus`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Pouze Red Hat OpenShift Container Platform 4.6

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.6 (kanál `stable-v1`)

Novinky

- Přidá nový operand verze 9.2.0.5-r3-eus.

- Zranitelnosti, které jsou řešeny, jsou podrobně popsány na této Vývěsce o zabezpečení.

IBM MQ Operator 1.3.4 (EUS)

EUS

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.4.1

Kanál operátoru
`v1.3-eus`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Pouze Red Hat OpenShift Container Platform 4.6

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.6 (kanál `stable-v1`)

Novinky

- Přidá nový operand verze 9.2.0.5-r2-eus
- Zranitelnosti, které jsou řešeny, jsou podrobně popsány na této Vývěsce o zabezpečení.

IBM MQ Operator 1.3.3 (EUS)

EUS

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.4.1

Kanál operátoru
`v1.3-eus`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Pouze Red Hat OpenShift Container Platform 4.6

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.6 (kanál `stable-v1`)

Novinky

- Přidá nový operand verze 9.2.0.5-r1-eus
- Zranitelnosti, které jsou řešeny, jsou podrobně popsány na této Vývěsce o zabezpečení.

IBM MQ Operator 1.3.2 (EUS)

EUS

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.4.1

Kanál operátoru
`v1.3-eus`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Pouze Red Hat OpenShift Container Platform 4.6

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.6 (kanál `stable-v1`)

Novinky

- Přidá novou verzi operandu 9.2.0.4-r1-eus

IBM MQ Operator 1.3.1 (EUS)

EUS

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.4.1

Kanál operátoru
`v1.3-eus`

Přípustné hodnoty pro `.spec.version`
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

Verze Red Hat OpenShift Container Platform
Pouze Red Hat OpenShift Container Platform 4.6

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.6 (kanál `stable-v1`)

Novinky

- Přidá novou verzi operandu 9.2.0.2-r1-eus

IBM MQ Operator 1.3.0 (EUS)

EUS

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.4.1

Kanál operátoru
`v1.3-eus`

Přípustné hodnoty pro `.spec.version`
9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2, 9.2.0.1-r1-eus

Verze Red Hat OpenShift Container Platform
Pouze Red Hat OpenShift Container Platform 4.6

Verze IBM Cloud Pak foundational services
IBM Cloud Pak foundational services 3.6 (kanál `stable-v1`)

Novinky

- Podpora rozšířené aktualizace (EUS) je nabízena pro pole `.spec.version` končící řetězcem `-eus`, pokud používáte licenci produktu IBM Cloud Pak for Integration.
- Přidává nový způsob nastavení popisků a anotací na prostředku `QueueManager` pomocí `.spec.labels` a `.spec.annotations`

Změny

- Lepší ošetření chyb při pokusu o změnu z jedné instance do více instancí.
- Zlepšení způsobu vykreslování vlastností produktu `QueueManager` v IBM Cloud Pak for Integration Platform Navigator a "pohledu Formulář" webové konzoly Red Hat OpenShift Container Platform
- Opravuje výchozí metriku licence při použití licence produktu IBM Cloud Pak for Integration, aby byla `VirtualProcessorCore`
- Opravuje kartu **Prostředky** pro `QueueManager` ve webové konzole Red Hat OpenShift Container Platform, která nyní správně zobrazuje prostředky spravované produktem IBM MQ Operator pro tohoto správce front.

Známé problémy a omezení

- Když aktualizujete `QueueManager` s typem dostupnosti `MultiInstance`, budou oba pody okamžitě odstraněny. Oba by měli být rychle restartovány Red Hat OpenShift Container Platform.

IBM MQ Operator 1.2.0

CD

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.3.1

Kanál operátoru
`v1.2`

Přípustné hodnoty pro `.spec.version`
9.1.5.0-r2, 9.2.0.0-r1, 9.2.0.0-r2

Verze Red Hat OpenShift Container Platform
Red Hat OpenShift Container Platform 4.4 a vyšší

Novinky

- Přidá podporu pro systém z/Linux.
- Přidává podrobnější stavové podmínky do prostředku `QueueManager`. Další informace viz “Stavové podmínky pro správce front (mq.ibm.com/v1beta1)” na stránce 142.
- Přidá další běhové kontroly, aby se zabránilo použití neplatných tříd úložiště. Další informace naleznete v tématu “Zakázání běhových kontrol webhooku” na stránce 112
- Zjednodušuje zkušenost pro správce front s více instancemi: lze jej nyní zvolit pouze s jednou vlastností (`.spec.queueManager.availability.type`) v prostředku `QueueManager`.
- Zjednodušuje výběr jiné než výchozí třídy úložiště, zavedením vlastnosti `.spec.queueManager.storage.defaultClass` v prostředku `QueueManager`.

Změny

- Zlepšení způsobu vykreslování vlastností produktu `QueueManager` v IBM Cloud Pak for Integration Platform Navigator a "pohledu Formulář" webové konzoly Red Hat OpenShift Container Platform
- Je-li k dispozici upgradovaná verze správce front, bude nyní označena v IBM Cloud Pak for Integration Platform Navigator.

IBM MQ Operator 1.1.0

CD

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.2.1

Kanál operátoru
`v1.1`

Přípustné hodnoty pro `.spec.version`
9.1.5.0-r2, 9.2.0.0-r1

Verze Red Hat OpenShift Container Platform
Red Hat OpenShift Container Platform 4.4 a vyšší

Novinky

- Přidá IBM MQ Advanced 9.2.0 jako vydání continuous delivery
- Přidá funkci k určení informací INI a MQSC v ConfigMap nebo Secret.
- Povoluje navigátor schématu při použití webové konzoly Red Hat OpenShift Container Platform.

Změny

- Opravuje problém se zásadou sítě, dopad na Red Hat OpenShift na IBM Cloud
- Zlepšení pro ověření webového háčku, aby se zabránilo neplatným kombinacím nastavení v prostředcích `QueueManager`

IBM MQ Operator 1.0.0

CD

Verze IBM Cloud Pak for Integration
IBM Cloud Pak for Integration 2020.2.1

Kanál operátoru
`v1.0`

Přípustné hodnoty pro `.spec.version`
9.1.5.0-r2

Verze Red Hat OpenShift Container Platform
Red Hat OpenShift Container Platform 4.4 a vyšší

Novinky

- Počáteční verze operátoru, uvedení rozhraní API `mq.ibm.com/v1beta1`

Kontejnerové obrazy správce front pro použití s IBM MQ Operator

9.2.5.0-r3

CD

Požadovaná verze operátoru
1.8.2 nebo vyšší

Podporované architektury
`amd64`, `s390x`

Obrázky

- `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`

Novinky

- Novinky v IBM MQ 9.2.5

Změny

- Co se změnilo v IBM MQ 9.2.5
- Založeno na Red Hat Universal Base Image 8.6-751

9.2.5.0-r2

CD

Požadovaná verze operátoru
1.8.1 nebo vyšší

Podporované architektury
`amd64`, `s390x`

Obrázky

- `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`

Novinky

- Novinky v IBM MQ 9.2.5

Změny

- Co se změnilo v IBM MQ 9.2.5
- Založeno na Red Hat Universal Base Image 8.5-240.1648458092

9.2.5.0-r1

Požadovaná verze operátoru
1.8.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

- `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`

Novinky

- Novinky v IBM MQ 9.2.5

Změny

- Co se změnilo v IBM MQ 9.2.5
- Nyní byla z produktu IBM MQ Console odstraněna neplatná volba `Vzdálení správci front`
- Založeno na Red Hat Universal Base Image 8.5-240

9.2.4.0-r1

Požadovaná verze operátoru
1.7.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

- `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`

Novinky

- Novinky v IBM MQ 9.2.4

Změny

- Co se změnilo v IBM MQ 9.2.4
- Založeno na Red Hat Universal Base Image 8.5-204

9.2.3.0-r1

Požadovaná verze operátoru
1.6.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

- `cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.3.0-r1` (pouze amd64)
- `cp.icr.io/cp/ibm-mqadvanced-server:9.2.3.0-r1`
- `docker.io/ibmcom/mq:9.2.3.0-r1`

Novinky

- Novinky v IBM MQ 9.2.3
- Podpora pro Nativní vysoké dostupnosti produktu MQ pro produkční použití při použití s licencí IBM Cloud Pak for Integration. Nezapomeňte, že správci front používající Nativní vysokou dostupnost v rámci licence pro vyhodnocení s produktem IBM MQ 9.2.2 nelze upgradovat na verzi 9.2.3. Testovací období bylo ukončeno.

Změny

- Co se změnilo v IBM MQ 9.2.3
- Založeno na Red Hat Universal Base Image 8.4-205

9.2.2.0-r1

CD

Požadovaná verze operátoru
1.5.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

- `cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.2.0-r1` (pouze amd64)
- `cp.icr.io/cp/ibm-mqadvanced-server:9.2.2.0-r1`
- `docker.io/ibmcom/mq:9.2.2.0-r1`

Novinky

- Novinky v IBM MQ 9.2.2
- Podpora nativní vysoké dostupnosti produktu MQ pro účely vyhodnocení, při použití s licencí IBM Cloud Pak for Integration

Změny

- Co se změnilo v IBM MQ 9.2.2
- Odstraněn problém způsobující FDC při vypínání správce front IBM MQ Advanced for Developers
- Založeno na Red Hat Universal Base Image 8.3-291

9.2.1.0-r2

CD

Požadovaná verze operátoru
1.5.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

- `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`

Změny

- Opraví problém s jednotným přihlášením pomocí produktu IBM Cloud Pak foundational services 3.7 a vyšší.
- Založeno na Red Hat Universal Base Image 8.3-291

9.2.1.0-r1

CD

Požadovaná verze operátoru
1.4.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

- `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`

Novinky

- Novinky v IBM MQ 9.2.1
- Informace o připojení pro výchozí trasu jsou k dispozici ve webové konzole MQ.

Změny

- Co se změnilo v IBM MQ 9.2.1
- Založeno na Red Hat Universal Base Image 8.3-230

9.2.0.6-r3-eus

EUS

Požadovaná verze operátoru
1.3.8 a budoucí opravné sady

Podporované architektury
amd64, s390x

Obrázky

- `cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.6-r3-eus`

Změny

- Obsahuje IBM MQ 9.2.0 Fix Pack 6. Další informace naleznete v tématu Seznam oprav pro produkt IBM MQ verze 9.2 LTS.
- Založeno na Red Hat Universal Base Image 8.6-941.

9.2.0.6-r2-eus

EUS

Požadovaná verze operátoru
1.3.7 a budoucí opravné sady

Podporované architektury
amd64, s390x

Obrázky

- `cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.6-r2-eus`

Změny

- Obsahuje IBM MQ 9.2.0 Fix Pack 6. Další informace naleznete v tématu Seznam oprav pro produkt IBM MQ verze 9.2 LTS.
- Založeno na Red Hat Universal Base Image 8.6-902.

9.2.0.6-r1-eus

EUS

Požadovaná verze operátoru
1.3.6 a budoucí opravné sady

Podporované architektury
amd64, s390x

Obrázky

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

Změny

- Obsahuje IBM MQ 9.2.0 Fix Pack 6. Další informace naleznete v tématu Seznam oprav pro produkt IBM MQ verze 9.2 LTS.
- Založeno na Red Hat Universal Base Image 8.6-854.

9.2.0.5-r3-eus

EUS

Požadovaná verze operátoru
1.3.5 a budoucí opravné sady

Podporované architektury
amd64, s390x

Obrázky

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

Změny

- Obsahuje IBM MQ 9.2.0 Fix Pack 5. Další informace najdete v tématu Co se změnilo v IBM MQ 9.2.0 Fix Pack 5 , a Seznam oprav pro produkt IBM MQ verze 9.2 LTS.
- Založeno na Red Hat Universal Base Image 8.6-751.1655117800.

9.2.0.5-r2-eus

EUS

Požadovaná verze operátoru
1.3.4 a budoucí opravné sady

Podporované architektury
amd64, s390x

Obrázky

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

Změny

- Obsahuje IBM MQ 9.2.0 Fix Pack 5. Další informace viz Co se změnilo v IBM MQ 9.2.0 Fix Pack 5 a Seznam oprav pro produkt IBM MQ verze 9.2 LTS
- Založeno na Red Hat Universal Base Image 8.6-751

9.2.0.5-r1-eus

EUS

Požadovaná verze operátoru
1.3.3 a budoucí opravné sady

Podporované architektury
amd64, s390x

Obrázky

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

Změny

- Obsahuje IBM MQ 9.2.0 Fix Pack 5. Další informace viz Co se změnilo v IBM MQ 9.2.0 Fix Pack 5 a Seznam oprav pro produkt IBM MQ verze 9.2 LTS
- Založeno na Red Hat Universal Base Image 8.5-240.1648458092

9.2.0.4-r1-eus

EUS

Požadovaná verze operátoru
1.3.2 a budoucí opravné sady

Podporované architektury
amd64, s390x

Obrázky

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

Změny

- Obsahuje IBM MQ 9.2.0 Fix Pack 4. Další informace viz Co se změnilo v IBM MQ 9.2.0 Fix Pack 4 a Seznam oprav pro produkt IBM MQ verze 9.2 LTS
- Založeno na Red Hat Universal Base Image 8.5-204

9.2.0.2-r2-eus

EUS

Požadovaná verze operátoru
1.6.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

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

Změny

- Opraví problém s jednotným přihlášením pomocí produktu IBM Cloud Pak foundational services 3.7 a vyšší, což je zapotřebí pouze při migraci z vydání EUS na vydání CD.
- Založeno na Red Hat Universal Base Image 8.4-200.1622548483

9.2.0.2-r1-eus

EUS

Požadovaná verze operátoru
1.3.1 a budoucí opravné sady ; 1.6.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

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

Změny

- Integrace produktu Operations Dashboard používá agenta trasování a kolektor verze 1.0.8
- Obsahuje IBM MQ 9.2.0 Fix Pack 2. Další informace viz Co se změnilo v IBM MQ 9.2.0 Fix Pack 2 a Seznam oprav pro produkt IBM MQ verze 9.2 LTS
- Založeno na Red Hat Universal Base Image 8.4-200.1622548483

9.2.0.1-r1-eus

Požadovaná verze operátoru
1.3.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

- `cp.icr.io/cp/ibm-mqadvanced-server-integration:9.2.0.1-r1-eus`

Novinky

- K dispozici pouze v případě použití licence IBM Cloud Pak for Integration.
- Extended Update Support (EUS) je k dispozici při použití IBM MQ Operator 1.3.x a IBM Common Services 3.6, na verzi Red Hat OpenShift Container Platform 4.6

Změny

- Obsahuje IBM MQ 9.2.0 Fix Pack 1. Další informace viz Co se změnilo v IBM MQ 9.2.0 Fix Pack 1 a Seznam oprav pro produkt IBM MQ verze 9.2 LTS
- Založeno na Red Hat Universal Base Image 8.3-201
- Opravuje problém s testem aktivity (`chkmqhealthy`) a testem připravenosti (`chkmqready`) při spuštění pod `SecurityContextContraints`, které umožňují eskalaci oprávnění.

9.2.0.0-r3

Požadovaná verze operátoru
1.5.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

- `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`

Změny

- Založeno na Red Hat Universal Base Image 8.3-291

9.2.0.0-r2

Požadovaná verze operátoru
1.2.0 nebo vyšší

Podporované architektury
amd64, s390x

Obrázky

- `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`

Novinky

- Nyní k dispozici v systému z/Linux

Změny

- Založeno na Red Hat Universal Base Image 8.2-349

9.2.0.0-r1

CD

Požadovaná verze operátoru
1.1.0 nebo vyšší

Podporované architektury
amd64

Obrázky

- `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`

Novinky

- Novinky v IBM MQ 9.2.0

Změny

- Co se změnilo v IBM MQ 9.2.0
- Používá argument `-ic` pro `crtmqm` k automatickému použití souborů MQSC. Nahrazuje předchozí použití příkazů `runmqsc`
- Založeno na Red Hat Universal Base Image 8.2-301.1593113563

9.1.5.0-r2

CD

Požadovaná verze operátoru
1.0.0 nebo vyšší

Podporované architektury
amd64

Obrázky

- `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`

Změny

- Založeno na Red Hat Universal Base Image 8.2-267

OpenShift V 9.2.1 CD EUS Migrace IBM MQ do produktu IBM Cloud Pak for Integration

Tato sada témat popisuje klíčové kroky k migraci stávajícího správce front IBM MQ do prostředí kontejneru pomocí IBM MQ Operator v IBM Cloud Pak for Integration.

Informace o této úloze

Klienti, kteří implementují produkt IBM MQ na serveru Red Hat OpenShift, lze rozdělit do následujících scénářů:

1. Vytvoření nové implementace IBM MQ v Red Hat OpenShift pro nové aplikace.
2. Rozšíření sítě IBM MQ do Red Hat OpenShift pro nové aplikace v Red Hat OpenShift.
3. Přesunutí implementace IBM MQ do Red Hat OpenShift bude pokračovat v podpoře existujících aplikací.

To platí pouze pro scénář 3, který potřebujete k migraci konfigurace produktu IBM MQ. Ostatní scénáře se považují za nové implementace.

Tato sada témat se zaměřuje na scénář 3 a popisuje klíčové kroky k migraci stávajícího správce front IBM MQ do prostředí kontejneru s použitím IBM MQ Operator. Vzhledem k flexibilitě a rozsáhlému použití produktu IBM MQ existuje několik volitelných kroků. Každý z nich obsahuje sekci "Musím to udělat?". Ověření, že byste měli během migrace ušetřit čas.

Musíte také zvážit, jaká data se mají migrovat:

1. Migrovat produkt IBM MQ se stejnou konfigurací, ale bez jakýchkoli existujících zpráv ve frontě.
2. Migrovat produkt IBM MQ se stejnou konfigurací a existujícími zprávami.

Typická verze pro migraci verzí může použít jeden z přístupů. V typickém správci front produktu IBM MQ v místě migrace existuje několik málo zpráv, pokud jsou nějaké uložené ve frontách, díky čemuž je volba 1 vhodná pro mnoho případů. V případě migrace na platformu kontejneru je volba 1 ještě běžnější, aby se snížila složitost migrace a umožnil tzv. blue-green deployment (modrozelené nasazení). Proto se pokyny zaměřují na tento scénář.

Cílem tohoto scénáře je vytvořit správce front v prostředí kontejneru, který odpovídá definici existujícího správce front. To umožňuje jednoduše překonfigurovat existující aplikace připojené k síti tak, aby ukazovaly na nového správce front, aniž by se změnila jakákoli jiná konfigurace nebo logika aplikace.

V rámci této migrace generujete více konfiguračních souborů, které mají být použity pro nového správce front. Chcete-li zjednodušit správu těchto souborů, měli byste vytvořit adresář a soubory vygenerovat do tohoto adresáře.

Postup

1. “Kontrola, zda jsou k dispozici požadované funkce” na stránce 37
2. “Extrakce konfigurace správce front” na stránce 37
3. Volitelné: “Volitelné: Extrakce a získání klíčů a certifikátů správce front” na stránce 38
4. Volitelné: “Volitelné: Konfigurace protokolu LDAP” na stránce 40
5. Volitelné: “Volitelné: Změna adres IP a názvů hostitelů v konfiguraci produktu IBM MQ” na stránce 47
6. “Aktualizace konfigurace správce front pro prostředí kontejnerů” na stránce 49
7. “Výběr cílové architektury vysoké dostupnosti pro produkt IBM MQ spuštěný v kontejnerech” na stránce 51
8. “Vytvoření prostředků pro správce front” na stránce 52
9. “Vytvoření nového správce front v Red Hat OpenShift” na stránce 53
10. “Ověření implementace nového kontejneru” na stránce 57

OpenShift V 9.2.1 CD EUS Kontrola, zda jsou k dispozici požadované funkce

IBM MQ Operator nezahrnuje všechny dostupné funkce v rámci IBM MQ Advanced a vy musíte ověřit, že tyto funkce nejsou vyžadovány. Jiné funkce jsou částečně podporovány a lze je překonfigurovat tak, aby odpovídaly obsahu, který je k dispozici v kontejneru.

Než začnete

Jedná se o první krok v "Migrace IBM MQ do produktu IBM Cloud Pak for Integration" na stránce 36.

Postup

1. Ověřte, že obraz cílového kontejneru obsahuje všechny požadované funkce.

   Nejnovější informace viz "Zvolení, jak se má produkt IBM MQ používat v kontejnerech" na stránce 5.
2. IBM MQ Operator má jeden provozní port IBM MQ, známý jako listener. Máte-li více modulů listener, zjednodušte to na použití jednoho modulu listener v kontejneru. Vzhledem k tomu, že se nejedná o běžný scénář, tato úprava není podrobně dokumentována.
3. Jsou-li použity uživatelské procedury IBM MQ, proveďte jejich migraci do kontejneru, a to vrstvením v binárních souborech uživatelské procedury IBM MQ. Jedná se o scénář rozšířené migrace, a proto zde není zahrnut. Informace o postupu viz "Sestavení obrazu pomocí vlastních souborů MQSC a INI s použitím rozhraní CLI Red Hat OpenShift" na stránce 110.
4. Pokud váš systém IBM MQ zahrnuje vysokou dostupnost, zkontrolujte dostupné volby.

   Viz "Vysoká dostupnost pro IBM MQ v kontejnerech" na stránce 16.

Jak pokračovat dále

Nyní jste připraveni extrahovat konfiguraci správce front.

OpenShift V 9.2.1 CD EUS Extrakce konfigurace správce front

Většina konfigurace je přenositelná mezi správci front. Například věci, se kterými aplikace interaguje, jako jsou definice front, témat a kanálů. Tato úloha slouží k extrakci konfigurace z existujícího správce front IBM MQ.

Než začnete

Tato úloha předpokládá, že jste zkontrolovali dostupnost požadovaných funkcí.

Postup

1. Přihlaste se k počítači s existující instalací IBM MQ.
2. Zazálohujte konfiguraci.

   Spusťte tento příkaz:

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

   Poznámky k používání pro tento příkaz:

   - Tento příkaz ukládá zálohu do adresáře `tmp`. Zálohování můžete uložit do jiného umístění, ale tento scénář předpokládá adresář `tmp` pro následné příkazy.
   - *QMGR_NAME* nahraďte názvem správce front z vašeho prostředí. Pokud si nejste jisti hodnotou, spusťte příkaz **dspmq** a prohlédněte si dostupné správce front na počítači. Zde je uveden příklad výstupu příkazu **dspmq** pro správce front s názvem `qm1`:

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

Příkaz **dspmq** vyžaduje spuštění správce front IBM MQ, jinak se zobrazí následující chyba:

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

V případě potřeby spusťte správce front spuštěním následujícího příkazu:

```
strmqm QMGR_NAME
```

Jak pokračovat dále

Nyní jste připraveni extrahovat a získat klíče a certifikáty správce front.

OpenShift V 9.2.1 CD EUS Volitelné: Extrakce a získání klíčů a certifikátů správce front

Produkt IBM MQ lze konfigurovat pomocí TLS k zašifrování provozu v rámci správce front. Tato úloha slouží k ověření, zda správce front používá TLS, k extrakci klíčů a certifikátů a ke konfiguraci TLS v migrovaném správci front.

Než začnete

Tato úloha předpokládá, že jste extrahovali konfiguraci správce front.

Informace o této úloze

Musím to udělat?

IBM MQ lze konfigurovat k zašifrování provozu v rámci správce front. Toto šifrování je dokončeno pomocí úložiště klíčů konfigurovaného ve správci front. Kanály IBM MQ potom povolí komunikaci TLS. Pokud si nejste jisti, zda je ve vašem prostředí nakonfigurována, spusťte následující příkaz k ověření:

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

Nejsou-li nalezeny žádné výsledky, TLS se nepoužívá. To však neznamená, že TLS by nemělo být nakonfigurováno v migrovaném správci front. Existuje řada důvodů, proč byste mohli chtít změnit toto chování:

- Přístup zabezpečení v prostředí Red Hat OpenShift by měl být vylepšen ve srovnání s předchozím prostředím.
- Potřebujete-li přístup k migrovanému správci front mimo prostředí Red Hat OpenShift, je třeba, aby TLS prošlo přes trasu Red Hat OpenShift.

Postup

1. Extrahujte všechny důvěryhodné certifikáty z existujícího úložiště.

   Používáte-li se aktuálně TLS ve správci front, může mít správce front uložený počet důvěryhodných certifikátů. Je třeba je extrahovat a zkopírovat do nového správce front. Proveďte jeden z následujících volitelných kroků:

   - Chcete-li zefektivnit extrakci certifikátů, spusťte následující skript v lokálním systému:

     ```
     #!/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
```

Při spuštění skriptu zadejte umístění zálohy IBM MQ jako argument a certifikáty jsou extrahovány. Pokud je například skript nazván `extractCert.sh` a záloha IBM MQ se nachází v `/tmp/backup.mqsc`, spusťte následující příkaz:

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

- Volitelně spusťte následující příkazy v uvedeném pořadí:

  a. Identifikujte umístění úložiště TLS:

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

  Ukázkový výstup:

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

  Kde úložiště klíčů se nachází v umístění `/run/runmqserver/tls/key.kdb`

  b. Na základě těchto informací o umístění se dotázat na úložiště klíčů, a určit tak uložené certifikáty:

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

  Ukázkový výstup:

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

  c. Extrahujte každý z uvedených certifikátů. Proveďte to spuštěním následujícího příkazu:

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

  V předchozích ukázkách se tento stav rovnal následujícím:

  ```
  runmqckm -cert -extract -db /run/runmqserver/tls/key.kdb -label "CN=cs-ca-
  certificate,O=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. Získejte nový klíč a certifikát pro správce front.

   Chcete-li nakonfigurovat TLS v migrovaném správci front, vygenerujte nový klíč a certifikát. To se pak použije během implementace. V mnoha organizacích to znamená kontaktovat svůj bezpečnostní tým a požádat o klíč a certifikát. V některých organizacích tato volba není k dispozici a používají se certifikáty podepsané držitelem.

   Následující příklad generuje certifikát podepsaný držitelem, kde je vypršení platnosti nastaveno na 10 let:

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

Vytvoří se dva nové soubory:

- `qmgr.key` je soukromý klíč pro správce front.
- `qmgr.crt` je veřejný certifikát

Jak pokračovat dále

Nyní jste připraveni konfigurovat LDAP.

OpenShift V 9.2.1 CD EUS Volitelné: Konfigurace protokolu LDAP

IBM MQ Operator může být konfigurován tak, aby používal několik různých přístupů zabezpečení. Protokol LDAP je obvykle nejefektivnější pro podnikovou implementaci, a pro tento scénář migrace se používá LDAP.

Než začnete

Tato úloha předpokládá, že jste extrahovali konfiguraci správce front pro prostředí kontejneru.

Informace o této úloze

Musím to udělat?

Používáte-li již LDAP pro ověření a autorizaci, pak se nepožadují žádné změny.

Pokud si nejste jisti, zda se LDAP používá, spusťte následující příkaz:

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

Ukázkový výstup:

```
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) +
*  ALTDATE(2020-11-26) +
*  ALTTIME(15.44.38) +
   REPLACE
```

Ve výstupu jsou dva atributy, které jsou zvláště zajímavé:

AUTHTYPE
Pokud je tato hodnota `IDPWLDAP`, potom pro ověření používáte LDAP.

Pokud je hodnota prázdná, nebo jiná hodnota, pak LDAP není nakonfigurováno. V takovém případě zkontrolujte atribut `AUTHORMD`, abyste zjistili, zda se uživatelé LDAP používali pro autorizaci.

AUTHORMD
Pokud je tato hodnota `OS`, potom pro autorizaci nepoužíváte LDAP.

Chcete-li upravit autorizaci a ověření pro použití LDAP, proveďte následující úlohy:

Postup

1. Aktualizujte zálohu produktu IBM MQ pro server LDAP.
2. Aktualizujte zálohu produktu IBM MQ pro informace o autorizaci LDAP.

Část LDAP 1: Aktualizace zálohy IBM MQ pro server LDAP

Úplný popis, jak nastavit LDAP, je mimo rozsah tohoto scénáře. Toto téma poskytuje souhrn procesu, ukázky a odkazy na další informace.

Než začnete

Tato úloha předpokládá, že jste extrahovali konfiguraci správce front pro prostředí kontejneru.

Informace o této úloze

Musím to udělat?

Používáte-li již LDAP pro ověření a autorizaci, pak se nepožadují žádné změny. Pokud si nejste jisti, zda se LDAP používá, podívejte se na téma “Volitelné: Konfigurace protokolu LDAP” na stránce 40.

Pro nastavení serveru LDAP jsou k dispozici dvě části:

1. Definujte konfiguraci LDAP.
2. Přidružte konfiguraci LDAP k definici správce front.

Další informace, které vám pomohou s touto konfigurací:

- Přehled úložiště uživatelů
- Referenční příručka k příkazu AUTHINFO

Postup

1. Definujte konfiguraci LDAP.

   Upravte soubor `backup.mqsc`, abyste definovali nový objekt **AUTHINFO** pro systém LDAP. Příklad:

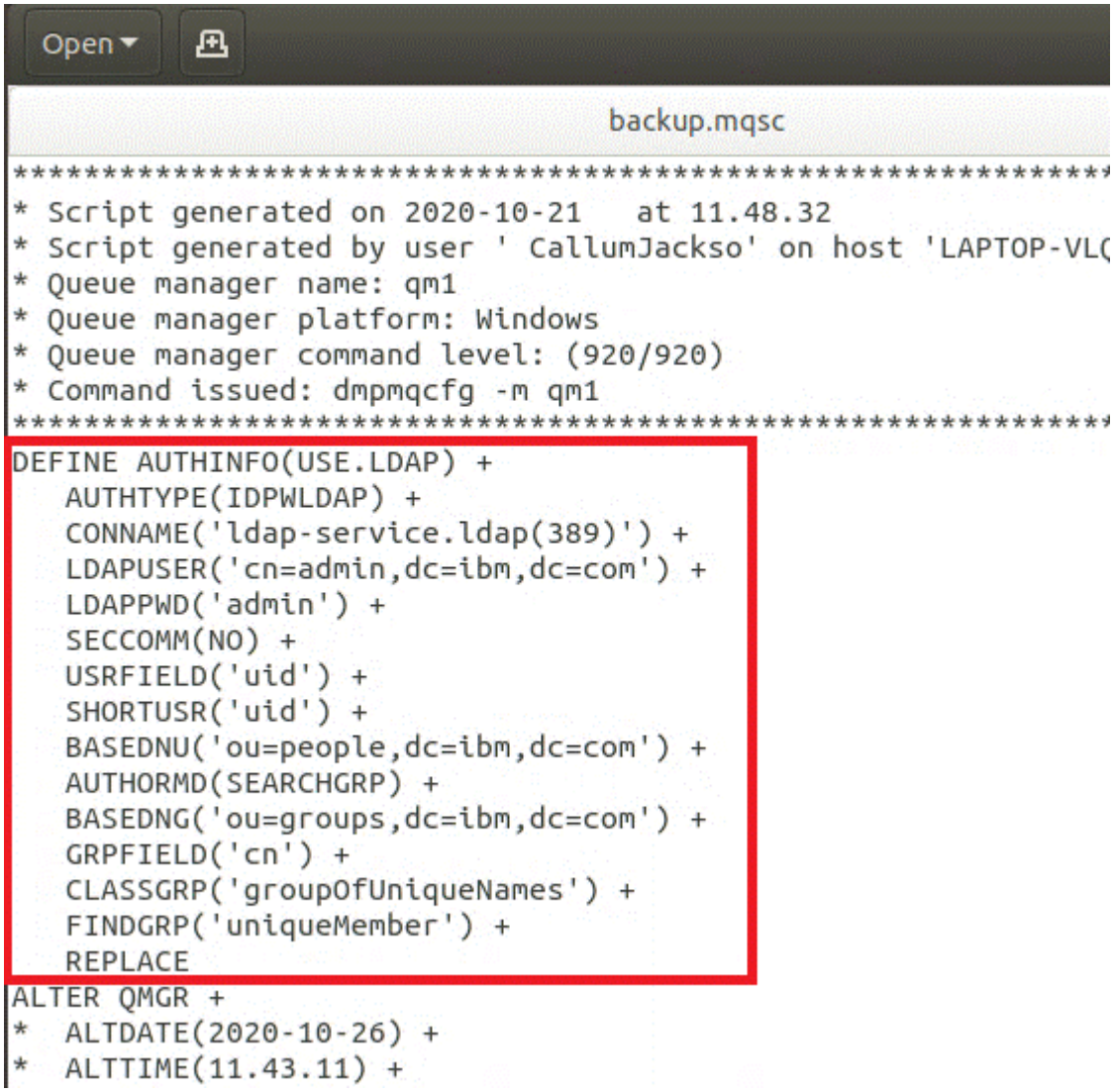
```
DEFINE AUTHINFO(USE.LDAP) +
   AUTHTYPE(IDPWLDAP) +
   CONNAME('ldap-service.ldap(389)') +
   LDAPUSER('cn=admin,dc=ibm,dc=com') +
   LDAPPWD('admin') +
   SECCOMM(NO) +
   USRFIELD('uid') +
   SHORTUSR('uid') +
   BASEDNU('ou=people,dc=ibm,dc=com') +
   AUTHORMD(SEARCHGRP) +
   BASEDNG('ou=groups,dc=ibm,dc=com')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

   kde:

   - **CONNAME** je název hostitele a port odpovídající serveru LDAP. Pokud pro odolnost existuje více adres, je možné je konfigurovat pomocí seznamu odděleného čárkami.
   - **LDAPUSER** je rozlišující název odpovídající uživateli, který produkt IBM MQ používá při připojování k LDAP, aby se dotázal na záznamy uživatelů.
   - **LDAPPWD** je heslo, které odpovídá uživateli **LDAPUSER**.
   - Parametr **SECCOM** určuje, zda by komunikace se serverem LDAP měla používat zabezpečení TLS. Možné hodnoty jsou:
     - YES: Používá se TLS a certifikát je prezentován serverem IBM MQ.

  - ANON: Používá se TLS bez toho, že by byl certifikát prezentován serverem IBM MQ.
  - NO: TLS se nepoužívá během připojení.
- **USRFIELD** určuje pole v záznamu LDAP, vůči němuž je prezentované jméno uživatele porovnáváno.
- **SHORTUSR** je pole v rámci záznamu LDAP, jehož délka nepřesahuje 12 znaků. Hodnota v tomto poli je deklarovaná identita, je-li ověření úspěšné.
- **BASEDNU** je základní rozlišující název, který by měl být použit pro vyhledávání LDAP.
- **BASEDNG** je základní rozlišující název pro skupiny v rámci LDAP.
- **AUTHORMD** definuje mechanizmus používaný k vyřešení členství ve skupinách pro daného uživatele. K dispozici jsou čtyři volby:
  - OS: Dotazování na operační systém pro skupiny přidružené ke krátkému názvu.
  - SEARCHGRP: Vyhledá položky skupiny v LDAP pro ověřeného uživatele.
  - SEARCHUSR: Vyhledá informace o členství ve skupinách v záznamu ověřeného uživatele.
  - SRCHGRPSN: Vyhledá položky skupiny v LDAP pro krátké jméno uživatele (definované pomocí pole SHORTUSR) pro ověřené uživatele.
- **GRPFIELD** je atribut v rámci záznamu skupiny LDAP, který odpovídá jednoduchému názvu. Je-li uveden, lze jej použít pro definování záznamů autorizace.
- **CLASSUSR** je třída objektů LDAP, která odpovídá uživateli.
- **CLASSGRP** je třída objektů LDAP, která odpovídá skupině.
- **FINDGRP** je atribut v rámci záznamu LDAP, který odpovídá členství ve skupinách.

Nová položka může být umístěna kdekoli v souboru, ale může být užitečné mít nové položky na začátku souboru:



2. Přidružte konfiguraci LDAP k definici správce front.

   Je třeba přidružit konfiguraci LDAP k definici správce front. Bezprostředně pod položkou `DEFINE AUTHINFO` je položka `ALTER QMGR`. Upravte položku CONNAUTH tak, aby odpovídala nově vytvořenému názvu AUTHINFO. Například v předchozím příkladu bylo definováno `AUTHINFO(USE.LDAP)`, což znamená, že název je `USE.LDAP`. Proto změňte `CONNAUTH('SYSTEM.DEFAULT.AUTHINFO.IDPWOS')` na `CONNAUTH('USE.LDAP')`:

Chcete-li, aby k přepnutí na LDAP došlo okamžitě, volejte příkaz REFRESH SECURITY přidáním řádku ihned za příkaz ALTER QMGR:

*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) +
   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
```

Jak pokračovat dále

Nyní jste připraveni aktualizovat zálohu produktu IBM MQ pro informace o autorizaci LDAP.

OpenShift V 9.2.1 CD EUS ***Část LDAP 2: Aktualizace zálohy IBM MQ pro informace o autorizaci LDAP***

Produkt IBM MQ poskytuje autorizační pravidla s vysokou úrovní granularity, která řídí přístup k objektům IBM MQ. Pokud jste změnili ověření a autorizaci na LDAP, mohou být autorizační pravidla neplatná a vyžadovat aktualizaci.

Než začnete

Tato úloha předpokládá, že jste aktualizovali zálohu pro server LDAP.

Informace o této úloze

Musím to udělat?

Používáte-li již LDAP pro ověření a autorizaci, pak se nepožadují žádné změny. Pokud si nejste jisti, zda se LDAP používá, podívejte se na téma “Volitelné: Konfigurace protokolu LDAP” na stránce 40.

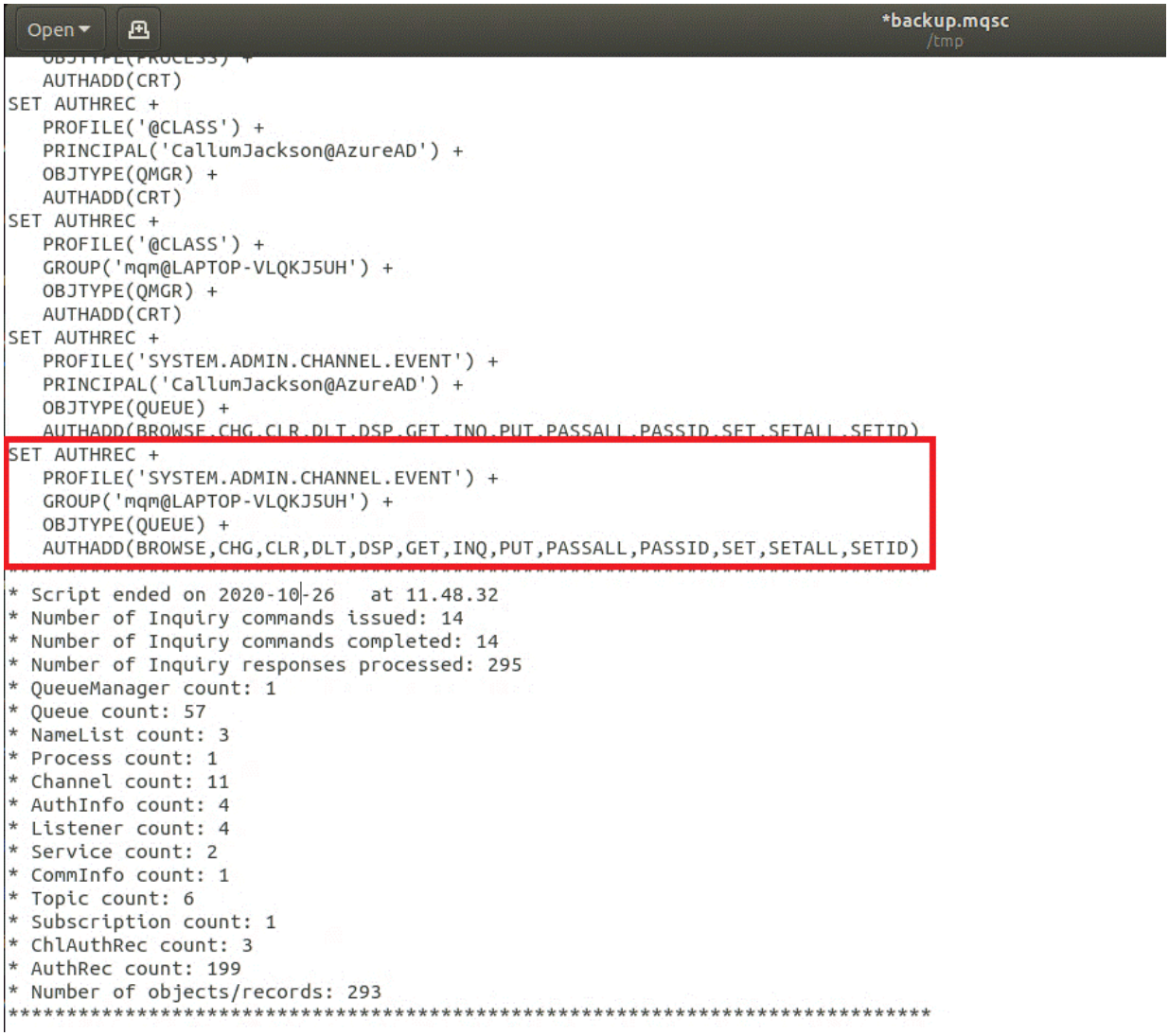
Existují dvě části pro aktualizaci informací o autorizaci LDAP:

1. Odeberte všechna existující oprávnění ze souboru.
2. Definujte nové informace o autorizaci pro LDAP.

Postup

1. Odeberte všechna existující oprávnění ze souboru.

   V souboru zálohy, v blízkosti konce souboru, byste měli vidět několik položek, které začínají položkou SET AUTHREC:



   Najděte existující položky a odstraňte je. Nejjednodušším přístupem je odebrat všechna existující pravidla SET AUTHREC a poté vytvořit nové položky založené na položkách LDAP.
2. Definujte nové informace o autorizaci pro LDAP.

V závislosti na vaší konfiguraci správce front, a počtu prostředků a skupin, může být tato činnost buď časově náročná, anebo jednoduchá. Následující příklad předpokládá, že správce front má pouze jedinou frontu s názvem Q1 a vy chcete povolit přístup skupině LDAP apps.

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

První příkaz AUTHREC přidá oprávnění pro přístup ke správci front a druhý poskytuje přístup ke frontě. Je-li požadován přístup ke druhé frontě, je zapotřebí třetí příkaz AUTHREC, pokud jste se nerozhodli použít zástupné znaky k poskytnutí obecnějšího přístupu.

Zde je další příklad. Potřebuje-li skupina administrátorů (s názvem admins) úplný přístup ke správci front, přidejte následující příkazy:

```
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)
```

Jak pokračovat dále

Nyní jste připraveni změnit adresy IP a názvy hostitelů v konfiguraci IBM MQ.

OpenShift V 9.2.1 CD EUS Volitelné: Změna adres IP a názvů hostitelů v konfiguraci produktu IBM MQ

Je možné, že konfigurace produktu IBM MQ má zadané adresy IP a názvy hostitelů. V některých situacích mohou zůstat, zatímco v jiných situacích je třeba je aktualizovat.

Než začnete

Tato úloha předpokládá, že máte nakonfigurovaný protokol LDAP.

Informace o této úloze

Musím to udělat?

Nejprve určete, zda jsou k dispozici nějaké adresy IP nebo názvy hostitelů, kromě konfigurace LDAP definované v předchozí sekci. Chcete-li to provést, spusťte následující příkaz:

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

Ukázkový výstup:

```
************************************************
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(' ') +
```

V tomto příkladu hledání vrátí tři výsledky. Jeden výsledek odpovídá dříve definované konfiguraci LDAP. To může být ignorováno, protože název hostitele serveru LDAP zůstává stejný. Další dva výsledky jsou prázdné položky připojení, takže je lze také ignorovat. Nemáte-li žádné další položky, můžete zbývající část tématu přeskočit.

Postup

1. Seznamte se s vrácenými položkami.

   Produkt IBM MQ může zahrnovat adresy IP, názvy hostitelů a porty v rámci mnoha aspektů konfigurace. Můžeme je klasifikovat do dvou kategorií:

   a. **Umístění tohoto správce front**: Informace o umístění, které tento správce front používá nebo publikuje, které mohou ostatní správci front nebo aplikace v rámci sítě IBM MQ používat pro konektivitu.

   b. **Umístění závislostí správce front**: Umístění jiných správců front nebo systémů, které tento správce front potřebuje znát.

   Protože se tento scénář zaměřuje pouze na změny provedené v této konfiguraci správce front, zpracujeme pouze aktualizace konfigurace pro kategorii (a). Je-li však toto umístění správce front odkazováno jinými správci front nebo aplikacemi, může jejich konfigurace vyžadovat aktualizaci, aby odpovídala novému umístění správce front.

   Existují dva klíčové objekty, které mohou obsahovat informace, které je třeba aktualizovat:

   - Moduly listener: Představují síťovou adresu, na které produkt IBM MQ naslouchá.
   - Kanál RECEIVER CLUSTER: Pokud je správce front částí klastru IBM MQ, pak tento objekt existuje. Určuje síťovou adresu, ke které se mohou připojit další správci front.

2. V původním výstupu příkazu `grep 'CONNAME\|LOCLADDR\|IPADDRV' -B 3 backup.mqsc` zjistěte, zda jsou definovány kanály CLUSTER RECEIVER. Pokud tomu tak je, aktualizujte adresy IP.

   Chcete-li identifikovat, zda jsou nadefinovány kanály CLUSTER RECEIVER, vyhledejte v původním výstupu všechny položky s CHLTYPE (CLUSRCVR):

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

   Pokud položky existují, aktualizujte CONNAME pomocí IBM MQ Red Hat OpenShift Route. Tato hodnota je založena na prostředí Red Hat OpenShift a používá předvídatelnou syntaxi:

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

   Je-li například implementace správce front pojmenována qm1 v rámci oboru názvů cp4i a *openshift_app_route_hostname* je `apps.callumj.icp4i.com`, potom je adresa URL trasy tato:

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

   Číslo portu pro trasu je obvykle 443. Pokud vám administrátor Red Hat OpenShift neřekne jinak, jedná se obvykle o správnou hodnotu. Pomocí těchto informací aktualizujte pole CONNAME. Příklad:

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

   V původním výstupu příkazu `grep 'CONNAME\|LOCLADDR\|IPADDRV' -B 3 backup.mqsc` ověřte, zda existují nějaké položky pro LOCLADDR nebo IPADDRV. Pokud ano, odstraňte je. Nejsou relevantní v prostředí kontejnerů.

Jak pokračovat dále

Nyní jste připraveni aktualizovat konfiguraci správce front pro prostředí kontejnerů.

Aktualizace konfigurace správce front pro prostředí kontejnerů

Při spuštění v kontejneru jsou určité aspekty konfigurace definovány kontejnerem a mohou být v konfliktu s exportovanou konfigurací.

Než začnete

Tato úloha předpokládá, že jste změnili konfiguraci IBM MQ adres IP a názvů hostitelů.

Informace o této úloze

Následující aspekty konfigurace jsou definovány kontejnerem:

- Definice modulu listener (které odpovídají vystaveným portům).
- Umístění jakéhokoli potenciálního úložiště TLS.

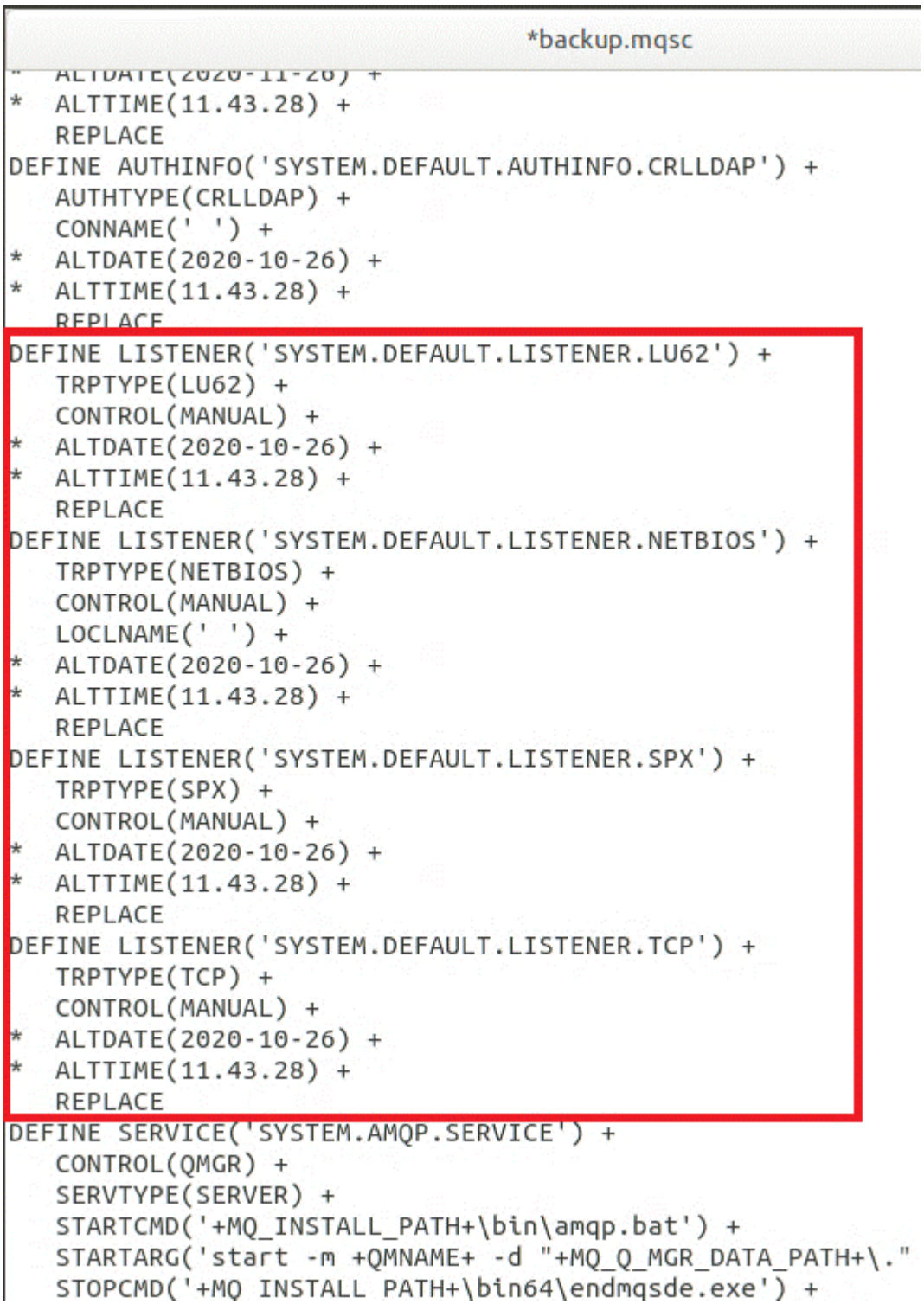
Proto je nutné aktualizovat vyexportovanou konfiguraci:

1. Odeberte všechny definice modulu listener.
2. Definujte umístění úložiště klíčů TLS.

Postup

1. Odeberte všechny definice modulu listener.

   V záložní konfiguraci vyhledejte `DEFINE LISTENER`. To by mělo být mezi definicemi `AUTHINFO` a `SERVICE`. Zvýrazněte oblast a odstraňte ji.



2. Definujte umístění úložiště klíčů TLS.

   Záloha správce front obsahuje konfiguraci TLS pro původní prostředí. To se liší od prostředí kontejneru, a proto je zapotřebí několik aktualizací:

   - Změňte položku **CERTLABL** na `default`.
   - Změňte umístění úložiště klíčů TLS (**SSLKEYR**) na: `/run/runmqserver/tls/key`.

   Chcete-li najít umístění atributu **SSLKEYR** v souboru, vyhledejte **SSLKEYR**. Obvykle je nalezena pouze jedna položka. Je-li nalezeno více položek, zkontrolujte, zda upravujete objekt **QMGR**, jak je znázorněno na následujícím obrázku:

*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) +
   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
```

Jak pokračovat dále

Nyní jste připraveni vybrat cílovou architekturu pro produkt IBM MQ spuštěný v kontejnerech.

OpenShift V 9.2.1 CD EUS

Výběr cílové architektury vysoké dostupnosti pro produkt IBM MQ spuštěný v kontejnerech

Vyberte si mezi jednou instancí (jediný pod Kubernetes) a více instancemi (dva pody), abyste splnili požadavky na vysokou dostupnost.

Než začnete

Tato úloha předpokládá, že jste aktualizovali konfiguraci správce front pro prostředí kontejneru.

Informace o této úloze

IBM MQ Operator poskytuje dvě volby vysoké dostupnosti:

- **Jediná instance**: Jeden kontejner (Pod) je spuštěn a Red Hat OpenShift je zodpovědný za restartování v případě selhání. Vzhledem k charakteristice stavové sady v rámci Kubernetes existuje několik situací, kdy může toto překonání selhání trvat delší dobu, nebo vyžadovat provedení administrativní akce.
- **Více instancí**: Dva kontejnery (každý v odděleném podu) jsou spuštěny, jeden v aktivním režimu a druhý v pohotovostním režimu. Tato topologie umožňuje mnohem rychlejší překonání selhání. Vyžaduje systém souborů RWX (Read Write Many) vyhovující požadavkům produktu IBM MQ.

V této úloze vyberete pouze cílovou architekturu HA. Kroky pro konfiguraci zvolené architektury jsou popsány v následující úloze v tomto scénáři (“Vytvoření nového správce front v Red Hat OpenShift” na stránce 53).

Postup

1. Zkontrolujte dvě volby.

   Úplný popis těchto dvou voleb viz “Vysoká dostupnost pro IBM MQ v kontejnerech” na stránce 16.
2. Vyberte cílovou architekturu HA.

   Pokud si nejste jisti tím, jakou volbu vybrat, začněte volbou **Jedna instance** a ověřte, zda tato volba splňuje vaše požadavky na vysokou dostupnost.

Jak pokračovat dále

Nyní jste připraveni vytvořit konfiguraci správce front.

OpenShift V 9.2.1 CD EUS Vytvoření prostředků pro správce front

Naimportujte konfiguraci produktu IBM MQ a certifikáty a klíče TLS do prostředí Red Hat OpenShift.

Než začnete

Tato úloha předpokládá, že jste vybrali cílovou architekturu pro produkt IBM MQ spuštěný v kontejnerech.

Informace o této úloze

V předchozích sekcích jste extrahovali, aktualizovali a definovali dva prostředky:

- Konfigurace produktu IBM MQ
- Certifikáty a klíče TLS

Tyto prostředky je třeba importovat do prostředí Red Hat OpenShift před implementací správce front.

Postup

1. Importujte konfiguraci IBM MQ do produktu Red Hat OpenShift.

   Následující pokyny předpokládají, že máte konfiguraci produktu IBM MQ v aktuálním adresáři, v souboru s názvem `backup.mqsc`. Jinak je nutné upravit název souboru na základě vašeho prostředí.

   a) Přihlaste se do klastru pomocí `oc login`.

   b) Načtěte konfiguraci produktu IBM MQ do `configmap`.

Spusťte tento příkaz:

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

c) Ověřte, zda byl soubor úspěšně načten.

Spusťte tento příkaz:

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

2. Naimportujte prostředky TLS produktu IBM MQ.

Jak je uvedeno v tématu “Volitelné: Extrakce a získání klíčů a certifikátů správce front” na stránce 38, TLS může být vyžadováno pro implementaci správce front. Pokud tomu tak je, měli byste již mít počet souborů ukončených pomocí `.crt` a `.key`. Musíte je přidat do tajných klíčů Kubernetes pro správce front, na který se odkazuje v době implementace.

Máte-li například klíč a certifikát pro správce front, mohou být volány:

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

Chcete-li tyto soubory naimportovat, spusťte následující příkaz:

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

Kubernetes poskytuje tento užitečný obslužný program, když importujete odpovídající veřejný a soukromý klíč. Máte-li k dispozici další certifikáty, které chcete přidat, například do úložiště údajů o důvěryhodnosti správce front, spusťte následující příkaz:

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

Pokud jsou například soubory, které mají být importovány, `trust1.crt`, `trust2.crt` a `trust3.crt`, příkaz je následující:

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

Jak pokračovat dále

Nyní jste připraveni vytvořit nového správce front v Red Hat OpenShift.

Vytvoření nového správce front v Red Hat OpenShift

Implementujte buď pouze jednu instanci, nebo správce front s více instancemi v Red Hat OpenShift.

Než začnete

Tato úloha předpokládá, že máte vytvořené prostředky správce front a nainstalován IBM MQ Operator do Red Hat OpenShift.

Informace o této úloze

Jak je uvedeno v části “Výběr cílové architektury vysoké dostupnosti pro produkt IBM MQ spuštěný v kontejnerech” na stránce 51, existují dvě možné topologie implementace. V tomto tématu jsou k dispozici dvě různé šablony:

- Implementujte správce front s jednou instancí.
- Implementujte správce front s více instancemi.

Důležité: Na základě vaší preferované topologie proveďte pouze jednu z těchto dvou šablon.

Procedura

- Implementujte správce front s jednou instancí.

  Migrovaný správce front je implementován do Red Hat OpenShift pomocí souboru YAML. Zde je ukázka, která je založena na názvech použitých v předchozích tématech:

```
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
```

  V závislosti na krocích, které jste provedli, může být nutné předchozí YAML upravit. Chcete-li s tímto pomoci, je zde vysvětlení tohoto YAML:

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

  Definuje objekt Kubernetes, typ a název. Jediné pole vyžadující přizpůsobení je pole `name`.

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

  This corresponds to the version and license information for the deployment. If you need to customize this, use the information provided in “Odkaz na licenci pro mq.ibm.com/v1beta1” na stránce 124.

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

  Aby mohl být správce front konfigurován tak, aby používal TLS, musí odkazovat na příslušné certifikáty a klíče. Pole `secretName` odkazuje na tajný údaj Kubernetes vytvořený v sekci Importovat prostředky TLS produktu IBM MQ a seznam položek (`tls.key` a `tls.crt`) jsou standardní názvy, které produkt Kubernetes přiřazuje při použití syntaxe `oc create secret tls`. Máte-li další certifikáty, které

chcete přidat do úložiště údajů o důvěryhodnosti, můžete tyto certifikáty přidat podobným způsobem, ale tyto položky představují odpovídající názvy souborů použité během importu. K vytvoření certifikátů úložiště údajů o důvěryhodnosti lze například použít následující kód:

```
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
```

Důležité: Není-li zabezpečení TLS vyžadováno, odstraňte sekci TLS v YAML.

```
  web:
    enabled: true
```

To umožňuje webovou konzolu pro implementaci

```
  queueManager:
    name: QM1
```

Definuje název správce front jako QM1. Správce front je upraven na základě vašich požadavků, například jaký byl původní název správce front.

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

Předchozí kód se stáhne do konfigurace správce front, která byla naimportována v sekci Import konfigurace IBM MQ. Pokud jste použili jiné názvy, je třeba upravit `my-mqsc-migrated` a `backup.mqsc`.

Všimněte si, že ukázka YAML předpokládá, že výchozí paměťová třída pro prostředí Red Hat OpenShift je definována jako třída úložiště RWX nebo RWO. Není-li ve vašem prostředí definováno výchozí nastavení, je nutné určit paměťovou třídu, která má být použita. Toto můžete provést rozšířením YAML takto:

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

Přidejte zvýrazněný text s přizpůsobeným atributem třídy, aby odpovídal vašemu prostředí. Chcete-li zjistit názvy paměťových tříd ve svém prostředí, spusťte následující příkaz:

```
oc get storageclass
```

Zde je příklad výstupu vráceného tímto příkazem:

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

Následující kód ukazuje, jak odkazovat na konfiguraci IBM MQ, která byla naimportována v sekci Import konfigurace produktu IBM MQ. Pokud jste použili jiné názvy, je třeba upravit `my-mqsc-migrated` a `backup.mqsc`.

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

Implementovali jste správce front s jednou instancí. Tím je dokončena šablona. Nyní jste připraveni ověřit novou implementace kontejneru.

- Implementujte správce front s více instancemi.

  Migrovaný správce front je implementován do Red Hat OpenShift pomocí souboru YAML. Následující ukázka je založena na názvech použitých v předchozích sekcích.

```
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
```

Zde je vysvětlení tohoto YAML. Většina konfigurace má stejný přístup jako implementace správce front s jednou instancí, proto jsou zde vysvětleny pouze aspekty dostupnosti správce front a úložiště.

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

Určuje název správce front jako `QM1` a nastavuje implementaci `MultiInstance` místo výchozí jediné instance.

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

```
      recoveryLogs:
        enabled: true
```

Správce front s více instancemi produktu IBM MQ závisí na úložišti RWX. Standardně je správce front implementován v režimu jedné instance, a proto jsou při přechodu na režim více instancí zapotřebí další volby úložiště. V předchozí ukázce YAML jsou definovány tři trvalé svazky úložišť a trvalá třída svazku. Tato trvalá třída svazku musí být paměťová třída RWX. Pokud si nejste jisti názvy paměťových tříd ve vašem prostředí, můžete spustit následující příkaz a zjistit je:

```
oc get storageclass
```

Zde je příklad výstupu vráceného tímto příkazem:

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

Následující kód ukazuje, jak odkazovat na konfiguraci IBM MQ, která byla naimportována v sekci Import konfigurace produktu IBM MQ. Pokud jste použili jiné názvy, je třeba upravit `my-mqsc-migrated` a `backup.mqsc`.

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

Implementovali jste správce front s více instancemi. Tím je dokončena šablona. Nyní jste připraveni ověřit novou implementace kontejneru.

OpenShift V 9.2.1 CD EUS Ověření implementace nového kontejneru

Nyní, když je produkt IBM MQ implementován v Red Hat OpenShift, můžete ověřit prostředí pomocí ukázek produktu IBM MQ.

Než začnete

Tato úloha předpokládá, že jste vytvořili nového správce front v Red Hat OpenShift.

Důležité: Tato úloha předpokládá, že TLS není v daném správci front povoleno.

Informace o této úloze

V této úloze spustíte ukázky produktu IBM MQ z kontejneru správce front převedeného migrací. Je však možné, že budete chtít používat vlastní aplikace spuštěné v jiném prostředí.

Potřebujete tyto informace:

- Jméno uživatele LDAP
- Heslo LDAP
- IBM MQ - Název kanálu
- Název fronty

Tento vzorový kód používá následující nastavení. Všimněte si, že vaše nastavení se bude lišit.

- Jméno uživatele LDAP: mqapp
- Heslo LDAP: mqapp
- Název kanálu produktu IBM MQ: DEV.APP.SVRCONN
- Název fronty: Q1

Postup

1. Exec do spuštěného kontejneru IBM MQ.

   Zadejte následující příkaz:

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

   Kde `qm1-ibm-mq-0` je pod, který jsme implementovali v “Vytvoření nového správce front v Red Hat OpenShift” na stránce 53. Pokud jste implementaci nazvali něčím jiným, upravte tuto hodnotu.
2. Odešlete zprávu.

   Spusťte následující příkazy:

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

   Zobrazí se výzva k zadání hesla a poté můžete odeslat zprávu.
3. Ověřte, zda byla zpráva úspěšně přijata.

   Spusťte ukázku GET:

   ```
   ./amqsgetc Q1 QM1
   ```

Výsledky

Dokončili jste “Migrace IBM MQ do produktu IBM Cloud Pak for Integration” na stránce 36.

Jak pokračovat dále

Použijte následující informace, které vám pomohou se složitějšími scénáři migrace:

Migrace zpráv ve frontě
Chcete-li migrovat existující zprávy ve frontě, postupujte podle pokynů v následujícím tématu pro export a import zpráv po zavedení nového správce front: Použití obslužného programu dmpmqmsg mezi dvěma systémy.

Připojení k produktu IBM MQ mimo prostředí Red Hat OpenShift
Implementovaný správce front může být vystaven klientům IBM MQ a správcům front mimo prostředí Red Hat OpenShift. Proces závisí na verzi produktu IBM MQ, která se připojuje k prostředí Red Hat OpenShift. Viz téma “Konfigurace trasy pro připojení ke správci front mimo klastr Red Hat OpenShift” na stránce 107.

OpenShift CP4I Instalace a odinstalace produktu IBM MQ Operator v systému Red Hat OpenShift

Produkt IBM MQ Operator může být nainstalován na Red Hat OpenShift pomocí produktu Operator Hub.

Procedura

- “Závislosti pro IBM MQ Operator” na stránce 10.
- “Oprávnění s vymezeným klastrem vyžadovaná produktem IBM MQ Operator” na stránce 10.
- “Instalace produktu IBM MQ Operator pomocí webové konzoly Red Hat OpenShift” na stránce 59.
- “Instalace IBM MQ Operator pomocí rozhraní CLI Red Hat OpenShift” na stránce 61.
- “Instalace komponenty IBM MQ Operator v prostředí airgap” na stránce 65.

Související úlohy

“Odinstalace produktu IBM MQ Operator pomocí webové konzoly Red Hat OpenShift” na stránce 60
K odinstalování produktu IBM MQ Operator z Red Hat OpenShift můžete použít webovou konzolu Red Hat OpenShift .

“Odinstalace produktu IBM MQ Operator pomocí rozhraní CLI Red Hat OpenShift” na stránce 63
Rozhraní CLI Red Hat OpenShift můžete použít k odinstalaci IBM MQ Operator z Red Hat OpenShift. Existují rozdíly v procesu odinstalace v závislosti na tom, zda je IBM MQ Operator nainstalován v jednom oboru názvů nebo nainstalován a zda je k dispozici pro všechny obory názvů v klastru.

OpenShift CP4I Instalace produktu IBM MQ Operator pomocí webové konzoly Red Hat OpenShift

Produkt IBM MQ Operator může být nainstalován na Red Hat OpenShift pomocí produktu Operator Hub.

Než začnete

Přihlaste se na webovou konzolu klastru Red Hat OpenShift.

Postup

1. EUS
   Volitelné: Přidejte operátory IBM Common Services do seznamu instalovatelných operátorů.

   Poznámka:

   Tento krok platí pro vydání IBM MQ Operator 1.5 a dřívější. Krok přidá samostatný katalog Common Services. Pro pozdější vydání operátora jsou služby Common Services zahrnuty do katalogu IBM.

   a) Klepněte na ikonu plus v pravém horním rohu obrazovky. Zobrazí se dialogové okno **Importovat YAML**.

   b) Vložte následující definici prostředku v dialogovém okně.

   ```
   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) Klepněte na volbu **Vytvořit**.
2. Přidejte operátory IBM do seznamu instalovatelných operátorů.

   a) Klepněte na ikonu plus v pravém horním rohu obrazovky. Zobrazí se dialogové okno **Importovat YAML**.

   b) Vložte následující definici prostředku v dialogovém okně.

   ```
   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) Klepněte na volbu **Vytvořit**.

3. Vytvořte obor názvů, který se má použít pro IBM MQ Operator.

   IBM MQ Operator lze nainstalovat s vymezeným rozsahem do jednoho nebo všech oborů názvů. Tento krok je nezbytný pouze v případě, že chcete instalovat do konkrétního oboru názvů, který ještě neexistuje.

   a) V navigačním podokně klepněte na volbu **Domů** > **Projekty**.

   Zobrazí se stránka Projekty.

   b) Klepněte na volbu **Vytvořit projekt**. Zobrazí se oblast Vytvořit projekt.

   c) Zadejte podrobnosti o oboru názvů, který vytváříte. Např. můžete určit "ibm-mq" jako název.

   d) Klepněte na volbu **Vytvořit**. Vytvoří se obor názvů pro IBM MQ Operator.

4. Nainstalujte IBM MQ Operator.

   a) V navigačním podokně klepněte na volbu **Operators** > **OperatorHub**.

   Zobrazí se stránka OperatorHub.

   b) Do pole **Všechny položky** zadejte hodnotu "IBM MQ".

   Zobrazí se položka katalogu IBM MQ.

   c) Vyberte volbu **IBM MQ**.

   Zobrazí se okno IBM MQ.

   d) Klepněte na volbu **Instalovat**.

   Zobrazí se stránka Vytvořit odběr operátoru.

   e) Chcete-li určit, na který kanál operátoru se má použít, zkontrolujte “Podpora verze pro IBM MQ Operator” na stránce 7.

   f) Nastavte režim instalace buď na specifický obor názvů, který jste vytvořili, nebo na rozsah klastru.

   Výběr rozsahu celého klastru se doporučuje, protože při instalaci různých verzí operátoru v různých oborech názvů může dojít k problémům. Operátory jsou navrženy tak, aby byly rozšířením řídicí roviny.

   g) Klepněte na volbu **Odebírat**.

   Na stránce Instalované operátory uvidíte IBM MQ.

   h) Zkontrolujte stav operátoru na stránce Instalované operátory, stav se změní po dokončení instalace na Succeeded.

Jak pokračovat dále

“Příprava projektu Red Hat OpenShift pro produkt IBM MQ pomocí webové konzoly Red Hat OpenShift” na stránce 79

OpenShift CP4I

Odinstalace produktu IBM MQ Operator pomocí webové konzoly Red Hat OpenShift

K odinstalování produktu IBM MQ Operator z Red Hat OpenShift můžete použít webovou konzolu Red Hat OpenShift .

Než začnete

Přihlaste se na webovou konzolu klastru Red Hat OpenShift.

Je-li produkt IBM MQ Operator nainstalován ve všech projektech/oborech názvů v klastru, zopakujte kroky 1-5 následující procedury pro každý projekt, u kterého chcete odstranit správce front.

Postup

1. Vyberte **Operátory** > **Instalované operátory**.
2. Z rozevíracího seznamu **Projekt** vyberte projekt.

3. Klepněte na operátor **IBM MQ**.
4. Chcete-li zobrazit správce front spravovaných tímto produktem IBM MQ Operator, klepněte na kartu **Správci front** .
5. Odstraňte jednoho nebo více správců front.

   Mějte na zřeteli, že i když jsou tito správci front nadále spuštěni, nemusí bez IBM MQ Operator fungovat podle očekávání.
6. Volitelné: V případě potřeby zopakujte kroky 1-5 pro každý projekt, u kterého chcete odstranit správce front.
7. Vraťte se na volbu **Operátory** > **Instalované operátory**.
8. Vedle operátoru **IBM MQ** klepněte na tři tečky a vyberte volbu **Odinstalovat operátor**.
9. Používáte-li Red Hat OpenShift Container Platform 4.7, může být nutné ručně odstranit ověřovací webhook z příkazového řádku:

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

OpenShift CP4I Instalace IBM MQ Operator pomocí rozhraní CLI Red Hat OpenShift

Produkt IBM MQ Operator může být nainstalován na Red Hat OpenShift pomocí produktu Operator Hub.

Než začnete

Přihlaste se do rozhraní příkazového řádku (CLI) Red Hat OpenShift pomocí **oc login**. V rámci těchto kroků budete muset být administrátorem klastru.

Postup

1. EUS

   Volitelné: Vytvořte **CatalogSource** pro operátory obecných služeb IBM.

   Poznámka:

   Tento krok platí pro vydání IBM MQ Operator 1.5 a dřívější. Krok přidá samostatný katalog Common Services. Pro pozdější vydání operátora jsou služby Common Services zahrnuty do katalogu IBM.

   a) Vytvořte soubor YAML definující prostředek **CatalogSource**.

   Vytvořte soubor s názvem "operator-source-cs.yaml" s následujícím obsahem:

```
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) Použijte **CatalogSource** na server.

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

2. Vytvořit **CatalogSource** pro operátory IBM

   a) Vytvořit soubor YAML definující prostředek **CatalogSource**

   Vytvořte soubor s názvem "operator-source-ibm.yaml" s následujícím obsahem:

```
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) Použijte **CatalogSource** na server.

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

3. Vytvořte obor názvů, který se má použít pro IBM MQ Operator.

   IBM MQ Operator lze nainstalovat s vymezeným rozsahem do jednoho nebo všech oborů názvů. Tento krok je nezbytný pouze v případě, že chcete instalovat do konkrétního oboru názvů, který ještě neexistuje.

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

4. Zobrazte seznam operátorů dostupných pro klastr z OperatorHub.

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

5. Zkontrolujte IBM MQ Operator a ověřte jeho podporované režimy InstallModes a dostupné kanály.

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

6. Vytvořit soubor YAML objektu **OperatorGroup**

   OperatorGroup je prostředek OLM, který vybírá cílové obory názvů, v nichž se má generovat požadovaný přístup RBAC pro všechny operátory ve stejném oboru názvů jako server **OperatorGroup**.

   Obor názvů, k jehož odběru přihlašujete operátora, musí mít **OperatorGroup**, který odpovídá **InstallMode** operátora, ať už v režimu `Všechny obory názvů`, nebo `Jeden obor názvů`. Pokud operátor, kterého chcete instalovat, používá režim `Všechny obory názvů`, pak již má obor názvů `openshift-operators` umístěn vhodný **OperatorGroup**.

   Pokud však operátor používá režim `Jeden obor názvů` a vy ještě nemáte umístěn vhodný **OperatorGroup**, je třeba jeden vytvořit.

   a) Vytvořte soubor s názvem "mq-operator-group.yaml" s následujícím obsahem:

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

   b) Vytvořit objekt **OperatorGroup**

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

7. Vytvořte soubor YAML objektu **Subscription** a přihlaste obor názvů k odběru IBM MQ Operator.

   a) Chcete-li určit, na který kanál operátoru se má použít, zkontrolujte “Podpora verze pro IBM MQ Operator” na stránce 7.

   b) Vytvořte soubor s názvem "mq-sub.yaml" s následujícím obsahem, který však mění **channel**, aby se shodoval s kanálem pro verzi IBM MQ Operator, kterou chcete instalovat.

```
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
```

Pro použití `Všech oborů názvů` **`InstallMode`** uveďte v oboru názvů **`openshift-operators`**. Jinak uveďte jeden příslušný obor názvů pro použití `Jednoho oboru názvů` **`InstallMode`**. Všimněte si, že byste měli změnit pouze pole **`namespace`** a ponechat pole **`sourceNamespace`** tak, jak je.

c) Vytvořit objekt **`Subscription`**

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

8. Zkontrolovat stav operátora

Jakmile je instalace operátora úspěšná, stav podu se zobrazí jako *Spuštěný*. Pro použití `Všech oborů názvů` **`InstallMode`** uveďte jako obor názvů **`openshift-operators`**. Jinak uveďte jeden příslušný obor názvů pro použití `Jednoho oboru názvů` **`InstallMode`**.

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

Jak pokračovat dále

"Příprava projektu produktu Red Hat OpenShift pro IBM MQ pomocí rozhraní CLI Red Hat OpenShift" na stránce 79

OpenShift CP4I *Odinstalace produktu IBM MQ Operator pomocí rozhraní CLI Red Hat OpenShift*

Rozhraní CLI Red Hat OpenShift můžete použít k odinstalaci IBM MQ Operator z Red Hat OpenShift. Existují rozdíly v procesu odinstalace v závislosti na tom, zda je IBM MQ Operator nainstalován v jednom oboru názvů nebo nainstalován a zda je k dispozici pro všechny obory názvů v klastru.

Než začnete

Přihlaste se ke svému klastru Red Hat OpenShift pomocí `oc login`.

Procedura

- Je-li IBM MQ Operator nainstalován v jednom oboru názvů, proveďte následující dílčí kroky:

a) Ujistěte se, že jste ve správném projektu:

```
oc project <project_name>
```

b) Zobrazte správce front nainstalovaného v projektu:

```
oc get qmgr
```

c) Odstraňte jednoho nebo více správců front:

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

Mějte na zřeteli, že i když jsou tito správci front nadále spuštěni, nemusí bez IBM MQ Operator fungovat podle očekávání.

d) Zobrazte instance **`ClusterServiceVersion`**:

```
oc get csv
```

e) Odstraňte IBM MQ **ClusterServiceVersion**:

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

f) Zobrazte odběry:

```
oc get subscription
```

g) Odstraňte všechny odběry:

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

h) Volitelné: Pokud nikdo další nepoužívá běžné služby, může být vhodné odinstalovat operátor běžných služeb a odstranit skupinu operátorů:

a. Odinstalujte operátor běžných služeb podle pokynů v části Odinstalace běžných služeb v dokumentaci produktu IBM Cloud Pak foundational services.

b. Zobrazte skupinu operátorů:

```
oc get operatorgroup
```

c. Odstraňte skupinu operátorů:

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

- Je-li IBM MQ Operator nainstalován a k dispozici pro všechny obory názvů v klastu, proveďte následující dílčí kroky:

a) Zobrazit všechny instalované správce front:

```
oc get qmgr -A
```

b) Odstraňte jednoho nebo více správců front:

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

Mějte na zřeteli, že i když jsou tito správci front nadále spuštěni, nemusí bez IBM MQ Operator fungovat podle očekávání.

c) Zobrazte instance **ClusterServiceVersion**:

```
oc get csv -A
```

d) Odstraňte IBM MQ **ClusterServiceVersion** z klastru:

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

e) Zobrazte odběry:

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

f) Odstraňte odběry:

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

g) Používáte-li Red Hat OpenShift Container Platform 4.7, může být nutné ručně odstranit ověřovací webhook:

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

h) Volitelné: Pokud nikdo další nepoužívá běžné služby, může být vhodné odinstalovat operátor běžných služeb:

Postupujte podle pokynů v části Odinstalace běžných služeb v dokumentaci produktu IBM Cloud Pak foundational services.

Instalace komponenty IBM MQ Operator v prostředí airgap

Tento výukový program vás provede instalací produktu IBM MQ Operator do klastru Red Hat OpenShift, který nemá žádné připojení k internetu. IBM MQ Operator můžete nainstalovat v prostředí airgap pomocí přenosného úložného zařízení nebo pomocí počítače bastion.

Instalace IBM MQ Operator v prostředí airgap pomocí přenosného úložného zařízení

Kroky k dokončení instalace viz Zrcadlení obrázků s přenosným úložným zařízením v dokumentaci IBM Cloud Pak for Integration. Pokud instalujete pouze produkt IBM MQ, potom nahraďte všechny výskyty následujících proměnných prostředí hodnotami uvedenými zde:

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

Kde *version_number* je verze případu, kterou chcete použít k provedení instalace airgap. Seznam dostupných verzí případu viz https://github.com/IBM/cloud-pak/tree/master/repo/case/ibm-mq. Chcete-li určit, na který kanál operátoru se má použít, zkontrolujte “Podpora verze pro IBM MQ Operator” na stránce 7.

Instalace IBM MQ Operator v prostředí airgap pomocí počítače bastion

Požadavky

1. Musí být nainstalován klastr Red Hat OpenShift Container Platform. Informace o podporovaných verzích produktu Red Hat OpenShift Container Platform viz “Podpora verze pro IBM MQ Operator” na stránce 7.
2. Registr Docker musí být k dispozici. Další informace viz téma “Příprava registru Docker” na stránce 65.
3. Opevněný (bastion) server musí být nakonfigurován. Další informace viz téma “Příprava opevněného (bastion) hostitele” na stránce 66.

Příprava registru Docker

Lokální registr Docker se používá k ukládání všech obrazů v lokálním prostředí. Musíte vytvořit takový registr a zajistit, aby vyhovoval následujícím požadavkům:

- Podporuje Docker Manifest V2, Schema 2.
- Podporuje obrazy s více architekturami.
- Je přístupný jak z opevněného serveru, tak i z uzlů klastru Red Hat OpenShift Container Platform.

- Má jméno uživatele a heslo uživatele, který může zapisovat do cílového registru z opevněného hostitele.
- Má jméno uživatele a heslo uživatele, který může číst z cílového registru, který se nachází na uzlech klastru Red Hat OpenShift.
- Umožňuje oddělovače cest v názvu obrazu.

Po vytvoření registru Docker musíte nakonfigurovat registr:

1. Vytvoření oborů názvů registru
   - `ibmcom` - Obor názvů k uložení všech obrazů z oboru názvů `dockerhub.io/ibmcom`.

     Obor názvů `ibmcom` je určen pro všechny obrazy IBM, které jsou veřejně dostupné a nevyžadují pověření pro stažení.
   - `cp` - Obor názvů k uložení obrazů IBM z úložiště `cp.icr.io/cp`.

     Obor názvů `cp` je určen pro obrazy v produktu IBM Entitled Registry, které vyžadují klíč oprávnění k produktu a pověření ke stažení. Chcete-li získat klíč oprávnění, přihlaste se do MyIBM Container Software Library s ID IBM a heslem, které jsou přidruženy k oprávněnému softwaru. V sekci **Klíče oprávnění** vyberte **Kopírovat klíč** ke zkopírování klíče oprávnění do schránky (clipboardu), potom jej uložte pro použití v následujícím postupu.
   - `opencloudio` - Obor názvů pro uložení obrazů z `quay.io/opencloudio`.

     Obor názvů `opencloudio` je určen pro výběr obrazů komponenty IBM typu open source, které jsou k dispozici v quay.io. Obrazy IBM Cloud Pak foundational services jsou hostovány na `opencloudio`.
2. Ověřte, že každý obor názvů splňuje následující požadavky:
   - Podporuje automatické vytváření úložišť.
   - Má pověření uživatele, který může zapisovat a vytvářet úložiště. Tato pověření opevněný hostitel používá.
   - Má pověření uživatele, který může číst všechna úložiště. Klastr Red Hat OpenShift Container Platform používá tato pověření.

Příprava opevněného (bastion) hostitele

Připravte opevněného hostitele (opevněný hostitelský počítač), který může přistupovat ke klastru Red Hat OpenShift Container Platform, lokálnímu registru Docker a Internetu. Bastion hostitel musí být na platformě Linux for x86-64 s libovolným operačním systémem, který podporuje rozhraní IBM Cloud Pak CLI a Red Hat OpenShift Container Platform CLI.

Na vašem opevněném uzlu proveďte tyto kroky:

1. Nainstalujte OpenSSL verze 1.11.1 nebo vyšší.
2. Nainstalujte Docker nebo Podman na opevněný uzel.
   - Chcete-li nainstalovat Docker, spusťte tyto příkazy:

     ```
     yum check-update
     yum install docker
     ```
   - Chcete-li nainstalovat Podman, postupujte podle pokynů k instalaci Podman
3. Nainstalujte skopieo verze 1.x.x na uzel bastion. Chcete-li nainstalovat skopeo, spusťte tyto příkazy:

   ```
   yum check-update
   yum install skopeo
   ```
4. Nainstalujte IBM Cloud Pak CLI. Nainstalujte nejnovější verzi binárního souboru na své platformě. Další informace viz cloud-pak-cli.
   a. Stáhněte binární soubor.

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

Příklad:

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

b. Extrahujte binární soubor.

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

c. Chcete-li upravit a přesunout soubor, spusťte následující příkazy.

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

d. Ověřte, že je nainstalován `cloudctl` :

```
cloudctl --help
```

5. Nainstalujte nástroj `oc` Red Hat OpenShift Container Platform CLI.

   Další informace viz Nástroje Red Hat OpenShift Container Platform CLI

6. Vytvořte adresář, který slouží jako úložiště offline.

   Zde je uveden příklad adresáře. Tento příklad se používá v následných krocích.

```
mkdir $HOME/offline
```

   Poznámka: Toto úložiště offline musí být trvalé, aby se zabránilo násobnému přenosu dat. Perzistence také pomáhá spustit proces zrcadlení vícekrát nebo podle plánu.

Vytvoření proměnných prostředí pro instalační program a inventář obrazů

Vytvořte následující proměnné prostředí s názvem obrazu instalačního programu a inventářem obrazů:

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

Kde *version_number* je verze případu, kterou chcete použít k provedení instalace airgap. Seznam dostupných verzí případu viz https://github.com/IBM/cloud-pak/tree/master/repo/case/ibm-mq. Chcete-li určit, který kanál operátora zvolit, prohlédněte si téma Podpora verzí pro server IBM MQ Operator .

Stáhněte si instalační program produktu IBM MQ a inventář obrazů

Stáhněte si instalační program produktu `ibm-mq` a inventář obrazů na hostitele typu bastion:

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

Přihlaste se do klastru Red Hat OpenShift Container Platform jako administrátor klastru.

Zde je příklad příkazu k přihlášení do klastru Red Hat OpenShift Container Platform:

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

Vytvoření oboru názvů Kubernetes pro IBM MQ Operator

Vytvořte proměnnou prostředí s oborem názvů pro instalaci produktu IBM MQ Operatora poté vytvořte obor názvů:

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

Zrcadlení obrazů a konfigurace klastru

Proveďte tyto kroky, chcete-li zrcadlit obrazy a nakonfigurovat klastr:

Poznámka: V žádném příkazu nepoužívejte vlnovku v uvozovkách. Nepoužívejte například `args "--registry registry --user registry_userid --pass registry_password --inputDir ~/offline"`. Tato vlnovka se nerozbalí a vaše příkazy mohou selhat.

1. Uložte ověřovací pověření pro všechny zdrojové registry Docker.

   Všechny obrazy IBM Cloud Platform Common Services, IBM MQ Operator a IBM MQ Advanced Developer jsou uloženy do veřejných registrů, které nevyžadují ověření. Nicméně IBM MQ Advanced Server (neDeveloper), jiné produkty a komponenty třetích stran vyžadují jeden nebo více ověřených registrů. Následující registry vyžadují ověření:

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

   Další informace o těchto registrech viz Vytvoření oborů názvů registru.

   Chcete-li nakonfigurovat pověření pro všechny registry vyžadující ověření, musíte spustit následující příkaz. Spusťte příkaz samostatně pro každý takový registr:

   ```
   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"
   ```

   Příkaz ukládá, a zachytává v mezipaměti, pověření registru do souboru ve vašem systému souborů v umístění `$HOME/.airgap/secrets`.

2. Vytvořte proměnné prostředí s použitím informací o připojení lokálního registru Docker.

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

   Poznámka: Registr Docker používá standardní porty, např. 80 nebo 443. Pokud váš registr Docker používá nestandardní port, uveďte port pomocí syntaxe *host*:*port*. Příklad:

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

3. Nakonfigurujte tajný údaj ověření pro lokální registr Docker.

   Poznámka: Tento krok je třeba provést pouze jednou.

   ```
   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}"
   ```

Příkaz ukládá, a zachytává v mezipaměti, pověření registru do souboru ve vašem systému souborů v umístění $HOME/.airgap/secrets.

4. Nakonfigurujte globální tajný údaj stažení obrazu a **ImageContentSourcePolicy**.

   a. Zkontrolujte, zda je vyžadován restart uzlu.

   - V produktu Red Hat OpenShift Container Platform verze 4.4 a vyšší a v nové instalaci produktu IBM MQ Operator pomocí airgap tento krok restartuje všechny uzly klastru. Prostředky klastru mohou být nedostupné, dokud není použit nový tajný údaj stažení.
   - V produktu IBM MQ Operator 1.8je CASE aktualizován tak, aby zahrnoval další zdroj zrcadlení pro obrazy. Proto se při upgradu z předchozích verzí produktu IBM MQ Operator na verzi 1.8 nebo vyšší spustí restart uzlu.
   - Chcete-li zkontrolovat, zda tento krok vyžaduje restart uzlu, přidejte volbu `--dry-run` do kódu tohoto kroku. Tím se vygeneruje nejnovější soubor **ImageContentSourcePolicy** a zobrazí se v okně konzoly (**stdout**). Pokud se **ImageContentSourcePolicy** liší od konfigurovaného klastru **ImageContentSourcePolicy**, dojde k restartování.

     ```
     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. Chcete-li nakonfigurovat tajný klíč stažení globálního obrazu a **ImageContentSourcePolicy**, spusťte kód pro tento krok bez volby `--dry-run` :

   ```
   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. Ověřte, že je vytvořen prostředek **ImageContentSourcePolicy**.

   ```
   oc get imageContentSourcePolicy
   ```

6. Volitelné: Používáte-li nezabezpečený registr, musíte přidat lokální registr do seznamu **insecureRegistries** klastru.

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

7. Ověřte stav uzlu klastru.

   ```
   oc get nodes
   ```

   Po uplatnění **imageContentsourcePolicy** a tajného údaje stažení globálního obrazu, můžete vidět stav uzlu jako **Ready**, **Scheduling** nebo **Disabled**. Počkejte, dokud všechny uzly nebudou zobrazovat stav **Ready**.

8. Zrcadlete obrazy do lokálního registru.

   ```
   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"
   ```

Nainstalujte IBM MQ Operator.

1. Přihlaste se na webovou konzolu klastru Red Hat OpenShift.
2. Vytvořte zdroj katalogu. Použijte stejný terminál, který provedl předchozí kroky.

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

3. Ověřte, že je pro operátor instalačního programu běžných služeb vytvořen **CatalogSource**.

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

4. Nainstalujte IBM MQ Operator pomocí OLM.
   a. V navigačním podokně klepněte na volbu **Operators** > **OperatorHub**.

      Zobrazí se stránka **OperatorHub**.
   b. Do pole **Všechny položky** zadejte hodnotu `IBM MQ`.

      Zobrazí se položka katalogu IBM MQ.
   c. Vyberte volbu **IBM MQ**.

      Zobrazí se okno **IBM MQ**.
   d. Klepněte na volbu **Instalovat**.

      Zobrazí se stránka **Vytvořit odběr operátoru**.
   e. Chcete-li určit, na který kanál operátoru se má použít, zkontrolujte “Podpora verze pro IBM MQ Operator” na stránce 7.
   f. Nastavte **Režim instalace** buď na specifický obor názvů, který jste vytvořili, nebo na rozsah celého klastru.
   g. Klepněte na volbu **Odebírat**.

      Produkt **IBM MQ** je přidán na stránku **Instalované operátory**.
   h. Zkontrolujte stav operátoru na stránce **Instalované operátory**. Stav se změní na **Succeeded** po dokončení instalace.

Implementace správce front produktu IBM MQ

Vytvoření nového správce front pod instalovaným operátorem viz “Implementace a konfigurace správců front pomocí IBM MQ Operator” na stránce 78.

Související úlohy

“Příprava na přechod na vyšší verzi produktu IBM MQ Operator nebo správce front v prostředí s přechodovou mezerou” na stránce 71
V klastru Red Hat OpenShift , který nemá připojení k Internetu, existují přípravné kroky, které je třeba provést před přechodem na vyšší verzi produktu IBM MQ Operator.

OpenShift CP4I Upgrade produktu IBM MQ Operator a správců front

Upgrade produktu IBM MQ Operator vám umožní upgradovat správce front.

Procedura

- “Upgrade produktu IBM MQ Operator pomocí webové konzoly Red Hat OpenShift” na stránce 74.
- “Upgrade produktu IBM MQ Operator pomocí rozhraní CLI Red Hat OpenShift” na stránce 75.
- “Upgrade správce front IBM MQpomocí webové konzoly Red Hat OpenShift” na stránce 76.

- “Upgrade správce front IBM MQ pomocí rozhraní CLI Red Hat OpenShift” na stránce 77.

OpenShift CP4I Linux Příprava na přechod na vyšší verzi produktu IBM MQ Operator nebo správce front v prostředí s přechodovou mezerou

V klastru Red Hat OpenShift , který nemá připojení k Internetu, existují přípravné kroky, které je třeba provést před přechodem na vyšší verzi produktu IBM MQ Operator.

Než začnete

Toto téma předpokládá, že jste již nakonfigurovali lokální registr obrázků, ve kterém se zrcadlí předchozí vydané obrazy IBM Cloud Pak for Integration .

Informace o této úloze

Než budete moci upgradovat produkt IBM MQ Operator nebo správce front v prostředí airgap, musíte zrcadlit nejnovější obrazy IBM Cloud Pak for Integration .

Všimněte si, že první čtyři kroky v této úloze jsou stejné jako kroky, které provádíte, když je “Instalace komponenty IBM MQ Operator v prostředí airgap” na stránce 65.

Postup

1. Vytvořte proměnné prostředí pro instalační program a soupis obrazů.

   Vytvořte následující proměnné prostředí s názvem obrazu instalačního programu a inventářem obrazů:

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

   Kde *version_number* je verze případu, kterou chcete použít k provedení instalace airgap. Seznam dostupných verzí případu viz https://github.com/IBM/cloud-pak/tree/master/repo/case/ibm-mq. Chcete-li určit, který kanál operátora zvolit, prohlédněte si téma Podpora verzí pro server IBM MQ Operator .
2. Stáhněte si instalační program produktu IBM MQ a soupis obrazů.

   Stáhněte si instalační program produktu `ibm-mq` a inventář obrazů na hostitele typu bastion:

   ```
   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. Přihlaste se do klastru produktu Red Hat OpenShift Container Platform jako administrátor klastru.

   Zde je příklad příkazu k přihlášení do klastru Red Hat OpenShift Container Platform:

   ```
   oc login cluster_host:port --username=cluster_admin_user --password=cluster_admin_password
   ```
4. Zrcadlete obrazy a nakonfigurujte klastr.

   Proveďte tyto kroky, chcete-li zrcadlit obrazy a nakonfigurovat klastr:

   Poznámka: V žádném příkazu nepoužívejte vlnovku v uvozovkách. Nepoužívejte například `args "--registry registry --user registry_userid --pass registry_password --inputDir ~/offline"`. Tato vlnovka se nerozbalí a vaše příkazy mohou selhat.

   a. Uložte ověřovací pověření pro všechny zdrojové registry Docker.

      Všechny obrazy IBM Cloud Platform Common Services, IBM MQ Operator a IBM MQ Advanced Developer jsou uloženy do veřejných registrů, které nevyžadují ověření. Nicméně IBM MQ Advanced Server (neDeveloper), jiné produkty a komponenty třetích stran vyžadují jeden nebo více ověřených registrů. Následující registry vyžadují ověření:

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

Další informace o těchto registrech viz Vytvoření oborů názvů registru.

Chcete-li nakonfigurovat pověření pro všechny registry vyžadující ověření, musíte spustit následující příkaz. Spusťte příkaz samostatně pro každý takový registr:

```
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"
```

Příkaz ukládá, a zachytává v mezipaměti, pověření registru do souboru ve vašem systému souborů v umístění `$HOME/.airgap/secrets`.

b. Vytvořte proměnné prostředí s použitím informací o připojení lokálního registru Docker.

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

Poznámka: Registr Docker používá standardní porty, např. 80 nebo 443. Pokud váš registr Docker používá nestandardní port, uveďte port pomocí syntaxe *host:port*. Příklad:

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

c. Nakonfigurujte tajný údaj ověření pro lokální registr Docker.

Poznámka: Tento krok je třeba provést pouze jednou.

```
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}"
```

Příkaz ukládá, a zachytává v mezipaměti, pověření registru do souboru ve vašem systému souborů v umístění `$HOME/.airgap/secrets`.

d. Nakonfigurujte globální tajný údaj stažení obrazu a **ImageContentSourcePolicy**.

i) Zkontrolujte, zda je vyžadován restart uzlu.

- V produktu Red Hat OpenShift Container Platform verze 4.4 a vyšší a v nové instalaci produktu IBM MQ Operator pomocí airgap tento krok restartuje všechny uzly klastru. Prostředky klastru mohou být nedostupné, dokud není použit nový tajný údaj stažení.
- V produktu IBM MQ Operator 1.8je CASE aktualizován tak, aby zahrnoval další zdroj zrcadlení pro obrazy. Proto se při upgradu z předchozích verzí produktu IBM MQ Operator na verzi 1.8 nebo vyšší spustí restart uzlu.
- Chcete-li zkontrolovat, zda tento krok vyžaduje restart uzlu, přidejte volbu `--dry-run` do kódu tohoto kroku. Tím se vygeneruje nejnovější soubor **ImageContentSourcePolicy** a zobrazí se v okně konzoly (**stdout**). Pokud se **ImageContentSourcePolicy** liší od konfigurovaného klastru **ImageContentSourcePolicy**, dojde k restartování.

```
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) Chcete-li nakonfigurovat tajný klíč stažení globálního obrazu a **ImageContentSourcePolicy**, spusťte kód pro tento krok bez volby `--dry-run` :

```
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. Ověřte, že je vytvořen prostředek **ImageContentSourcePolicy**.

```
oc get imageContentSourcePolicy
```

f. Volitelné: Používáte-li nezabezpečený registr, musíte přidat lokální registr do seznamu **insecureRegistries** klastru.

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

g. Ověřte stav uzlu klastru.

```
oc get nodes
```

Po uplatnění **imageContentsourcePolicy** a tajného údaje stažení globálního obrazu, můžete vidět stav uzlu jako **Ready**, **Scheduling** nebo **Disabled**. Počkejte, dokud všechny uzly nebudou zobrazovat stav **Ready**.

h. Zrcadlete obrazy do lokálního registru.

```
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. Proveďte upgrade zdroje katalogu.

Použijte stejný terminál, který provedl předchozí kroky.

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

Jak pokračovat dále

Nyní jste připraveni provést upgrade produktu IBM MQ Operator a správce front provedením jedné z následujících úloh:

- “Upgrade produktu IBM MQ Operator pomocí webové konzoly Red Hat OpenShift” na stránce 74
- “Upgrade produktu IBM MQ Operator pomocí rozhraní CLI Red Hat OpenShift” na stránce 75
- “Upgrade správce front IBM MQpomocí webové konzoly Red Hat OpenShift” na stránce 76
- “Upgrade správce front IBM MQ pomocí rozhraní CLI Red Hat OpenShift” na stránce 77
- “Upgrade správce front IBM MQ v produktu Red Hat OpenShift pomocí Platform Navigator” na stránce 78

OpenShift CP4I Upgrade produktu IBM MQ Operator pomocí webové konzoly Red Hat OpenShift

IBM MQ Operator může být upgradován pomocí Operator Hub.

Než začnete

Přihlaste se na webovou konzolu klastru Red Hat OpenShift.

Než budete moci upgradovat produkt IBM MQ Operator v prostředí airgap, musíte zrcadlit nejnovější obrazy IBM Cloud Pak for Integration . Viz téma Příprava na upgrade produktu IBM MQ Operator nebo správce front v prostředí airgap.

Postup

1. Chcete-li určit, na který kanál operátoru se má upgradovat, zkontrolujte “Podpora verze pro IBM MQ Operator” na stránce 7.
2. Volitelné: Pokud provádíte upgrade z verze produktu IBM MQ Operator starší než 1.5 na produkt IBM MQ Operator 1.5 nebo novější, musíte nejprve provést upgrade verze produktu IBM Cloud Pak foundational services.

   Další informace viz téma “Upgrade produktu IBM Cloud Pak foundational services pomocí webové konzoly Red Hat OpenShift” na stránce 74.
3. Upgragujte IBM MQ Operator. Nové hlavní nebo vedlejší verze produktu IBM MQ Operator jsou dodávány prostřednictvím nových kanálů odběru. Chcete-li provést upgrade vašeho operátora na novou hlavní nebo vedlejší verzi, budete muset aktualizovat vybraný kanál ve svém odběru produktu IBM MQ Operator.
   a) V navigačním podokně klepněte na volbu **Operátory** > **Instalované operátory**.

      Zobrazí se všechny nainstalované operátory v uvedeném projektu.
   b) Vyberte volbu **IBM MQ Operator**.
   c) Přejděte na kartu **Odběr**.
   d) Klepněte na volbu **Kanál**.

      Zobrazí se okno **Změnit kanál aktualizace odběru**.
   e) Vyberte požadovaný kanál a klepněte na tlačítko **Uložit**.

      Operátor provede upgrade na nejnovější verzi dostupnou pro nový kanál. Viz “Podpora verze pro IBM MQ Operator” na stránce 7.

Jak pokračovat dále

Pokud jste provedli upgrade na IBM Cloud Pak foundational services 3.7, je nutné všechny správce front, kteří používají licenci IBM Cloud Pak for Integration, upgradovat nebo restartovat. Další informace o tom, jak to provést, viz “Upgrade správce front IBM MQpomocí webové konzoly Red Hat OpenShift” na stránce 76.

OpenShift CP4I *Upgrade produktu IBM Cloud Pak foundational services pomocí webové konzoly Red Hat OpenShift*

Pokud provádíte upgrade z verze produktu IBM MQ Operator starší než 1.5 na produkt IBM MQ Operator 1.5 nebo novější, musíte nejprve provést upgrade verze produktu IBM Cloud Pak foundational services.

Než začnete

Poznámka: Tuto úlohu je třeba provést pouze v případě, že provádíte upgrade z verze produktu IBM MQ Operator starší než 1.5 na produkt IBM MQ Operator 1.5 nebo novější.

CP4I Pokud máte nějaké správce front, kteří používají licenci IBM Cloud Pak for Integration, pak po tomto upgradu bude pro přístup k webové konzole vyžadován restart správce front a také

se zobrazí další chyby přihlášení do webové konzoly. Tyto chyby můžete opravit upgradem nejnovější hodnotu `.spec.version` pro zvolenou verzi produktu IBM MQ po dokončení upgradu operátora.

CP4I Máte-li existující správce front a použijete IBM Cloud Pak for Integration Operations Dashboard, podívejte se před upgradem na "Implementace nebo upgrade produktu IBM MQ 9.2.2 nebo 9.2.3 s integrací Operations Dashboard v produktu IBM Cloud Pak for Integration 2021.4" na stránce 109.

Postup

1. Přihlaste se na webovou konzolu klastru Red Hat OpenShift.
2. V navigačním podokně klepněte na volbu **Operátory** > **Instalované operátory**.
   Zobrazí se všechny nainstalované operátory v uvedeném projektu.
3. Vyberte **IBM Cloud Pak foundational services Operator**. Všimněte si, že před verzí 3.7 se toto nazývalo **IBM Common Services Operator**
4. Přejděte na kartu **Odběr**.
5. Klepněte na volbu **Kanál**.
   Zobrazí se okno **Změnit kanál aktualizace odběru**.
6. Vyberte kanál **v3** a klepněte na tlačítko **Uložit**.
   Operátor IBM Cloud Pak foundational services provede upgrade na nejnovější verzi dostupnou pro nový kanál. Viz téma "Podpora verze pro IBM MQ Operator" na stránce 7.

Jak pokračovat dále

Nyní jste připraveni na Upgrade produktu IBM MQ Operator.

OpenShift CP4I Upgrade produktu IBM MQ Operator pomocí rozhraní CLI Red Hat OpenShift

Produkt IBM MQ Operator lze upgradovat z příkazového řádku.

Než začnete

Přihlaste se do svého klastru pomocí **`cloudctl login`** (pro IBM Cloud Pak for Integration) nebo **`oc login`**.

Než budete moci upgradovat produkt IBM MQ Operator v prostředí airgap, musíte zrcadlit nejnovější obrazy IBM Cloud Pak for Integration . Viz téma Příprava na upgrade produktu IBM MQ Operator nebo správce front v prostředí airgap.

Postup

1. Chcete-li určit, na který kanál operátoru se má upgradovat, zkontrolujte "Podpora verze pro IBM MQ Operator" na stránce 7.
2. Volitelné: Pokud provádíte upgrade z verze produktu IBM MQ Operator starší než 1.5 na produkt IBM MQ Operator 1.5 nebo novější, musíte nejprve provést upgrade verze produktu IBM Cloud Pak foundational services.

   Další informace viz téma "Upgrade produktu IBM Cloud Pak foundational services pomocí rozhraní CLI produktu Red Hat OpenShift" na stránce 76.
3. Upgragujte IBM MQ Operator. Nové hlavní/vedlejší verze IBM MQ Operator se dodávají prostřednictvím nových kanálů odběru. Chcete-li upgradovat operátor na novou hlavní/vedlejší verzi, budete muset aktualizovat vybraný kanál ve svém odběru IBM MQ Operator.
   a) Ujistěte se, že je k dispozici požadovaný kanál upgradu produktu IBM MQ Operator.

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

b) Chcete-li přejít na požadovaný aktualizační kanál (kde *vX.Y* je požadovaný aktualizační kanál uvedený v předchozím kroku), proveďte opravu `Subscription`.

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

Jak pokračovat dále

Pokud jste provedli upgrade na IBM Cloud Pak foundational services 3.7, je nutné všechny správce front, kteří používají licenci IBM Cloud Pak for Integration, upgradovat nebo restartovat. Další informace o tom, jak to provést, viz “Upgrade správce front IBM MQ pomocí rozhraní CLI Red Hat OpenShift” na stránce 77.

OpenShift CP4I *Upgrade produktu IBM Cloud Pak foundational services pomocí rozhraní CLI produktu Red Hat OpenShift*

Pokud provádíte upgrade z verze produktu IBM MQ Operator starší než 1.5 na produkt IBM MQ Operator 1.5 nebo novější, musíte nejprve provést upgrade verze produktu IBM Cloud Pak foundational services.

Než začnete

Poznámka: Tuto úlohu je třeba provést pouze v případě, že provádíte upgrade z verze produktu IBM MQ Operator starší než 1.5 na produkt IBM MQ Operator 1.5 nebo novější.

CP4I Pokud máte nějaké správce front, kteří používají licenci IBM Cloud Pak for Integration, pak po tomto upgradu bude pro přístup k webové konzole vyžadován restart správce front a také se zobrazí další chyby přihlášení do webové konzoly. Tyto chyby můžete opravit upgradem nejnovější hodnotu `.spec.version` pro zvolenou verzi produktu IBM MQ po dokončení upgradu operátora.

CP4I Máte-li existující správce front a použijete IBM Cloud Pak for Integration Operations Dashboard, podívejte se před upgradem na “Implementace nebo upgrade produktu IBM MQ 9.2.2 nebo 9.2.3 s integrací Operations Dashboard v produktu IBM Cloud Pak for Integration 2021.4” na stránce 109.

Postup

1. Přihlaste se do svého klastru pomocí **`cloudctl login`** (pro IBM Cloud Pak for Integration) nebo **`oc login`**.
2. Ujistěte se, že je k dispozici kanál upgradu produktu v3 IBM Cloud Pak foundational services.

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

3. Chcete-li přejít na požadovaný kanál aktualizace, proveďte opravu `Subscription`: v3

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

Jak pokračovat dále

Nyní jste připraveni na Upgrade produktu IBM MQ Operator.

OpenShift CP4I Upgrade správce front IBM MQpomocí webové konzoly Red Hat OpenShift

Správce front IBM MQ implementovaný pomocí IBM MQ Operator lze upgradovat v Red Hat OpenShift pomocí Operator Hub.

Než začnete

- Přihlaste se na webovou konzolu klastru Red Hat OpenShift.

- Ujistěte se, že IBM MQ Operator používá požadovaný kanál aktualizace. Viz téma “Upgrade produktu IBM MQ Operator a správců front” na stránce 70.

Než budete moci upgradovat správce front v prostředí airgap, musíte zrcadlit nejnovější obrazy IBM Cloud Pak for Integration . Viz téma Příprava na upgrade produktu IBM MQ Operator nebo správce front v prostředí airgap.

Postup

1. V navigačním podokně klepněte na volbu **Operátory** > **Instalované operátory**.
   Zobrazí se všechny nainstalované operátory v uvedeném projektu.
2. Vyberte volbu **IBM MQ Operator**.
   Zobrazí se okno **IBM MQ Operator**.
3. Přejděte na kartu **Správce front**.
   Zobrazí se okno **Podrobnosti správce front**.
4. Vyberte správce front, kterého chcete upgradovat.
5. Přejděte na kartu YAML.
6. V případě potřeby aktualizujte následující pole tak, aby odpovídala požadovanému upgradu verze správce front IBM MQ.
   - spec.version
   - spec.license.licence

   Viz “Podpora verze pro IBM MQ Operator” na stránce 7, kde jsou informace o mapování kanálů na verze IBM MQ Operator a verze správce front IBM MQ.
7. Uložte aktualizovaného správce front YAML.

OpenShift CP4I Upgrade správce front IBM MQ pomocí rozhraní CLI Red Hat OpenShift

Správce front IBM MQ implementovaný pomocí IBM MQ Operator lze upgradovat v Red Hat OpenShift pomocí příkazového řádku.

Než začnete

Chcete-li provést tyto kroky, musíte být administrátorem klastru.

- Přihlaste se do rozhraní příkazového řádku (CLI) produktu Red Hat OpenShift pomocí příkazu `oc login`.
- Ujistěte se, že IBM MQ Operator používá požadovaný kanál aktualizace. Viz téma “Upgrade produktu IBM MQ Operator a správců front” na stránce 70.

Než budete moci upgradovat správce front v prostředí airgap, musíte zrcadlit nejnovější obrazy IBM Cloud Pak for Integration . Viz téma Příprava na upgrade produktu IBM MQ Operator nebo správce front v prostředí airgap.

Postup

Upravte prostředek **`QueueManager`** k aktualizaci následujících polí tak, aby odpovídala požadovanému upgradu verze správce front IBM MQ.

- spec.version
- spec.license.licence

Viz “Podpora verze pro IBM MQ Operator” na stránce 7, kde jsou informace o mapování kanálů na verze IBM MQ Operator a verze správce front IBM MQ.

Zadejte následující příkaz:

```
oc edit queuemanager my_qmgr
```

Kde *my_qmgr* je název prostředku QueueManager, který chcete upgradovat.

CP4I Upgrade správce front IBM MQ v produktu Red Hat OpenShift pomocí Platform Navigator

Správce front IBM MQ implementovaný pomocí produktu IBM MQ Operator, lze upgradovat v produktu Red Hat OpenShift pomocí IBM Cloud Pak for Integration Platform Navigator.

Než začnete

- Přihlaste se k produktu IBM Cloud Pak for Integration Platform Navigator v oboru názvů, který obsahuje správce front, kterého chcete upgradovat.
- Ujistěte se, že IBM MQ Operator používá požadovaný kanál aktualizace. Viz téma “Upgrade produktu IBM MQ Operator a správců front” na stránce 70.

Než budete moci upgradovat správce front v prostředí airgap, musíte zrcadlit nejnovější obrazy IBM Cloud Pak for Integration . Viz téma Příprava na upgrade produktu IBM MQ Operator nebo správce front v prostředí airgap.

Postup

1. Na domovské stránce IBM Cloud Pak for Integration Platform Navigator klepněte na kartu **Běhová prostředí**.
2. Správci front s dostupnými upgrady mají modré **i** vedle položky **Verze**. Klepnutím na písmeno **i** zobrazíte **K dispozici je nová verze**.
3. Klepněte na tři tečky v pravém rohu správce front, kterého chcete upgradovat, a poté klepněte na volbu **Změnit verzi**.
4. V části **Vybrat nový kanál nebo verzi** vyberte požadovanou verzi upgradu.
5. Klepněte na volbu **Změnit verzi**.

Výsledky

Správce front je upgradován.

OpenShift CP4I Implementace a konfigurace správců front pomocí IBM MQ Operator

IBM MQ 9.1.5 a pozdější jsou implementovány v Red Hat OpenShift pomocí IBM MQ Operator.

Informace o této úloze

Procedura

- “Příprava projektu Red Hat OpenShift pro IBM MQ” na stránce 78.
- “Implementace správce front do klastru Red Hat OpenShift Container Platform” na stránce 80.

OpenShift CP4I Příprava projektu Red Hat OpenShift pro IBM MQ

Připravte klastr Red Hat OpenShift Container Platform tak, aby byl připraven implementovat správce front.

Procedura

- “Příprava projektu Red Hat OpenShift pro produkt IBM MQ pomocí webové konzoly Red Hat OpenShift” na stránce 79.

- “Příprava projektu produktu Red Hat OpenShift pro IBM MQ pomocí rozhraní CLI Red Hat OpenShift” na stránce 79.

Související úlohy

“Implementace správce front do klastru Red Hat OpenShift Container Platform” na stránce 80
Pomocí vlastního prostředku správce front implementujte správce front do klastru Red Hat OpenShift Container Platform.

OpenShift CP4I *Příprava projektu Red Hat OpenShift pro produkt IBM MQ pomocí webové konzoly Red Hat OpenShift*

Připravte si klastr Red Hat OpenShift Container Platform tak, aby byl připraven implementovat správce front pomocí IBM MQ Operator. Tuto úlohu by měl dokončit administrátor projektu.

Než začnete

Poznámka: Plánujete-li použít produkt IBM MQ v projektu s dalšími již nainstalovanými komponentami produktu IBM Cloud Pak for Integration, nemusíte se těmito pokyny řídit.

Přihlaste se na webovou konzolu klastru Red Hat OpenShift.

Informace o této úloze

Obrazy IBM MQ Operator se stahují z registru kontejnerů, který provádí kontrolu licenčních oprávnění. Tato kontrola vyžaduje klíč oprávnění, který je uložen v tajném údaji stažení `docker-registry`. Nemáte-li ještě klíč oprávnění, postupujte podle těchto pokynů, abyste získali klíč oprávnění a vytvořili tajný údaj stažení.

Postup

1. Získejte klíč oprávnění, který je přiřazen k vašemu ID.
   a) Přihlaste se k MyIBM Container Software Library s ID a heslem IBM přidruženým k oprávněnému softwaru.
   b) V sekci **Klíče oprávnění** vyberte **Kopírovat klíč** ke zkopírování klíče oprávnění do schránky (clipboardu).
2. Vytvořte tajný údaj obsahující váš klíč oprávnění, v projektu, kam chcete implementovat správce front.
   a) V navigačním podokně klepněte na volbu **Pracovní zátěž** > **Tajný údaj**.
   Zobrazí se stránka Tajné údaje.
   b) V rozevírací nabídce **Projekt** vyberte projekt, do kterého chcete produkt IBM MQ nainstalovat.
   c) Klepněte na tlačítko **Vytvořit** a vyberte volbu **Tajný údaj stažení obrazu**.
   d) Do pole **Název** zadejte `ibm-entitlement-key`.
   e) Do pole **Adresa serveru registru** zadejte `cp.icr.io`.
   f) Do pole **Jméno uživatele** zadejte `cp`.
   g) Do pole **Heslo** zadejte klíč oprávnění, který jste zkopírovali v předchozím kroku.
   h) Do pole **E-mail** zadejte ID IBM přidružené k oprávněnému softwaru.

Jak pokračovat dále

“Implementace správce front pomocí webové konzoly Red Hat OpenShift” na stránce 82

OpenShift CP4I *Příprava projektu produktu Red Hat OpenShift pro IBM MQ pomocí rozhraní CLI Red Hat OpenShift*

Připravte si klastr Red Hat OpenShift Container Platform tak, aby byl připraven implementovat správce front pomocí IBM MQ Operator. Tuto úlohu by měl dokončit administrátor projektu.

Než začnete

Poznámka: Plánujete-li použít produkt IBM MQ v projektu s dalšími již nainstalovanými komponentami produktu IBM Cloud Pak for Integration, nemusíte se těmito pokyny řídit.

Přihlaste se do svého klastru pomocí **`cloudctl login`** (pro IBM Cloud Pak for Integration) nebo **`oc login`**.

Informace o této úloze

Obrazy IBM MQ Operator se stahují z registru kontejnerů, který provádí kontrolu licenčních oprávnění. Tato kontrola vyžaduje klíč oprávnění, který je uložen v tajném údaji stažení `docker-registry`. Nemáte-li ještě klíč oprávnění, postupujte podle těchto pokynů, abyste získali klíč oprávnění a vytvořili tajný údaj stažení.

Postup

1. Získejte klíč oprávnění, který je přiřazen k vašemu ID.
   a) Přihlaste se k MyIBM Container Software Library s ID a heslem IBM přidruženým k oprávněnému softwaru.
   b) V sekci **Klíče oprávnění** vyberte **Kopírovat klíč** ke zkopírování klíče oprávnění do schránky (clipboardu).
2. Vytvořte tajný údaj obsahující váš klíč oprávnění, v projektu, kam chcete implementovat správce front.
   Spusťte následující příkaz, kde *<entitlement-key>* je klíč načtený v kroku 1, a *<user-email>* je ID IBM přidružené k oprávněnému softwaru.

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

Jak pokračovat dále

“Implementace správce front pomocí rozhraní CLI Red Hat OpenShift” na stránce 83

OpenShift CP4I Implementace správce front do klastru Red Hat OpenShift Container Platform

Pomocí vlastního prostředku správce front implementujte správce front do klastru Red Hat OpenShift Container Platform.

Procedura

- CP4I
  “Implementace správce front pomocí produktu IBM Cloud Pak for Integration Platform Navigator” na stránce 81.
- OpenShift
  “Implementace správce front pomocí webové konzoly Red Hat OpenShift” na stránce 82.
- OpenShift
  “Implementace správce front pomocí rozhraní CLI Red Hat OpenShift” na stránce 83.

Související úlohy

“Příklady konfigurace správce front” na stránce 84
Správce front lze konfigurovat úpravou obsahu vlastního prostředku správce front.

CP4I *Implementace správce front pomocí produktu IBM Cloud Pak for Integration Platform Navigator*

Pomocí vlastního prostředku správce front implementujte správce front do klastru Red Hat OpenShift Container Platform pomocí produktu IBM Cloud Pak for Integration Platform Navigator. Tuto úlohu by měl dokončit administrátor projektu.

Než začnete

V prohlížeči spusťte produkt IBM Cloud Pak for Integration Platform Navigator.

Jedná-li se o první implementaci správce front do tohoto projektu Red Hat OpenShift, postupujte podle kroků pro “Příprava projektu Red Hat OpenShift pro IBM MQ” na stránce 78.

Postup

1. Implementujte správce front.

   Následující příklad implementuje základního "quick start" správce front, který používá přechodné (dočasné) úložiště a vypíná zabezpečení MQ. Zprávy nebudou po restartu správce front zachovány. Konfiguraci můžete upravit tak, aby bylo možné změnit mnoho nastavení správce front.

   a) V produktu IBM Cloud Pak for Integration Platform Navigator klepněte na volbu **Administrace**, potom na **Běhová prostředí integrace**. Ve starších verzích produktu IBM Cloud Pak for Integration Platform Navigator klepněte na volbu **Běhové prostředí a instance**.

   b) Klepněte na volbu **Vytvořit instanci**.

   c) Vyberte volbu **Systém zpráv** a klepněte na tlačítko **Další**. Ve starších verzích produktu IBM Cloud Pak for Integration Platform Navigator klepněte na volbu **Správce front** a poté klepněte na tlačítko **Další**.

   Zobrazí se formulář pro vytvoření instance `QueueManager`.

   Poznámka: Můžete také klepnout na volbu **Kód** a zobrazit nebo změnit konfiguraci `QueueManager` YAML.

   d) V sekci **Podrobnosti** zkontrolujte nebo aktualizujte pole **Název** a zadejte **Obor názvů**, ve kterém se má vytvořit instance správce front.

   e) Jestliže přijmete licenční smlouvu produktu IBM Cloud Pak for Integration, změňte volbu **Přijetí licence** na hodnotu **Zapnuto**.

   Chcete-li implementovat správce front, musíte přijmout licenci.

   f) V sekci **Správce front** zkontrolujte nebo aktualizujte **Název** základního správce front. Ve starších verzích produktu IBM Cloud Pak for Integration Platform Navigator použijte sekci **Konfigurace správce front**.

   Standardně se název správce front používaného aplikacemi klienta IBM MQ bude shodovat s názvem `QueueManager`, ale s odebranými neplatnými znaky (jako jsou pomlčky).

   g) Klepněte na volbu **Vytvořit**.

   Nyní je zobrazen seznam správců front v aktuálním projektu (obor názvů). Nový správce `QueueManager` by měl mít stav `Pending`.

2. Zkontrolujte, zda je správce front spuštěn.

   Vytvoření je dokončeno, když stav `QueueManager` je `Running`.

Související úlohy

“Konfigurace trasy pro připojení ke správci front mimo klastr Red Hat OpenShift” na stránce 107
Chcete-li připojit aplikaci ke správci front IBM MQ mimo klastr Red Hat OpenShift , potřebujete trasu Red Hat OpenShift . Musíte povolit TLS ve svém správci front IBM MQ a klientské aplikaci, protože SNI je k dispozici pouze v protokolu TLS, když se používá protokol TLS 1.2 nebo vyšší. Red Hat OpenShift Container Platform Router používá SNI ke směrování požadavků na správce front IBM MQ.

“Připojení k serveru IBM MQ Console implementovanému v Red Hat OpenShift” na stránce 113

Jak se připojit k rozhraní IBM MQ Console správce front, který byl implementován do klastru Red Hat OpenShift Container Platform.

Implementace správce front pomocí webové konzoly Red Hat OpenShift

Pomocí vlastního prostředku správce front implementujte správce front do klastru Red Hat OpenShift Container Platform pomocí webové konzoly Red Hat OpenShift. Tuto úlohu by měl dokončit administrátor projektu.

Než začnete

Přihlaste se na webovou konzolu klastru Red Hat OpenShift. Budete muset vybrat existující projekt (obor názvů), který se má použít, nebo vytvořit nový projekt.

Jedná-li se o první implementaci správce front do tohoto projektu Red Hat OpenShift, postupujte podle kroků pro “Příprava projektu Red Hat OpenShift pro IBM MQ” na stránce 78.

Postup

1. Implementujte správce front.

   Následující příklad implementuje základního "quick start" správce front, který používá přechodné (dočasné) úložiště a vypíná zabezpečení MQ. Zprávy nebudou po restartu správce front zachovány. Konfiguraci můžete upravit tak, aby bylo možné změnit mnoho nastavení správce front.

   a) Ve webové konzole Red Hat OpenShift klepněte v navigačním podokně na volbu **Operátory** > **Instalované operátory**.

   b) Klepněte na volbu **IBM MQ**.

   c) Klepněte na kartu **Správce front**.

   d) Klepněte na tlačítko **Vytvořit správce front**.

   Zobrazí se editor YAML obsahující příklad YAML pro prostředek `QueueManager`.

   Poznámka: Můžete také klepnout na volbu **Upravit formulář** a zobrazit nebo změnit konfiguraci `QueueManager`.

   e) Jestliže přijmete licenční smlouvu, změňte volbu **Přijetí licence** na hodnotu **Zapnuto**.

   Produkt IBM MQ je k dispozici pod několika různými licencemi. Další informace o platných licencích viz “Odkaz na licenci pro mq.ibm.com/v1beta1” na stránce 124. Chcete-li implementovat správce front, musíte přijmout licenci.

   f) Klepněte na volbu **Vytvořit**.

   Nyní je zobrazen seznam správců front v aktuálním projektu (obor názvů). Nový správce `QueueManager` by měl být ve stavu `Pending`.

2. Zkontrolujte, zda je správce front spuštěn.

   Vytvoření je dokončeno, když stav `QueueManager` je `Running`.

Související úlohy

“Konfigurace trasy pro připojení ke správci front mimo klastr Red Hat OpenShift” na stránce 107
Chcete-li připojit aplikaci ke správci front IBM MQ mimo klastr Red Hat OpenShift , potřebujete trasu Red Hat OpenShift . Musíte povolit TLS ve svém správci front IBM MQ a klientské aplikaci, protože SNI je k dispozici pouze v protokolu TLS, když se používá protokol TLS 1.2 nebo vyšší. Red Hat OpenShift Container Platform Router používá SNI ke směrování požadavků na správce front IBM MQ.

“Připojení k serveru IBM MQ Console implementovanému v Red Hat OpenShift” na stránce 113
Jak se připojit k rozhraní IBM MQ Console správce front, který byl implementován do klastru Red Hat OpenShift Container Platform.

OpenShift CP4I *Implementace správce front pomocí rozhraní CLI Red Hat OpenShift*

Pomocí vlastního prostředku správce front implementujte správce front do klastru Red Hat OpenShift Container Platform pomocí rozhraní příkazového řádku (CLI). Tuto úlohu by měl dokončit administrátor projektu.

Než začnete

Musíte nainstalovat rozhraní příkazového řádku Red Hat OpenShift Container Platform.

Přihlaste se do svého klastru pomocí **`cloudctl login`** (pro IBM Cloud Pak for Integration) nebo **`oc login`**.

Jedná-li se o první implementaci správce front do tohoto projektu Red Hat OpenShift, postupujte podle kroků pro “Příprava projektu Red Hat OpenShift pro IBM MQ” na stránce 78.

Postup

1. Implementujte správce front.

   Následující příklad implementuje základního "quick start" správce front, který používá přechodné (dočasné) úložiště a vypíná zabezpečení MQ. Zprávy nebudou po restartu správce front zachovány. Obsah YAML můžete upravit tak, aby bylo možné změnit mnoho nastavení správce front.

   a) Vytvořit soubor YAML produktu `QueueManager`

   Chcete-li například nainstalovat základního správce front v IBM Cloud Pak for Integration, vytvořte soubor "mq-quickstart.yaml" s následujícím obsahem:

```
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"
```

   Důležité: Pokud přijmete licenční smlouvu produktu IBM Cloud Pak for Integration, změňte `accept: false` na `accept: true`. Podrobnosti o licenci viz “Odkaz na licenci pro mq.ibm.com/v1beta1” na stránce 124 .

   Tento příklad také zahrnuje webový server implementovaný se správcem front, s webovou konzolou povolenou pomocí jednotného přihlášení s produktem IBM Cloud Pak Identity and Access Manager.

   Chcete-li nainstalovat základního správce front nezávisle na produktu IBM Cloud Pak for Integration, vytvořte soubor "mq-quickstart.yaml" s následujícím obsahem:

```
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"
```

Důležité: Pokud přijmete licenční smlouvu produktu MQ, změňte `accept: false` na `accept: true`. Podrobnosti o licenci viz “Odkaz na licenci pro mq.ibm.com/v1beta1” na stránce 124 .

b) Vytvořit objekt `QueueManager`

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

2. Zkontrolujte, zda je správce front spuštěn.

   Implementaci můžete ověřit spuštěním

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

   a následnou kontrolou stavu.

   Např. spusťte

   ```
   oc describe queuemanager quickstart
   ```

   a zkontrolujte, že pole `status.Phase` udává `Running`

Související úlohy

“Konfigurace trasy pro připojení ke správci front mimo klastr Red Hat OpenShift” na stránce 107
Chcete-li připojit aplikaci ke správci front IBM MQ mimo klastr Red Hat OpenShift , potřebujete trasu Red Hat OpenShift . Musíte povolit TLS ve svém správci front IBM MQ a klientské aplikaci, protože SNI je k dispozici pouze v protokolu TLS, když se používá protokol TLS 1.2 nebo vyšší. Red Hat OpenShift Container Platform Router používá SNI ke směrování požadavků na správce front IBM MQ.

“Připojení k serveru IBM MQ Console implementovanému v Red Hat OpenShift” na stránce 113
Jak se připojit k rozhraní IBM MQ Console správce front, který byl implementován do klastru Red Hat OpenShift Container Platform.

OpenShift CP4I Příklady konfigurace správce front

Správce front lze konfigurovat úpravou obsahu vlastního prostředku správce front.

Informace o této úloze

Následující příklady vám pomohou nakonfigurovat správce front pomocí souboru QueueManager YAML.

Procedura

- “Příklad: Dodání souborů MQSC a INI” na stránce 84
- “Příklad: Konfigurace TLS” na stránce 86

OpenShift CP4I *Příklad: Dodání souborů MQSC a INI*

Tento příklad vytvoří objekt Kubernetes ConfigMap obsahující dva soubory MQSC a jeden soubor INI. Poté je implementován správce front, který zpracovává tyto soubory MQSC a INI.

Informace o této úloze

Soubory MQSC a INI lze dodat při implementaci správce front. Data MQSC a INI musí být definována v jednom nebo více souborech Kubernetes ConfigMaps a Secrets. Ty musí být vytvořeny v oboru názvů (projekt), do kterého budete implementovat správce front.

Poznámka: Tajný klíč Kubernetes by měl být použit, když soubor MQSC nebo INI obsahuje citlivá data.

Dodání MQSC a INI tímto způsobem vyžaduje IBM MQ Operator verze 1.1 nebo vyšší.

Příklad

Následující příklad vytvoří objekt Kubernetes ConfigMap obsahující dva soubory MQSC a jeden souborINI. Poté je implementován správce front, který zpracovává tyto soubory MQSC a INI.

Příklad ConfigMap - použijte následující YAML ve vašem klastru:

```
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
```

Příklad QueueManager implementujte správce front s následující konfigurací pomocí příkazového řádku nebo pomocí příkazu IBM Cloud Pak for Integration Platform Navigator:

```
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
```

Důležité: Pokud přijmete licenční smlouvu produktu IBM Cloud Pak for Integration, změňte `accept: false` na `accept: true`. Podrobnosti k licenci viz Odkaz na licenci pro mq.ibm.com/v1beta1.

Další informace:

- Správce front může být konfigurován tak, aby používal jeden objekt Kubernetes ConfigMap nebo Secret (jak ukazuje tento příklad), nebo více objektů Kubernetes Configmap a Secrets.
- Můžete zvolit použití všech dat MQSC a INI z objektu Kubernetes ConfigMap nebo Secret (jak je uvedeno v tomto příkladu) nebo můžete nakonfigurovat správce front tak, aby používal pouze dílčí sadu dostupných souborů.

- Soubory MQSC a INI se zpracovávají v abecedním pořadí podle jejich klíče. Takže `example1.mqsc` bude vždy zpracováno před `example2.mqsc`, bez ohledu na pořadí, ve kterém se objeví v konfiguraci správce front.
- Pokud má více souborů MQSC nebo INI stejný klíč napříč více objekty Kubernetes ConfigMap nebo Secret, pak je tato sada souborů zpracována dle pořadí, v němž jsou soubory definovány v konfiguraci správce front.

OpenShift CP4I Linux *Příklad: Konfigurace TLS*

Tento příklad implementuje správce front do produktu Red Hat OpenShift Container Platform pomocí IBM MQ Operator. Jednosměrná komunikace TLS je konfigurována mezi ukázkovým klientem a správcem front. Příklad demonstruje úspěšnou konfiguraci tím, že vkládá a získává zprávy.

Než začnete

Chcete-li dokončit tento příklad, musíte nejprve dokončit následující nezbytné předpoklady:

- Nainstalujte IBM MQ client a přidejte `samp/bin` a `bin` do cesty *PATH*. Potřebujete aplikace **runmqakm**, **amqsputc** a **amqsgetc**, které lze nainstalovat jako součást produktu IBM MQ client následujícím způsobem:
  - Windows Linux V případě Windows and Linux: Nainstalujte redistribuovatelného klienta IBM MQ pro operační systém z https://ibm.biz/mq92redistclients
  - macOS Pro Mac: Stáhněte a nastavte IBM MQ MacOS Toolkit: https://developer.ibm.com/tutorials/mq-macos-dev/
- Nainstalujte nástroj OpenSSL pro váš operační systém.
- V tomto příkladu vytvořte projekt/obor názvů pro Red Hat OpenShift Container Platform (OCP).
- V příkazovém řádku se přihlaste ke klastru OCP a přepněte se do výše uvedeného oboru názvů.
- Ujistěte se, že je IBM MQ Operator nainstalován a k dispozici ve výše uvedeném oboru názvů.

Informace o této úloze

Tento příklad poskytuje vlastní prostředek YAML definující správce front, který má být implementován do Red Hat OpenShift Container Platform. Obsahuje také podrobné informace o dalších krocích potřebných k implementaci správce front s povoleným protokolem TLS. Po dokončení vkládání a přijímání zpráv ověřuje, že je správce front nakonfigurován pomocí TLS.

Vytvoření soukromého klíče TLS a certifikátů pro server IBM MQ

Následující příklady kódu ukazují, jak vytvořit certifikát podepsaný svým držitelem pro správce front a jak přidat certifikát do databáze klíčů, která bude sloužit jako úložiště údajů o důvěryhodnosti pro klienta. Pokud již máte soukromý klíč a certifikát, můžete je použít místo něj.

Nezapomeňte, že certifikáty podepsané svým držitelem by měly být použity pouze pro účely vývoje.

Vytvořte soukromý klíč s vlastním podpisem a veřejný certifikát v aktuálním adresáři.

Spusťte tento příkaz:

```
openssl req -newkey rsa:2048 -nodes -keyout tls.key -subj "/CN=localhost" -x509 -days 3650
-out tls.crt
```

Přidat veřejný klíč serveru do databáze klíčů klienta.

Databáze klíčů se používá jako úložiště údajů o důvěryhodnosti pro aplikaci klienta.

Vytvořte databázi klíčů klienta:

```
runmqakm -keydb -create -db clientkey.kdb -pw password -type cms -stash
```

Přidejte dříve vygenerovaný veřejný klíč do databáze klíčů klienta:

```
runmqakm -cert -add -db clientkey.kdb -label mqservercert -file tls.crt -format ascii
-stashed
```

Nakonfigurujte certifikáty TLS pro implementaci správce front.

Takže váš správce front může odkazovat a použít klíč a certifikát, vytvořit tajný klíč TLS a odkazovat na soubory vytvořené výše. Provedete-li to tak, ujistěte se, že jste v oboru názvů, který jste vytvořili před zahájením této úlohy.

```
oc create secret tls example-tls-secret --key="tls.key" --cert="tls.crt"
```

Vytvořit mapu konfigurace obsahující příkazy MQSC

Vytvořte mapu konfigurace programu Kubernetes obsahující příkazy MQSC pro vytvoření nové fronty a kanálu SVRCONN a přidejte záznam ověření kanálu, který umožňuje přístup ke kanálu blokováním pouze těch uživatelů s názvem *nobody*.

Uvědomte si, že tento přístup by měl být používán pouze pro účely vývoje.

Ujistěte se, že jste v oboru názvů, který jste vytvořili dříve (viz Než začnete), poté zadejte následující YAML v uživatelském rozhraní OCP nebo použijte příkazový řádek.

```
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)
```

Vytvořte požadovanou trasu OCP.

Ujistěte se, že jste v oboru názvů, který jste vytvořili před započetím této úlohy, zadejte následující YAML v uživatelském rozhraní OCP nebo použijte příkazový řádek.

```
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: example-tls-route
spec:
  host: secureqmchl.chl.mq.ibm.com
  to:
    kind: Service
    name: secureqm-ibm-mq
  port:
    targetPort: 1414
  tls:
    termination: passthrough
```

Všimněte si, že Red Hat OpenShift Container Platform Router používá SNI pro směrování požadavků na správce front IBM MQ. Změníte-li název kanálu zadaný v prostředí MQSC v mapě konfigurace vytvořené dříve, musíte zde také změnit pole hostitele a v souboru CCDT vytvořeném později. Další informace viz téma “Konfigurace trasy pro připojení ke správci front mimo klastr Red Hat OpenShift” na stránce 107.

Implementujte správce front.

Důležité: V tomto příkladu použijeme proměnnou *MQSNOAUT* k zakázání autorizace ve správci front, což nám umožňuje zaměřit se na kroky potřebné k připojení klienta pomocí protokolu TLS. V produkční implementaci produktu IBM MQ se toto nedoporučuje, protože způsobuje, že se všechny připojující se aplikace mají úplná administrativní oprávnění, bez mechanismu, jak snížit oprávnění pro jednotlivé aplikace.

Vytvořte nového správce front s použitím následujícího vlastního prostředku YAML. Mějte na zřeteli, že se odkazuje na mapu konfigurace a tajný údaj vytvořené dříve, stejně jako na proměnnou *MQSNOAUT*.

Ujistěte se, že jste v oboru názvů, který jste vytvořili před zahájením této úlohy, a poté zadejte následující YAML v uživatelském rozhraní OCP pomocí příkazového řádku nebo pomocí IBM Cloud Pak for Integration Platform Navigator. Zkontrolujte, zda je uvedena správná licence, a přijměte licenci změnou `false` na `true`.

```
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
```

Zkontrolujte, zda je správce front spuštěn.

Správce front se nyní implementuje. Než budete pokračovat, potvrďte, že je ve stavu `Running`. Příklad:

```
oc get qmgr secureqm
```

Otestujte připojení ke správci front.

Chcete-li potvrdit, že je správce front nakonfigurován pro jednosměrnou komunikaci TLS, použijte ukázkové aplikace **amqsputc** a **amqsgetc**:

Najděte název hostitele správce front.

Použijte následující příkaz k vyhledání plně úplného názvu hostitele správce front pro trasu `secureqm-ibm-mq-qm`:

```
oc get routes secureqm-ibm-mq-qm
```

Zadejte podrobnosti o správci front.

Vytvořte soubor `CCDT.JSON`, který určuje podrobnosti správce front. Nahraďte hodnotu hostitele názvem hostitele z předchozího kroku.

```
{
    "channel":
    [
        {
            "name": "SECUREQMCHL",
            "clientConnection":
            {
                "connection":
                [
                    {
```

```
                        "host": "<hostname from previous step>",
                        "port": 443
                    }
                ],
                "queueManager": "SECUREQM"
            },
            "transmissionSecurity":
            {
              "cipherSpecification": "ECDHE_RSA_AES_128_CBC_SHA256"
            },
            "type": "clientConnection"
        }
    ]
}
```

Vyexportujte proměnné prostředí.

Vyexportujte následující proměnné prostředí, a to vhodným způsobem pro váš operační systém. Tyto proměnné budou načteny pomocí **amqsputc** a **amqsgetc**.

Aktualizujte cestu k souborům ve vašem systému:

```
export MQCCDTURL='<full path to file>/CCDT.JSON'
export MQSSLKEYR='<full path to file>/clientkey'
```

Vložte zprávy do fronty.

Spusťte tento příkaz:

```
amqsputc EXAMPLE.QUEUE SECUREQM
```

Je-li připojení ke správci front úspěšné, bude výstupem následující odezva:

```
target queue is EXAMPLE.QUEUE
```

Vložte několik zpráv do fronty zadáním nějakého textu a následným stisknutím klávesy **Enter**.

Chcete-li operaci dokončit, stiskněte dvakrát klávesu **Enter**.

Načtěte zprávy z fronty.

Spusťte tento příkaz:

```
amqsgetc EXAMPLE.QUEUE SECUREQM
```

Zprávy, které jste přidali v předchozím kroku, byly spotřebovány a jsou výstupem.

Po několika sekundách se příkaz ukončí.

Blahopřejeme, úspěšně jste implementovali správce front s povoleným protokolem TLS a ukázali, že můžete bezpečně vložit a načíst zprávy do správce front z klienta.

OpenShift CP4I *Příklad: Úprava anotací služby licence*

IBM MQ Operator automaticky přidá anotace IBM License Service do implementovaných prostředků. Tyto informace jsou monitorovány produktem IBM License Service a jsou generovány sestavy, které odpovídají požadovanému oprávnění.

Informace o této úloze

Anotace přidané serverem IBM MQ Operator jsou ty, které se očekávají ve standardních situacích, a jsou založeny na hodnotách licencí vybraných během implementace správce front.

Příklad

Je-li parametr **License** nastaven na hodnotu `L-RJON-BZFQU2 (IBM Cloud Pak for Integration 2021.2.1)` a parametr **Use** je nastaven na hodnotu `Neprodukční`, jsou použity následující anotace:

- cloudpakId: c8b82d189e7545f0892db9ef2731b90d
- cloudpakName: IBM Cloud Pak for Integration
- productChargedContainers: qmgr

- productCloudpakRatio: '4:1'
- productID: 21dfe9a0f00f444f888756d835334909
- productName: IBM MQ Advanced for Non-Production
- productMetric: VIRTUAL_PROCESSOR_CORE
- productVersion: 9.2.3.0

V rámci produktu IBM Cloud Pak for Integration zahrnují implementace IBM App Connect Enterprise omezené oprávnění pro IBM MQ. V těchto situacích je třeba tyto anotace potlačit, aby bylo zajištěno, že produkt IBM License Service zachytí správné použití. Chcete-li to provést, použijte přístup popsaný v tématu “Přidání vlastních anotací a štítků do prostředků správce front” na stránce 112.

Je-li například produkt IBM MQ implementován v rámci oprávnění IBM App Connect Enterprise, použijte přístup zobrazený v následujícím fragmentu kódu:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: mq4ace
  namespace: cp4i
spec:
  annotations:
    productMetric: FREE
```

Existují dva další běžné důvody, proč mohou anotace licencí vyžadovat úpravu:

1. Produkt IBM MQ Advanced je zahrnut do oprávnění jiného produktu IBM.
   - V této situaci použijte přístup popsaný dříve pro produkt IBM App Connect Enterprise.
2. Produkt IBM MQ je implementován na základě licence IBM Cloud Pak for Integration.
   - Máte-li licenci k IBM Cloud Pak for Integration, můžete rozhodnout o implementaci správce front buď v poměru IBM MQ, nebo IBM MQ Advanced. Pokud implementujete poměr IBM MQ, musíte se ujistit, že nepoužíváte žádné rozšířené schopnosti, jako je nativní HA nebo Advanced Message Security.
   - V této situaci použijte následující anotace pro provozní použití:

     ```
     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
     ```

   - Pro neprodukční použití použijte následující anotace:

     ```
     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 Konfigurace vysoké dostupnosti pro správce front pomocí IBM MQ Operator

Informace o této úloze

Procedura

- V 9.2.3 “Nativní vysoká dostupnost” na stránce 91.
- V 9.2.3 “Příklad: Konfigurace správce front nativní vysoké dostupnosti” na stránce 93.
- “Příklad: Konfigurace správce front s více instancemi” na stránce 101.

CP4I CD V 9.2.3 *Nativní vysoká dostupnost*

Nativní vysoká dostupnost je řešení nativní (vestavěné) vysoké dostupnosti pro produkt IBM MQ, které je vhodné pro použití s úložištěm bloků cloudu.

Konfigurace nativní vysoké dostupnosti poskytuje vysoce dostupného správce front, v němž jsou obnovitelná data MQ (například zprávy) replikována do více sad úložiště, a brání tak ztrátě v důsledku selhání úložiště. Správce front se skládá z více spuštěných instancí, jednou je vedoucí, další jsou připraveni rychle převzít kontrolu v případě selhání, maximalizovat přístup ke správci front a jeho zprávám.

Konfigurace nativní vysoké dostupnosti se skládá ze tří podů Kubernetes a každá s instancí správce front. Jedna instance je aktivním správcem front, zpracovává zprávy a zapisuje do svého protokolu pro zotavení. Kdykoli je zapsán protokol pro zotavení, aktivní správce front odešle data ostatním dvěma instancím, které jsou známy jako repliky. Každá replika zapisuje do svého vlastního protokolu pro zotavení, potvrzuje data a pak aktualizuje vlastní data fronty z replikovaného protokolu pro zotavení. Pokud se pod spuštěním aktivního správce front nezdaří, jedna z instancí repliky správce front převezme aktivní roli a bude mít aktuální data, se kterými bude pracovat.

Typ protokolu je známý jako 'replicated log'. Replikovaný protokol je v podstatě lineární protokol, s povoleným automatickým správou protokolů a automatickými obrazy médií. Viz Typy protokolování. Použijte stejné techniky pro správu replikovaného protokolu, které používáte pro správu lineárního protokolu.

Služba Kubernetes se používá ke směrování připojení klienta TCP/IP k aktuální aktivní instanci, která je identifikována jako jediný pod, který je připraven pro provoz na síti. K tomu dojde bez nutnosti, aby aplikace klienta byla informována o různých instancích.

Tři pody se používají k výraznému snížení možnosti vzniku situace známé jako "split-brain". Ve dvoupodovém systému s vysokou dostupností může k rozštěpení split-brain dojít, když se přeruší propojení mezi dvěma pody. Při absenci konektivity mohou současně oba pody spustit správce front a shromáždit odlišná data. Po obnově připojení by mohly být dvě různé verze dat ('split-brain') a bylo bynutné ručním zásahem rozhodnout, která datová sada se má zachovat a která vyřadit.

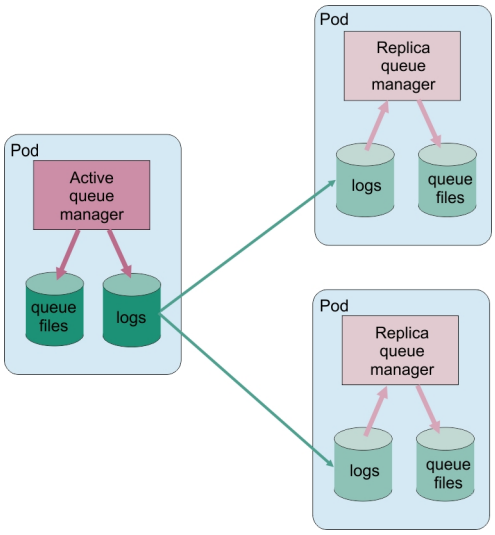
Nativní vysoká dostupnost používá třípodový systém s kvótou, aby se zabránilo situaci split-brain. Pody, které mohou komunikovat alespoň s jedním z ostatních podů, tvoří kvorum. Správce front se může stát pouze aktivní instancí v podu, který má kvorum. Správce front nemůže být aktivní v podu, který není připojen k alespoň jednomu podu, takže v daném okamžiku nemohou existovat dvě aktivní instance:

- Dojde-li k nezdaru jednoho podu, může správce front na jednom z ostatních dvou podů převzít řízení. Jestliže se nezdaří dva pody, správce front se nemůže stát aktivní instancí ve zbývajícím podu, protože pod nemá kvorum (zbývající pod nemůže určit, zda ostatní dva pody se nezdařily nebo jsou stále spuštěny a bylo ztraceno připojení).
- Pokud jeden pod ztratí připojení, nemůže být správce front aktivní instancí v tomto podu, protože pod nemá kvorum. Správce front na jednom ze zbývajících dvou podů může převzít řízení, které má kvorum.

Jestliže všechny pody ztratily připojení, správce front se nemůže stát aktivní instancí na některém z podů, protože žádný z podů nemá kvorum.

Pokud se aktivní pod nezdaří a následně se obnoví, může se znovu připojit ke skupině v roli repliky.

Na následujícím obrázku je znázorněna typická implementace se třemi instancemi správce front implementovaného ve třech kontejnerech.



Obrázek 1. Příklad konfigurace nativní vysoké dostupnosti

CP4I CD V 9.2.3 *Konfigurace nativní vysoké dostupnosti pomocí IBM MQ Operator*

Nativní vysoká dostupnost je nakonfigurována pomocí rozhraní API `QueueManager` a rozšířené volby jsou k dispozici prostřednictvím souboru INI.

Nativní vysoká dostupnost je nakonfigurována pomocí `.spec.queueManager.availability` rozhraní API `QueueManager`, například:

```
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
```

Pole `.spec.queueManager.availability.type` musí být nastaveno na `NativeHA`.

Nativní vysoká dostupnost je k dispozici v produktu IBM MQ 9.2.3 nebo vyšší.

Pod položkou `.spec.queueManager.availability` můžete také nakonfigurovat utajený údaj TLS a šifry, které mají být použity mezi instancemi správce front při replikaci. Tento postup se důrazně doporučuje a podrobný průvodce je k dispozici v “Příklad: Konfigurace správce front nativní vysoké dostupnosti” na stránce 93.

Související odkazy

“Příklad: Konfigurace správce front nativní vysoké dostupnosti” na stránce 93
Tento příklad ukazuje, jak implementovat správce front pomocí funkce nativní vysoké dostupnosti do Red Hat OpenShift Container Platform (OCP) pomocí IBM MQ Operator.

OpenShift CP4I Linux CD V 9.2.3 *Příklad: Konfigurace správce front nativní vysoké dostupnosti*

Tento příklad ukazuje, jak implementovat správce front pomocí funkce nativní vysoké dostupnosti do Red Hat OpenShift Container Platform (OCP) pomocí IBM MQ Operator.

Než začnete

Chcete-li dokončit tento příklad, musíte nejprve dokončit následující nezbytné předpoklady:

- Nainstalujte IBM MQ client a přidejte instalované adresáře `samp/bin` a `bin` do vaší cesty *PATH*. Klient poskytuje aplikace **`runmqakm`**, **`amqsputc`** a **`amqsgetc`**, které jsou vyžadovány tímto příkladem. Nainstalujte IBM MQ client následujícím způsobem:
  - Windows Linux V případě Windows and Linux: Nainstalujte redistribuovatelného klienta IBM MQ pro operační systém z https://ibm.biz/mq92redistclients
  - macOS Pro Mac: Stáhněte a nastavte IBM MQ MacOS Toolkit. Viz téma https://ibm.biz/mqdevmacclient.
- Nainstalujte nástroj OpenSSL pro váš operační systém. Chcete-li generovat certifikát podepsaný držitelem pro správce front, pokud již nemáte soukromý klíč a certifikát, musíte jej vygenerovat.
- V tomto příkladu vytvořte projekt/obor názvů produktu Red Hat OpenShift Container Platform (OCP) a postupujte podle kroků v úloze “Příprava projektu Red Hat OpenShift pro IBM MQ” na stránce 78
- V příkazovém řádku se přihlaste ke klastru OCP a přepněte se do právě vytvořeného oboru názvů.
- Ujistěte se, že je IBM MQ Operator nainstalován a k dispozici v oboru názvů.
- Nakonfigurujte výchozí paměťovou třídu v rámci OCP pro použití správcem front. Chcete-li dokončit tento výukový program bez nastavení výchozí paměťové třídy, viz Poznámka 2: Použití paměťové třídy jiné než výchozí.

Informace o této úloze

Správci front nativní vysoké dostupnosti zahrnují aktivní pody a dva pody repliky Kubernetes. Jsou spuštěny jako součást stavové sady Kubernetes s přesně třemi replikami a sadou trvalých svazků Kubernetes. Další informace o správcích front nativní vysoké dostupnosti viz “Vysoká dostupnost pro IBM MQ v kontejnerech” na stránce 16.

Příklad poskytuje vlastní prostředek YAML definující správce front nativní vysoké dostupnosti, který používá trvalé úložiště a je nakonfigurovaný s protokolem TLS. Po implementaci správce front do OCP se simuluje selhání aktivního podu správce front. Uvidíte uskutečnění automatické obnovy, a zadáním a přijetím zpráv po selhání prověříte úspěšné provedení.

Příklad

Vytvořte soukromý klíč a certifikáty TLS pro server MQ

Můžete vytvořit certifikát podepsaný svým držitelem pro správce front a přidat certifikát do databáze klíčů, která bude sloužit jako úložiště údajů o důvěryhodnosti pro klienta. Pokud již máte soukromý klíč a certifikát, můžete je použít místo něj. Nezapomeňte, že pro účely vývoje byste měli používat pouze certifikáty podepsané svým držitelem.

Chcete-li vytvořit soukromý klíč podepsaný svým držitelem a veřejný certifikát v aktuálním adresáři, spusťte následující příkaz:

```
openssl req -newkey rsa:2048 -nodes -keyout tls.key -subj "/CN=localhost" -x509 -days 3650
-out tls.crt
```

Vytvořte soukromý klíč a certifikáty TLS pro interní použití nativní vysokou dostupností

Tři pody ve správci front nativní vysoké dostupnosti replikují data po síti. Můžete vytvořit certifikát podepsaný držitelem, který se použije při interní replikaci. Nezapomeňte, že pro účely vývoje byste měli používat pouze certifikáty podepsané svým držitelem.

Chcete-li vytvořit soukromý klíč podepsaný svým držitelem a veřejný certifikát v aktuálním adresáři, spusťte následující příkaz:

```
openssl req -newkey rsa:2048 -nodes -keyout nativeha.key -subj "/CN=localhost" -x509 -days
3650 -out nativeha.crt
```

Přidejte veřejný klíč správce front do databáze klíčů klienta

Databáze klíčů klienta se používá jako úložiště údajů o důvěryhodnosti pro aplikaci klienta.

Vytvořte databázi klíčů klienta:

```
runmqakm -keydb -create -db clientkey.kdb -pw password -type cms -stash
```

Přidejte dříve vygenerovaný veřejný klíč do databáze klíčů klienta:

```
runmqakm -cert -add -db clientkey.kdb -label mqservercert -file tls.crt -format ascii
-stashed
```

Vytvořte tajný klíč obsahující certifikáty TLS pro implementaci správce front

Takže váš správce front může odkazovat a použít klíč a certifikát, vytvořit tajný klíč TLS Kubernetes a odkazovat na soubory vytvořené výše. Provedete-li to tak, ujistěte se, že jste v oboru názvů, který jste vytvořili před zahájením této úlohy.

```
oc create secret tls example-ha-secret --key="tls.key" --cert="tls.crt"
```

Vytvořte tajný klíč obsahující interní certifikát a klíč TLS nativní vysoké dostupnosti

Takže váš správce front může odkazovat a použít klíč a certifikát, vytvořit tajný klíč TLS Kubernetes a odkazovat na soubory vytvořené výše. Provedete-li to tak, ujistěte se, že jste v oboru názvů, který jste vytvořili před zahájením této úlohy.

```
oc create secret tls example-ha-secret-internal --key="nativeha.key" --cert="nativeha.crt"
```

Vytvořit mapu konfigurace obsahující příkazy MQSC

Vytvořte mapu konfigurace programu Kubernetes obsahující příkazy MQSC pro vytvoření nové fronty a kanálu SVRCONN a přidejte záznam ověření kanálu, který umožňuje přístup ke kanálu blokováním pouze těch uživatelů s názvem *nobody*.

Uvědomte si, že tento přístup by měl být používán pouze pro účely vývoje.

Ujistěte se, že jste v oboru názvů, který jste vytvořili dříve (viz “Než začnete” na stránce 93), potom zadejte následující YAML v uživatelském rozhraní OCP nebo použijte příkazový řádek:

```
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)
```

Konfigurace směrování

Používáte-li IBM MQ client nebo sadu nástrojů v produktu IBM MQ 9.2.1 nebo novější, můžete nakonfigurovat směrování ke správci front pomocí konfiguračního souboru správce front (soubor INI). V rámci souboru nastavíte proměnnou *OutboundSNI* na trasu založenou na názvu hostitele spíše než název kanálu.

Vytvořte soubor v adresáři, ve kterém spouštíte příkazy, s názvem `mqclient.ini`, obsahující přesně tento text:

```
SSL:
   OutboundSNI=HOSTNAME
```

Neměňte žádné hodnoty v tomto souboru INI. Nesmí být například změněn řetězec HOSTNAME.

Další informace viz Sekce SSL konfiguračního souboru klienta.

Používáte-li IBM MQ client nebo sadu nástrojů starší než IBM MQ 9.2.1, musíte vytvořit trasu OCP místo předchozího konfiguračního souboru. Postupujte podle kroků v části Poznámka 1: Vytvoření trasy.

Implementujte správce front.

Důležité: V tomto příkladu použijeme proměnnou *MQSNOAUT* k zakázání autorizace ve správci front, což nám umožňuje zaměřit se na kroky potřebné k připojení klienta pomocí protokolu TLS. V produkční implementaci produktu IBM MQ se toto nedoporučuje, protože způsobuje, že se všechny připojující se aplikace mají úplná administrativní oprávnění, bez mechanismu, jak snížit oprávnění pro jednotlivé aplikace.

Zkopírujte a aktualizujte následující YAML.

- Ujistěte se, že je uvedená správná licence. Viz Odkaz na licenci pro mq.ibm.com/v1beta1. V produktu IBM Cloud Pak for Integration 2021.1.1 musí být licence licencí pro vyhodnocení `L-RJON-BYRMYW`
- Přijměte licenci změnou `false` na `true`.

Vlastní prostředek YAML správce front:

```
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
```

Ujistěte se, že jste v oboru názvů, který jste vytvořili dříve, implementujte aktualizovaný YAML, pomocí příkazového řádku nebo pomocí webové konzoly Red Hat OpenShift Container Platform, příkazového řádku nebo pomocí IBM Cloud Pak for Integration Platform Navigator.

Při konfiguraci správce front nativní vysoké dostupnosti systému existuje krátká prodleva, po jejímž uplynutí by měl být správce front k dispozici pro použití.

Ověřování

V této sekci ověřujeme, že se správce front chová podle očekávání.

Zkontrolujte, zda je správce front spuštěn.

Správce front se nyní implementuje. Než budete pokračovat, potvrďte, že je ve stavu `Running`. Příklad:

```
oc get qmgr nativeha-example
```

Otestujte připojení ke správci front.

Chcete-li potvrdit, že je správce front nakonfigurován pro jednosměrnou komunikaci TLS, použijte ukázkové aplikace **amqsputc** a **amqsgetc**:

Najděte název hostitele správce front.

Chcete-li vyhledat název hostitele správce front pro trasu `nativeha-example-ibm-mq-qm`, spusťte následující příkaz. Název hostitele je vrácen v poli `HOST`.

```
oc get routes nativeha-example-ibm-mq-qm
```

Zadejte podrobnosti o správci front.

Vytvořte soubor `CCDT.JSON`, který určuje podrobnosti správce front. Nahraďte hodnotu hostitele názvem hostitele vráceného předchozím krokem.

```
{
    "channel":
    [
        {
            "name": "HAQMCHL",
            "clientConnection":
            {
                "connection":
                [
                    {
                        "host": "<host from previous step>",
                        "port": 443
                    }
                ],
                "queueManager": "HAEXAMPLE"
            },
            "transmissionSecurity":
            {
              "cipherSpecification": "ECDHE_RSA_AES_128_CBC_SHA256"
            },
            "type": "clientConnection"
        }
   ]
}
```

Vyexportujte proměnné prostředí.

Vyexportujte následující proměnné prostředí, a to vhodným způsobem pro váš operační systém. Tyto proměnné budou načteny pomocí **amqsputc** a **amqsgetc**.

Aktualizujte cestu k souborům ve vašem systému:

```
export MQCCDTURL='<full_path_to_file>/CCDT.JSON'
export MQSSLKEYR='<full_path_to_file>/clientkey'
```

Vložte zprávy do fronty.

Spusťte tento příkaz:

```
amqsputc EXAMPLE.QUEUE HAEXAMPLE
```

Je-li připojení ke správci front úspěšné, bude výstupem následující odezva:

```
target queue is EXAMPLE.QUEUE
```

Vložte několik zpráv do fronty zadáním nějakého textu a následným stisknutím klávesy **Enter**.

Chcete-li operaci dokončit, stiskněte dvakrát klávesu **Enter**.

Načtěte zprávy z fronty.

Spusťte tento příkaz:

```
amqsgetc EXAMPLE.QUEUE HAEXAMPLE
```

Zprávy, které jste přidali v předchozím kroku, byly spotřebovány a jsou výstupem.

Po několika sekundách se příkaz ukončí.

Vynuťte selhání aktivního podu

Chcete-li ověřit automatické zotavení správce front, simulujte selhání podu:

Zobrazte aktivní a pohotovostní pody

Spusťte tento příkaz:

```
oc get pods --selector app.kubernetes.io/instance=nativeha-example
```

Všimněte si, že v poli **READY** vrací aktivní sekce hodnotu 1/1, zatímco pody pro repliky vrací hodnotu 0/1.

Odstraňte aktivní pod

Spusťte následující příkaz a zadejte úplný název aktivního podu:

```
oc delete pod nativeha-example-ibm-mq-<value>
```

Zobrazte stav podu znovu

Spusťte tento příkaz:

```
oc get pods --selector app.kubernetes.io/instance=nativeha-example
```

Zobrazte stav správce front

Spusťte následující příkaz a zadejte úplný název jednoho z ostatních podů:

```
oc exec -t Pod -- dspmq -o nativeha -x -m HAEXAMPLE
```

Měli byste vidět stav, který ukazuje, že se aktivní instance změnila, např.:

```
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) ALTDATE(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
```

Vložte a získejte zprávy znovu

Poté, co se rezervní pod stane aktivním pode (tj. poté, co se hodnota pole READY stane 1/1), použijte následující příkazy znovu, jak bylo popsáno dříve, a umístěte zprávy do správce front, poté načtěte zprávy ze správce front:

```
amqsputc EXAMPLE.QUEUE HAEXAMPLE
```

```
amqsgetc EXAMPLE.QUEUE HAEXAMPLE
```

Blahopřejeme, úspěšně jste implementovali správce front nativní vysoké dostupnosti a ukázali, že se může automaticky zotavit po selhání podu.

Další informace

Poznámka 1: Vytvoření trasy

Používáte-li IBM MQ client nebo sadu nástrojů starší než IBM MQ 9.2.1, musíte vytvořit trasu.

Chcete-li vytvořit trasu, ujistěte se, že jste v oboru názvů, který jste vytvořili dříve (viz “Než začnete” na stránce 93), potom zadejte následující YAML ve webové konzole Red Hat OpenShift Container Platform nebo použijte příkazový řádek:

```
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: example-mi-route
spec:
```

```
  host: hamqchl.chl.mq.ibm.com
  to:
    kind: Service
    name: nativeha-example-ibm-mq
  port:
    targetPort: 1414
  tls:
    termination: passthrough
```

Všimněte si, že Red Hat OpenShift Container Platform Router používá SNI pro směrování požadavků na správce front IBM MQ. Změníte-li název kanálu zadaný v konfigurační mapě s příkazy MQSC, musíte zde rovněž změnit pole hostitele, a v souboru `CCDT.JSON`, který určuje podrobnosti správce front. Další informace viz téma “Konfigurace trasy pro připojení ke správci front mimo klastr Red Hat OpenShift” na stránce 107.

Poznámka 2: Použití paměťové třídy jiné než výchozí

Tento příklad očekává, že výchozí paměťová třída byla nakonfigurována v Red Hat OpenShift Container Platform, proto nejsou ve vlastním prostředku YAML správce front žádné informace o úložišti. Pokud nemáte nakonfigurovanou paměťovou třídu jako výchozí nastavení, nebo chcete použít jinou paměťovou třídu, přidejte `defaultClass: <storage_class_name>` pod `spec.queueManager.storage`.

Název paměťové třídy se musí přesně shodovat s názvem paměťové třídy, která již existuje. To znamená, že se musí shodovat s názvem vráceným příkazem `oc get storageclass`. Musí také podporovat `ReadWriteMany`. Další informace viz téma “Aspekty úložiště pro IBM MQ Operator” na stránce 11.

Související úlohy

“Zobrazení stavu správců front nativních HA pro certifikované kontejnery produktu IBM MQ” na stránce 98

V případě certifikovaných kontejnerů produktu IBM MQ můžete zobrazit stav nativních instancí vysoké dostupnosti spuštěním příkazu **dspmq** v rámci jedné z spuštěných funkcí Pods.

V 9.2.2 CP4I CD *Zobrazení stavu správců front nativních HA pro certifikované kontejnery produktu IBM MQ*

V případě certifikovaných kontejnerů produktu IBM MQ můžete zobrazit stav nativních instancí vysoké dostupnosti spuštěním příkazu **dspmq** v rámci jedné z spuštěných funkcí Pods.

Informace o této úloze

Důležité:

Chcete-li zobrazit provozní stav instance správce front, můžete použít příkaz **dspmq** v jednom ze spuštěných podů. Vrácené informace závisí na tom, zda je instance aktivní nebo zda je to replika. Informace poskytnuté aktivní instancí jsou konečné, informace z replikovaných uzlů mohou být zastaralé.

Můžete provést následující akce:

- Zobrazit, zda je instance správce front v aktuálním uzlu aktivní nebo zda je to replika.
- Zobrazit provozní stav nativní vysoké dostupnosti instance v aktuálním uzlu.
- Zobrazit provozní stav všech tří instancí v konfiguraci nativní vysoké dostupnosti.

Následující stavová pole se používají k hlášení stavu konfigurace nativní vysoké dostupnosti:

ROLE
Určuje aktuální roli instance a je jednou z hodnot `Active`, `Replica` nebo `Unknown`.

INSTANCE
Název poskytnutý pro tuto instanci správce front, když byl vytvořen pomocí volby **-lr** příkazu **crtmqm**.

INSYNC
Určuje, zda je instance v případě potřeby schopna převzít funkci aktivní instance.

QUORUM
Hlásí stav kvora ve formátu *počet_synchronizovaných_instancí*/*počet_nakonfigurovaných_instancí*.

REPLADDR
Adresa replikace instance správce front.

CONNACTV
Označuje, zda je uzel připojen k aktivní instanci.

BACKLOG
Označuje, kolik kB instance překročila.

CONNINST
Označuje, zda je pojmenovaná instance připojena k této instanci.

ALTDATE
Označuje datum, kdy byly tyto informace naposledy aktualizovány (prázdné, pokud dosud nebyly aktualizovány).

ALTTIME
Označuje čas poslední aktualizace těchto informací (prázdné, pokud dosud nebyla aktualizována).

Procedura

- Najděte si lusky, které jsou součástí vašeho správce front.

  ```
  oc get pod --selector app.kubernetes.io/instance=nativeha-qm
  ```

- Spusťte produkt dspmq v jednom z podů

  ```
  oc exec -t Pod dspmq
  ```

  ```
  oc rsh Pod
  ```

  pro interaktivní shell, kde můžete spustit produkt dspmq přímo.
- Chcete-li určit, zda je instance správce front spuštěna jako aktivní instance nebo jako replika:

  ```
  oc exec -t Pod dspmq -o status -m QMgrName
  ```

  Aktivní instance správce front s názvem BOB bude vykazovat následující stav:

  ```
  QMNAME(BOB)                STATUS(Running)
  ```

  Instance repliky správce front s názvem BOB bude vykazovat následující stav:

  ```
  QMNAME(BOB)                STATUS(Replica)
  ```

  Neaktivní instance bude hlásit následující stav:

  ```
  QMNAME(BOB)                STATUS(Ended Immediately)
  ```

- Chcete-li určit provozní stav nativní vysoké dostupnosti instance v uvedeném modulu:

  ```
  oc exec -t Pod dspmq -o nativeha -m QMgrName
  ```

  Aktivní instance správce front s názvem BOB může vykazovat následující stav:

  ```
  QMNAME(BOB)              ROLE(Active) INSTANCE(inst1) INSYNC(Yes) QUORUM(3/3)
  ```

  Instance repliky správce front s názvem BOB může vykazovat následující stav:

  ```
  QMNAME(BOB)              ROLE(Replica) INSTANCE(inst2) INSYNC(Yes) QUORUM(2/3)
  ```

  Neaktivní instance správce front s názvem BOB může vykazovat následující stav:

```
QMNAME(BOB)               ROLE(Unknown) INSTANCE(inst3) INSYNC(no) QUORUM(0/3)
```

- Chcete-li určit provozní stav nativní vysoké dostupnosti všech instancí v konfiguraci nativní vysoké dostupnosti:

```
oc exec -t Pod dspmq -o nativeha -x -m QMgrName
```

Pokud tento příkaz zadáte na uzlu, na kterém je spuštěna aktivní instance správce front BOB, můžete obdržet následující stav:

```
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) ALTDATE(2022-01-12) ALTTIME(12.03.44)
 INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
 INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
```

Pokud tento příkaz zadáte na uzlu, na kterém je spuštěna instance repliky správce front BOB, můžete obdržet následující stav, který znamená, že jedna z replik zaostává:

```
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) ALTDATE(2022-01-12) ALTTIME(12.03.44)
 INSTANCE(inst1) ROLE(Active)  REPLADDR(9.20.123.45) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
 INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(No)  BACKLOG(435)
CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
```

Pokud tento příkaz zadáte na uzlu, kde je spuštěna neaktivní instance správce front BOB, můžete obdržet následující stav:

```
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) ALTDATE() ALTTIME()
 INSTANCE(inst2) ROLE(Unknown) REPLADDR(9.20.123.46) CONNACTV(Unknown) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATE() ALTTIME()
 INSTANCE(inst3) ROLE(Unknown) REPLADDR(9.20.123.47) CONNACTV(No) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATE() ALTTIME()
```

Pokud zadáte příkaz, když se instance ještě domlouvají, která je aktivní a které jsou repliky, obdržíte následující stav:

```
QMNAME(BOB)              STATUS(Negotiating)
```

Související odkazy

dspmq (display queue managers) command

“Příklad: Konfigurace správce front nativní vysoké dostupnosti” na stránce 93

Tento příklad ukazuje, jak implementovat správce front pomocí funkce nativní vysoké dostupnosti do Red Hat OpenShift Container Platform (OCP) pomocí IBM MQ Operator.

CP4I CD V 9.2.3 *Rozšířené vyladění pro nativní vysokou dostupnost*

Rozšířené nastavení pro ladění časování a intervalů. Tato nastavení by se nemusela použít, pokud není známo, že výchozí hodnoty neodpovídají požadavkům vašeho systému.

Základní volby pro konfiguraci Nativní vysoké dostupnosti jsou zpracovávány pomocí rozhraní API `QueueManager`, které produkt IBM MQ Operator používá ke konfiguraci základních souborů INI správce front. Existuje několik dalších rozšířených možností, které lze konfigurovat pouze pomocí souboru INI, v sekci NativeHALocalInstance. Další informace o tom, jak konfigurovat soubor INI, naleznete v tématu “Příklad: Dodání souborů MQSC a INI” na stránce 84 .

`HeartbeatInterval`

Interval prezenčního signálu definuje, jak často, v milisekundách, aktivní instance správce front nativní vysoké dostupnosti odešle synchronizaci sítě. Platný rozsah hodnoty intervalu prezenčního signálu je 500 (0,5 s) do 60000 (1 min), hodnota mimo tento rozsah způsobí, že se správce front nespustí. Je-li

tento atribut vynechán, použije se výchozí hodnota 5000 (5 s). Každá instance musí používat stejný interval prezenčního signálu.

HeartbeatTimeout
Časový limit prezenčního signálu definuje, jak dlouho bude instance repliky správce front nativní vysoké dostupnosti čekat, než se rozhodne, že aktivní instance nereaguje. Platný rozsah hodnoty časového limitu intervalu prezenčního signálu je 500 (0,5 s) do 120000 (2 min). Hodnota časového limitu prezenčního signálu musí být větší než nebo rovna intervalu prezenčního signálu.

Neplatná hodnota způsobí, že se správce front nespustí. Je-li tento atribut vynechán, čeká replika 2 x `HeartbeatInterval` před spuštěním procesu pro výběr nové aktivní instance. Každá instance musí používat stejný časový limit prezenčního signálu.

RetryInterval
Interval opakování definuje, jak často by se měl správce front nativní vysoké dostupnosti, v milisekundách, opakovat nezdařený odkaz na replikaci. Platný rozsah intervalu opakování je 500 (0,5 s) do 120000 (2 min). Je-li tento atribut vynechán, čeká replika 2 x `HeartbeatInterval` před zopakováním nezdařeného odkazu na replikaci.

CP4I *Ukončení nativních správců front HA*
Příkaz **endmqm** můžete použít k ukončení aktivního nebo replikovaného správce front, který je součástí nativní skupiny HA.

Procedura

- Chcete-li ukončit aktivní instanci správce front, přečtěte si téma Ukončení nativních správců front HA v sekci Konfigurace této dokumentace.

V 9.2.2 CP4I CD *Vyhodnocení funkce Nativní vysoká dostupnost v IBM Cloud Pak for Integration 2021.1.1*
IBM Cloud Pak for Integration 2021.1.1 - Testovací období Nativní vysoké dostupnosti je ukončeno. Použijte aktualizovanou funkci nativní vysoké dostupnosti, která je k dispozici v IBM Cloud Pak for Integration 2021.2.1, s použitím produktu IBM MQ Operator 1.6 nebo vyšší s IBM MQ 9.2.3 nebo vyšší.

Související úlohy
“Zobrazení stavu správců front nativních HA pro certifikované kontejnery produktu IBM MQ” na stránce 98
V případě certifikovaných kontejnerů produktu IBM MQ můžete zobrazit stav nativních instancí vysoké dostupnosti spuštěním příkazu **dspmq** v rámci jedné z spuštěných funkcí Pods.

Související odkazy
“Příklad: Konfigurace správce front nativní vysoké dostupnosti” na stránce 93
Tento příklad ukazuje, jak implementovat správce front pomocí funkce nativní vysoké dostupnosti do Red Hat OpenShift Container Platform (OCP) pomocí IBM MQ Operator.

OpenShift CP4I
Příklad: Konfigurace správce front s více instancemi
Tento příklad ukazuje, jak implementovat správce front s více instancemi do Red Hat OpenShift Container Platform (OCP) pomocí IBM MQ Operator. V tomto příkladu také nakonfigurujete jednosměrnou komunikaci TLS mezi ukázkovým klientem a správcem front. Příklad demonstruje úspěšnou konfiguraci tím, že vkládá a získává zprávy před selháním simulovaného podu.

Než začnete

Chcete-li dokončit tento příklad, musíte nejprve dokončit následující nezbytné předpoklady:

- Nainstalujte IBM MQ client a přidejte instalované adresáře `samp/bin` a `bin` do vaší cesty *PATH*. Klient poskytuje aplikace **runmqakm**, **amqsputc** a **amqsgetc**, které jsou vyžadovány tímto příkladem. Nainstalujte IBM MQ client následujícím způsobem:
  - Windows Linux V případě Windows and Linux: Nainstalujte redistribuovatelného klienta IBM MQ pro operační systém z https://ibm.biz/mq92redistclients

– macOS Pro Mac: Stáhněte a nastavte IBM MQ MacOS Toolkit. Viz https://developer.ibm.com/tutorials/mq-macos-dev/.

- Nainstalujte nástroj OpenSSL pro váš operační systém. Chcete-li generovat certifikát podepsaný držitelem pro správce front, pokud již nemáte soukromý klíč a certifikát, musíte jej vygenerovat.
- V tomto příkladu vytvořte projekt/obor názvů pro Red Hat OpenShift Container Platform (OCP).
- V příkazovém řádku se přihlaste ke klastru OCP a přepněte se do výše uvedeného oboru názvů.
- Ujistěte se, že je IBM MQ Operator nainstalován a k dispozici ve výše uvedeném oboru názvů.
- Nakonfigurujte výchozí paměťovou třídu v rámci OCP pro použití správcem front. Chcete-li dokončit tento výukový program bez nastavení výchozí paměťové třídy, viz Poznámka 2: Použití paměťové třídy jiné než výchozí.

Informace o této úloze

Správci front s více instancemi zahrnují aktivní a pohotovostní pod Kubernetes. Jsou spuštěny jako součást stavové sady Kubernetes s přesně dvěma replikami a sadou trvalých svazků Kubernetes. Další informace o správcích front s více instancemi viz “Vysoká dostupnost pro IBM MQ v kontejnerech” na stránce 16.

Příklad poskytuje vlastní prostředek YAML definující správce front s více instancemi s trvalým úložištěm a nakonfigurovaný s protokolem TLS. Po implementaci správce front do OCP se simuluje selhání aktivního podu správce front. Uvidíte uskutečnění automatické obnovy, a zadáním a přijetím zpráv po selhání prověříte úspěšné provedení.

Příklad

Vytvořte soukromý klíč a certifikáty TLS pro server MQ

Tato sekce dokumentuje, jak vytvořit certifikát podepsaný svým držitelem pro správce front a jak přidat certifikát do databáze klíčů, která bude sloužit jako úložiště údajů o důvěryhodnosti pro klienta. Pokud již máte soukromý klíč a certifikát, můžete je použít místo něj. Nezapomeňte, že pro účely vývoje byste měli používat pouze certifikáty podepsané svým držitelem.

Chcete-li vytvořit soukromý klíč podepsaný svým držitelem a veřejný certifikát v aktuálním adresáři, spusťte následující příkaz:

```
openssl req -newkey rsa:2048 -nodes -keyout tls.key -subj "/CN=localhost" -x509 -days 3650
-out tls.crt
```

Přidejte veřejný klíč správce front do databáze klíčů klienta

Databáze klíčů klienta se používá jako úložiště údajů o důvěryhodnosti pro aplikaci klienta.

Vytvořte databázi klíčů klienta:

```
runmqakm -keydb -create -db clientkey.kdb -pw password -type cms -stash
```

Přidejte dříve vygenerovaný veřejný klíč do databáze klíčů klienta:

```
runmqakm -cert -add -db clientkey.kdb -label mqservercert -file tls.crt -format ascii
-stashed
```

Vytvořte tajný klíč obsahující certifikáty TLS pro implementaci správce front

Takže váš správce front může odkazovat a použít klíč a certifikát, vytvořit tajný klíč TLS Kubernetes a odkazovat na soubory vytvořené výše. Provedete-li to tak, ujistěte se, že jste v oboru názvů, který jste vytvořili před zahájením této úlohy.

```
oc create secret tls example-mi-secret --key="tls.key" --cert="tls.crt"
```

Vytvořit mapu konfigurace obsahující příkazy MQSC

Vytvořte mapu konfigurace programu Kubernetes obsahující příkazy MQSC pro vytvoření nové fronty a kanálu SVRCONN a přidejte záznam ověření kanálu, který umožňuje přístup ke kanálu blokováním pouze těch uživatelů s názvem *nobody*.

Uvědomte si, že tento přístup by měl být používán pouze pro účely vývoje.

Ujistěte se, že jste v oboru názvů, který jste vytvořili dříve (viz “Než začnete” na stránce 101), potom zadejte následující YAML v uživatelském rozhraní OCP nebo použijte příkazový řádek:

```
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)
```

Konfigurace směrování

Používáte-li IBM MQ client nebo sadu nástrojů v produktu IBM MQ 9.2.1 nebo novější, můžete nakonfigurovat směrování ke správci front pomocí konfiguračního souboru správce front (soubor INI). V rámci souboru nastavíte proměnnou *OutboundSNI* na trasu založenou na názvu hostitele spíše než název kanálu.

Vytvořte soubor v adresáři, ve kterém spouštíte příkazy, s názvem ` mqclient.ini', který obsahuje následující text:

```
#* 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
```

Poznámka: Na této stránce nezměníte žádné hodnoty. Řetězec `HOSTNAME` by měl být například ponechán tak, jak je.

Další informace viz Sekce SSL konfiguračního souboru klienta.

Používáte-li IBM MQ client nebo sadu nástrojů starší než IBM MQ 9.2.1, musíte vytvořit trasu OCP místo předchozího konfiguračního souboru. Postupujte podle kroků v části Poznámka 1: Vytvoření trasy.

Implementujte správce front.

Důležité: V tomto příkladu použijeme proměnnou *MQSNOAUT* k zakázání autorizace ve správci front, což nám umožňuje zaměřit se na kroky potřebné k připojení klienta pomocí protokolu TLS. V produkční implementaci produktu IBM MQ se toto nedoporučuje, protože způsobuje, že se všechny připojující se aplikace mají úplná administrativní oprávnění, bez mechanismu, jak snížit oprávnění pro jednotlivé aplikace.

Zkopírujte a aktualizujte následující YAML.

- Ujistěte se, že je uvedená správná licence. Viz Odkaz na licenci pro mq.ibm.com/v1beta1.
- Přijměte licenci změnou `false` na `true`.
- Používáte-li IBM Cloud File Storage, podívejte se na část Poznámka 3: Použití IBM Cloud File Storage

Vlastní prostředek YAML správce front:

```
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
```

Ujistěte se, že jste v oboru názvů, který jste vytvořili dříve, naimplementujte aktualizovaný YAML v uživatelském rozhraní OCP, pomocí příkazového řádku nebo pomocí IBM Cloud Pak for Integration Platform Navigator.

Ověřování

Po krátké prodlevě by měl být správce front s více instancemi nakonfigurován a k dispozici pro použití. V této sekci ověřujeme, že se správce front chová podle očekávání.

Zkontrolujte, zda je správce front spuštěn.

Správce front se nyní implementuje. Než budete pokračovat, potvrďte, že je ve stavu `Running`. Příklad:

```
oc get qmgr miexample
```

Otestujte připojení ke správci front.

Chcete-li potvrdit, že je správce front nakonfigurován pro jednosměrnou komunikaci TLS, použijte ukázkové aplikace **amqsputc** a **amqsgetc**:

Najděte název hostitele správce front.

Chcete-li vyhledat název hostitele správce front pro trasu `miexample-ibm-mq-qm`, spusťte následující příkaz. Název hostitele je vrácen v poli `HOST`.

```
oc get routes miexample-ibm-mq-qm
```

Zadejte podrobnosti o správci front.

Vytvořte soubor `CCDT.JSON`, který určuje podrobnosti správce front. Nahraďte hodnotu hostitele názvem hostitele vráceného předchozím krokem.

```
{
    "channel":
    [
        {
            "name": "MIQMCHL",
            "clientConnection":
            {
                "connection":
                [
                    {
```

```
                        "host": "<host from previous step>",
                        "port": 443
                    }
                ],
                "queueManager": "MIEXAMPLE"
            },
            "transmissionSecurity":
            {
              "cipherSpecification": "ECDHE_RSA_AES_128_CBC_SHA256"
            },
            "type": "clientConnection"
        }
    ]
}
```

Vyexportujte proměnné prostředí.

Vyexportujte následující proměnné prostředí, a to vhodným způsobem pro váš operační systém. Tyto proměnné budou načteny pomocí **amqsputc** a **amqsgetc**.

Aktualizujte cestu k souborům ve vašem systému:

```
export MQCCDTURL='<full_path_to_file>/CCDT.JSON'
export MQSSLKEYR='<full_path_to_file>/clientkey'
```

Vložte zprávy do fronty.

Spusťte tento příkaz:

```
amqsputc EXAMPLE.QUEUE MIEXAMPLE
```

Je-li připojení ke správci front úspěšné, bude výstupem následující odezva:

```
target queue is EXAMPLE.QUEUE
```

Vložte několik zpráv do fronty zadáním nějakého textu a následným stisknutím klávesy **Enter**.

Chcete-li operaci dokončit, stiskněte dvakrát klávesu **Enter**.

Načtěte zprávy z fronty.

Spusťte tento příkaz:

```
amqsgetc EXAMPLE.QUEUE MIEXAMPLE
```

Zprávy, které jste přidali v předchozím kroku, byly spotřebovány a jsou výstupem.

Po několika sekundách se příkaz ukončí.

Vynuťte selhání aktivního podu

Chcete-li ověřit automatické zotavení správce front s více instancemi, simulujte selhání podu:

Zobrazte aktivní a pohotovostní pody

Spusťte tento příkaz:

```
oc get pods
```

Všimněte si, že v poli **READY** aktivní pod vrací hodnotu `1/1`, zatímco rezervní pod vrací hodnotu `0/1`.

Odstraňte aktivní pod

Spusťte následující příkaz a zadejte úplný název aktivního podu:

```
oc delete pod miexample-ibm-mq-<value>
```

Zobrazte stav podu znovu

Spusťte tento příkaz:

```
oc get pods
```

Zobrazte protokol pohotovostního podu

Spusťte následující příkaz a zadejte úplný název pohotovostního podu:

```
oc logs miexample-ibm-mq-<value>
```

Měla by se zobrazit tato zpráva:

```
IBM MQ queue manager 'MIEXAMPLE' becoming the active instance.
```

Vložte a získejte zprávy znovu

Poté, co se rezervní pod stane aktivním pode (tj. poté, co se hodnota pole READY stane 1/1), použijte následující příkazy znovu, jak bylo popsáno dříve, a umístěte zprávy do správce front, poté načtěte zprávy ze správce front:

```
amqsputc EXAMPLE.QUEUE MIEXAMPLE
```

```
amqsgetc EXAMPLE.QUEUE MIEXAMPLE
```

Blahopřejeme, úspěšně jste implementovali správce front s více instancemi a ukázali, že se může automaticky zotavit po selhání podu.

Další informace

Poznámka 1: Vytvoření trasy

Používáte-li IBM MQ client nebo sadu nástrojů starší než IBM MQ 9.2.1, musíte vytvořit trasu OCP.

Chcete-li vytvořit trasu, ujistěte se, že jste v oboru názvů, který jste vytvořili dříve (viz "Než začnete" na stránce 101), potom zadejte následující YAML v uživatelském rozhraní OCP nebo použijte příkazový řádek:

```
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: example-mi-route
spec:
  host: miqmchl.chl.mq.ibm.com
  to:
    kind: Service
    name: miexample-ibm-mq
  port:
    targetPort: 1414
  tls:
    termination: passthrough
```

Všimněte si, že Red Hat OpenShift Container Platform Router používá SNI pro směrování požadavků na správce front IBM MQ. Změníte-li název kanálu zadaný v konfigurační mapě s příkazy MQSC, musíte zde rovněž změnit pole hostitele, a v souboru `CCDT.JSON`, který určuje podrobnosti správce front. Další informace viz téma "Konfigurace trasy pro připojení ke správci front mimo klastr Red Hat OpenShift" na stránce 107.

Poznámka 2: Použití paměťové třídy jiné než výchozí

Tento příklad očekává, že výchozí paměťová třída byla nakonfigurována v OCP, proto nejsou ve vlastním prostředku YAML správce front žádné informace o úložišti. Pokud nemáte nakonfigurovanou paměťovou třídu jako výchozí nastavení, nebo chcete použít jinou paměťovou třídu, přidejte `defaultClass: <storage_class_name>` pod `spec.queueManager.storage`.

Název paměťové třídy se musí přesně shodovat s názvem paměťové třídy, která existuje v systému OCP. To znamená, že se musí shodovat s názvem vráceným příkazem `oc get storageclass`. Musí také podporovat `ReadWriteMany`. Další informace viz téma "Aspekty úložiště pro IBM MQ Operator" na stránce 11.

Poznámka 3: Použití IBM Cloud File Storage

V některých situacích, například při použití IBM Cloud File Storage, je také třeba určit pole **`securityGroups`** v YAML vlastního prostředku správce front. Například přidáním následujícího podřízeného pole přímo pod `spec:`

```
securityContext:
  supplementalGroups: [99]
```

Další informace viz téma “Aspekty úložiště pro IBM MQ Operator” na stránce 11.

OpenShift CP4I CD Konfigurace trasy pro připojení ke správci front mimo klastr Red Hat OpenShift

Chcete-li připojit aplikaci ke správci front IBM MQ mimo klastr Red Hat OpenShift , potřebujete trasu Red Hat OpenShift . Musíte povolit TLS ve svém správci front IBM MQ a klientské aplikaci, protože SNI je k dispozici pouze v protokolu TLS, když se používá protokol TLS 1.2 nebo vyšší. Red Hat OpenShift Container Platform Router používá SNI ke směrování požadavků na správce front IBM MQ.

Informace o této úloze

Upozornění: Tento dokument se vztahuje na verze 9.2.1 Continuous Delivery a novější klientů IBM MQ. Používá-li klient verzi 9.2.0 Long Term Support nebo starší, podívejte se na stránku dokumentace IBM MQ 9.1 Připojení ke správci front implementovanému v klastru Red Hat OpenShift.

V 9.2.1 Požadovaná konfigurace Red Hat OpenShift Trasa závisí na chování SNI (Server Name Indication) (SNI) vaší klientské aplikace. IBM MQ podporuje dvě různá nastavení záhlaví SNI v závislosti na konfiguraci a typu klienta. Záhlaví SNI je nastaveno na název hostitele místa určení klienta nebo na název kanálu IBM MQ . Informace, jak IBM MQ mapuje název kanálu na název hostitele, viz Jak IBM MQ poskytuje schopnost s více certifikáty.

V 9.2.1 Zda je záhlaví SNI nastaveno na název kanálu IBM MQ nebo název hostitele je řízeno pomocí atributu **OutboundSNI** . Možné hodnoty jsou `OutboundSNI=CHANNEL` (výchozí hodnota) nebo `OutboundSNI=HOSTNAME`. Další informace viz Sekce SSL konfiguračního souboru klienta. Všimněte si, že `CHANNEL` a `HOSTNAME` jsou přesné hodnoty, které používáte; nejedná se o názvy proměnných, které nahradíte skutečným názvem kanálu nebo názvem hostitele.

V 9.2.1

Chování klienta s různými nastaveními `OutboundSNI`

Je-li parametr **OutboundSNI** nastaven na `HOSTNAME`, následující klienti nastaví název hostitele SNI, pokud je v názvu připojení uveden název hostitele:

- Klienti C
- Klienti .NET v nespravovaném režimu
- Klienti Java/JMS

Je-li parametr **OutboundSNI** nastaven na hodnotu `HOSTNAME` a v názvu připojení se používá adresa IP, následující klienti odesílají prázdné záhlaví SNI:

- Klienti C
- Klienti .NET v nespravovaném režimu
- Klienti Java/JMS (nelze provést zpětné vyhledání DNS názvu hostitele)

Je-li parametr **OutboundSNI** nastaven na `CHANNEL` nebo není nastaven, použije se místo něj název kanálu IBM MQ a je vždy odesláno, zda je použit název hostitele nebo název připojení s adresou IP.

Následující typy klientů nepodporují nastavení záhlaví SNI pro název kanálu IBM MQ, a tak se vždy pokuste nastavit záhlaví SNI na název hostitele bez ohledu na nastavení parametru **OutboundSNI**:

- Klienti AMQP
- Klienti XR
- Klienti .NET ve spravovaném režimu (před IBM MQ 9.2.0 Fix Pack 4 for Long Term Support a před IBM MQ 9.2.3 for Continuous Delivery.)

V 9.2.0.4 V 9.2.3

Z IBM MQ 9.2.0 Fix Pack 4 for Long Term Support a IBM MQ 9.2.3 for Continuous Delivery byli IBM MQspravovaní .NET klienti aktualizováni pro nastavení SERVERNAME na příslušný název hostitele, pokud je vlastnost **OutboundSNI** nastavena na HOSTNAME, což umožňuje IBM MQ spravovanému .NET klientovi připojit se ke správci front pomocí cest Red Hat OpenShift. Všimněte si, že v souboru IBM MQ 9.2.0 Fix Pack 4je vlastnost **OutboundSNI** přidána a podporována pouze ze souboru `mqclient.ini` ; nemůžete nastavit vlastnost z aplikace .NET.

V 9.2.5

Pokud se aplikace klienta připojuje ke správci front implementovanému v klastru Red Hat OpenShift prostřednictvím IBM MQ Internet Pass-Thru (MQIPT), může být MQIPT konfigurován tak, aby nastavil SNI na název hostitele pomocí vlastnosti SSLClientOutboundSNI v definici předepsané cesty.

OutboundSNI, více certifikátů a Red Hat OpenShift tras

Produkt IBM MQ používá záhlaví SNI k zajištění funkčnosti více certifikátů. Pokud se aplikace připojuje ke kanálu IBM MQ , který je konfigurován pro použití jiného certifikátu prostřednictvím pole CERTLABL, musí se aplikace připojit s nastavením **OutboundSNI** na hodnotu CHANNEL.

Pokud vaše konfigurace přenosové cesty Red Hat OpenShift vyžaduje HOSTNAME SNI, nemůžete použít funkci více certifikátů produktu IBM MQ a nemůžete nastavit nastavení CERTLABL na žádném objektu kanálu IBM MQ .

Pokud se aplikace s nastavením **OutboundSNI** na jinou hodnotu než CHANNEL připojí ke kanálu s nakonfigurovaným popiskem certifikátu, aplikace se odmítne s hodnotou MQRC_SSL_INITIALIZATION_ERROR a v protokolech chyb správce front se zobrazí zpráva AMQ9673 .

Další informace o tom, jak produkt IBM MQ poskytuje více funkcí certifikátu, naleznete v tématu Jak IBM MQ poskytuje více možností certifikátů .

Příklad

Aplikace klienta, které nastavují SNI na kanál MQ, vyžadují vytvoření nového Red Hat OpenShift Route pro každý kanál, ke kterému se chcete připojit. Musíte také použít jedinečné názvy kanálů napříč klastrem Red Hat OpenShift Container Platform, což umožňuje směrování do správného správce front.

Je důležité, aby se názvy kanálů MQ neukončovaly malými písmeny kvůli způsobu, jakým IBM MQ mapuje názvy kanálů na záhlaví SNI.

Chcete-li určit požadovaný název hostitele pro každý z vašich nových Red Hat OpenShift Routes, je třeba mapovat každý název kanálu na adresu SNI. Další informace viz Jak IBM MQ poskytuje schopnost s více certifikáty.

Potom musíte pro každý kanál vytvořit nový Red Hat OpenShift Route tak, že ve svém klastru použijete následující `yaml`:

```
  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
```

Konfigurace podrobností připojení aplikace klienta

Název hostitele, který má být použit pro připojení klienta, můžete určit spuštěním následujícího příkazu:

```
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}"
```

Port pro připojení klienta by měl být nastaven na port, který používá Red Hat OpenShift Container Platform Router - běžně 443.

Související úlohy

“Připojení k serveru IBM MQ Console implementovanému v Red Hat OpenShift” na stránce 113
Jak se připojit k rozhraní IBM MQ Console správce front, který byl implementován do klastru Red Hat OpenShift Container Platform.

CP4I Integrace s IBM Cloud Pak for Integration Operations Dashboard

Schopnost trasovat transakce pomocí produktu IBM Cloud Pak for Integration je poskytována produktem Operations Dashboard.

Informace o této úloze

Povolení integrace s produktem Operations Dashboard instaluje uživatelskou proceduru rozhraní MQ API do správce front. Uživatelská procedura rozhraní API odešle data trasování do datového úložiště Operations Dashboard; o zprávách, které jsou posílány prostřednictvím správce front.

Mějte na zřeteli, že jsou trasovány pouze zprávy, které jsou odesílány pomocí vazeb klienta MQ.

Všimněte si také, že pro verze IBM MQ Operator před 1.5, je-li povoleno trasování, byly obrazy agenta trasování a kolektoru implementované spolu se správcem front vždy dostupnými nejnovějšími verzemi, což může představovat nekompatibilitu, pokud nepoužíváte nejnovější verzi IBM Cloud Pak for Integration.

Postup

1. Implementujte správce front s povoleným trasováním.

   Standardně je funkce trasování zakázána.

   Implementujete-li pomocí produktu IBM Cloud Pak for Integration Platform Navigator, potom můžete povolit trasování při implementaci nastavením volby **Povolit trasování** na **Zapnuto** a nastavením **Obor názvů trasování** na obor názvů, kde je nainstalován produktu Operations Dashboard. Podrobnější informace o implementaci správce front viz “Implementace správce front pomocí produktu IBM Cloud Pak for Integration Platform Navigator” na stránce 81

   Implementujete-li pomocí rozhraní Red Hat OpenShift CLI nebo webové konzoly Red Hat OpenShift, potom můžete povolit trasování s následujícím úsekem kódu YAML:

   ```
   spec:
     tracing:
       enabled: true
       namespace: <Operations_Dashboard_Namespace
   ```

   Důležité: Správce front se nespustí, dokud produkt MQ nebude registrován s produktem Operations Dashboard (viz další krok).

   Mějte na zřeteli, že když je tato funkce povolena, spustí se kromě kontejneru správce front dva rozšiřující (tzv. sidecar) kontejnery ("Agent" a "Collector"). Obrazy pro tyto dva rozšiřující kontejnery budou k dispozici ve stejném registru jako hlavní obraz MQ a budou používat stejnou zásadu stažení a tajný údaj stažení. K dispozici jsou další nastavení konfigurace limitů CPU a paměti.
2. Pokud se jedná o prvního správce front s integrací produktu Operations Dashboard, který byl implementován v tomto oboru názvů, pak je třeba Registrace s produktem Operations Dashboard.

   Registrace vytvoří objekt Tajný údaj, který musí Pod správce front úspěšně spustit.

CP4I CD *Implementace nebo upgrade produktu IBM MQ 9.2.2 nebo 9.2.3 s integrací Operations Dashboard v produktu IBM Cloud Pak for Integration 2021.4*

Každá verze produktu IBM MQ je přidružena ke specifické verzi agenta a komponent kolektoru produktu Operations Dashboard, které jsou implementovány spolu se správcem front. Produkt IBM Cloud Pak for

Integration 2021.4.1 zavádí změnu, která způsobí, že starší agenti a komponenty kolektoru nebudou s produktem Operations Dashboard pracovat. Chcete-li to opravit, musíte přepsat verzi obrázků agenta a kolektoru Operations Dashboard, které používáte při použití IBM MQ 9.2.2 nebo 9.2.3.

Implementace nového správce front IBM MQ 9.2.2 nebo 9.2.3

Když používáte produkt IBM Cloud Pak for Integration 2021.4.1 s produktem IBM MQ 9.2.2 nebo 9.2.3, musíte ve vašem YAML produktu `QueueManager` přepsat obrazy agenta a kolektoru produktu Operations Dashboard na verze 2.4. Příklad:

```
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
```

Pokud tak neučiníte, bude váš pod `QueueManager` zablokován ve stavu `Pending`. Když upgradujete na IBM MQ 9.2.4, můžete tyto přepisy odebrat.

Upgrade na produkt IBM Cloud Pak for Integration 2021.4.1

Poznámka: Pokud zachováváte správce front IBM MQ 9.2.2 nebo 9.2.3, nedokončujte krok 3.

1. Aktualizujte produkt `QueueManager` a přepište obrazy agenta a kolektoru, jak bylo popsáno dříve.
2. Upgradujte své operátory IBM Cloud Pak for Integration, včetně Operations Dashboard a operátoru IBM MQ, jak je popsáno v “Upgrade produktu IBM MQ Operator a správců front” na stránce 70.
3. (volitelné) Chcete-li provést upgrade na produkt IBM MQ 9.2.4 nebo novější, aktualizujte produkt `QueueManager` tak, aby používal `.spec.version` pro vaši verzi produktu IBM MQ, a poté odeberte potlačení obrazů agenta a kolektoru.

OpenShift CP4I Sestavení obrazu pomocí vlastních souborů MQSC a INI s použitím rozhraní CLI Red Hat OpenShift

Pomocí propojení (pipeline) vytvoříte v platformě Red Hat OpenShift Container Platform nový kontejnerový obraz IBM MQ se soubory MQSC a INI, které mají správci front používající tento obraz aplikovat. Tuto úlohu by měl dokončit administrátor projektu.

Než začnete

Musíte nainstalovat rozhraní příkazového řádku Red Hat OpenShift Container Platform.

Přihlaste se do svého klastru pomocí **`cloudctl login`** (pro IBM Cloud Pak for Integration) nebo **`oc login`**.

Pokud nemáte v projektu produktu Red Hat OpenShift tajný klíč Red Hat OpenShift pro produkt IBM Entitled Registry, postupujte podle kroků pro “Příprava projektu Red Hat OpenShift pro IBM MQ” na stránce 78.

Postup

1. Vytvořit `ImageStream`

   Proud obrazu a jeho přidružené značky poskytují abstrakci pro odkazování na kontejnerové obrazy z produktu Red Hat OpenShift Container Platform. Proud obrazu a jeho značky vám umožňují zjistit, jaké obrazy jsou k dispozici, a ujistit se, že používáte specifický obraz, který potřebujete, i když se obraz v úložišti změní.

   ```
   oc create imagestream mymq
   ```

2. Vytvořit BuildConfig pro nový obraz

   Produkt BuildConfig umožní sestavení pro váš nový obraz, který nebude založen na oficiálních obrazech IBM, ale přidá všechny soubory MQSC nebo INI, které chcete spustit při spuštění kontejneru.

   a) Vytvořit soubor YAML definující prostředek BuildConfig

      Např. vytvořte soubor s názvem "mq-build-config.yaml" s následujícím obsahem:

```
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'
```

      Budete muset nahradit dvě místa, kde je základní produkt IBM MQ uveden, aby ukazoval na správný základní obraz pro verzi a opravu, kterou chcete použít (podrobnosti viz “Historie vydání pro IBM MQ Operator” na stránce 19). Při aplikování oprav budete muset tyto kroky zopakovat, abyste znovu sestavili obraz.

      Tento příklad vytvoří nový obraz založený na oficiálním obrazu IBM a přidá soubory s názvem „my.mqsc“ a „my.in“ do adresáře `/etc/mqm`. Všechny soubory MQSC nebo INI nalezené v tomto adresáři budou při spuštění použity kontejnerem. Soubory INI se aplikují pomocí volby **crtmqm -ii** a sloučí se s existujícími soubory INI. Soubory MQSC jsou použity v abecedním pořadí.

      Je důležité, aby byly vaše příkazy MQSC opakovatelné, protože budou spuštěny *vždy*, když se spustí správce front. To obvykle znamená přidání parametru REPLACE do všech příkazů DEFINE a přidání parametru `IGNSTATE(YES)` do všech příkazů START nebo STOP.

   b) Použijte BuildConfig na server.

```
oc apply -f mq-build-config.yaml
```

3. Spusťte sestavení k vytvoření obrazu.

   a) Spusťte sestavení.

```
oc start-build mymq
```

      Měl by se zobrazit výstup podobný tomuto:

```
build.build.openshift.io/mymq-1 started
```

   b) Zkontrolujte stav sestavení.
      Můžete například spustit následující příkaz s použitím identifikátoru sestavení vráceného v předchozím kroku:

```
oc describe build mymq-1
```

4. Implementujte správce front pomocí nového obrazu.

   Postupujte podle kroků popsaných v tématu “Implementace správce front do klastru Red Hat OpenShift Container Platform” na stránce 80 a přidejte nový vlastní obraz do YAML.

Do svého běžného YAML produktu `QueueManager` můžete přidat následující úsek YAML, kde *můj obor názvů* je projekt/obor názvů produktu Red Hat OpenShift, který používáte, a *obraz* je název obrazu, který jste vytvořili dříve (například "mymq:latest-amd64"):

```
spec:
  queueManager:
    image: image-registry.openshift-image-registry.svc:5000/my-namespace/my-image
```

Související úlohy

“Implementace správce front do klastru Red Hat OpenShift Container Platform” na stránce 80
Pomocí vlastního prostředku správce front implementujte správce front do klastru Red Hat OpenShift Container Platform.

OpenShift CP4I Přidání vlastních anotací a štítků do prostředků správce front

Do metadat správce front můžete přidávat vlastní anotace a štítky.

Informace o této úloze

Vlastní anotace a štítky jsou přidány ke všem prostředkům s výjimkou PVC. Jestliže se vlastní anotace nebo štítek shoduje s existujícím klíčem, použije se hodnota nastavená pomocí IBM MQ Operator.

Procedura

- Přidejte vlastní anotace.

  Chcete-li přidat vlastní anotace do prostředků správce front, včetně podu, přidejte anotace pod `metadata`. Příklad:

  ```
  apiVersion: mq.ibm.com/v1beta1
  kind: QueueManager
  metadata:
    name: quickstart-cp4i
    annotations:
      annotationKey: "value"
  ```

- Přidat vlastní štítky.

  Chcete-li přidat vlastní štítky do prostředků správce front, včetně podu, přidejte štítky pod `metadata`. Příklad:

  ```
  apiVersion: mq.ibm.com/v1beta1
  kind: QueueManager
  metadata:
    name: quickstart-cp4i
    labels:
      labelKey: "value"
  ```

OpenShift CP4I Zakázání běhových kontrol webhooku

Běhové kontroly webohooku zajišťují, že paměťové třídy jsou pro správce front životaschopné. Zakážete je pro zlepšení výkonu nebo protože nejsou platné pro vaše prostředí.

Informace o této úloze

Běhové kontroly webohooku jsou prováděny v konfiguraci správce front. Kontrolujete, zda jsou paměťové třídy vhodné pro vybraný typ správce front.

Můžete se rozhodnout zakázat tyto kontroly kvůli zkrácení doby potřebné pro vytvoření správce front, nebo kvůli tomu, že tyto kontroly nejsou platné pro vaše specifické prostředí.

Poznámka: Po zakázání běhových kontrol webhooku jsou povoleny všechny hodnoty paměťové třídy. V důsledku toho může dojít k poškození správce front.

Podpora běhových kontrol byla zavedena v produktu IBM MQ Operator verze 1.2.

Procedura

- Zakažte běhové kontroly webhooku.

  Přidejte následující anotaci pod `metadata`. Příklad:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: quickstart-cp4i
  annotations:
    "com.ibm.cp4i/disable-webhook-runtime-checks" : "true"
```

OpenShift CP4I Provozování produktu IBM MQ pomocí IBM MQ Operator

Informace o této úloze

Procedura

- “Příprava projektu Red Hat OpenShift pro IBM MQ” na stránce 78.
- “Implementace správce front do klastru Red Hat OpenShift Container Platform” na stránce 80.

OpenShift CP4I Připojení k serveru IBM MQ Console implementovanému v Red Hat OpenShift

Jak se připojit k rozhraní IBM MQ Console správce front, který byl implementován do klastru Red Hat OpenShift Container Platform.

Informace o této úloze

Adresu URL produktu IBM MQ Console lze nalézt na stránce s podrobnostmi `QueueManager` ve webové konzole Red Hat OpenShift nebo v příručce IBM Cloud Pak for Integration Platform Navigator. Eventuálně je možné jej nalézt z rozhraní CLI Red Hat OpenShift spuštěním následujícího příkazu:

```
oc get queuemanager <QueueManager Name> -n <namespace of your MQ deployment> --output
jsonpath='{.status.adminUiUrl}'
```

Používáte-li licenci IBM Cloud Pak for Integration, je webová konzola IBM MQ konfigurována pro použití produktu IBM Cloud PakIdentity and Access Manager (IAM). Komponenta IAM může být již nastavena administrátorem klastru. Je-li to však poprvé, co byl použit IAM na vašem klastru Red Hat OpenShift, pak budete muset načíst počáteční heslo administrátora. Další informace viz Získání počátečního hesla administrátora.

Pokud používáte licenci IBM MQ, není webová konzola MQ předkonfigurována a musíte si ji nakonfigurovat sami. Další informace viz Konfigurace uživatelů a rolí.

Související úlohy

“Konfigurace trasy pro připojení ke správci front mimo klastr Red Hat OpenShift” na stránce 107
Chcete-li připojit aplikaci ke správci front IBM MQ mimo klastr Red Hat OpenShift , potřebujete trasu Red Hat OpenShift . Musíte povolit TLS ve svém správci front IBM MQ a klientské aplikaci, protože SNI je k dispozici pouze v protokolu TLS, když se používá protokol TLS 1.2 nebo vyšší. Red Hat OpenShift Container Platform Router používá SNI ke směrování požadavků na správce front IBM MQ.

OpenShift CP4I *Udělení oprávnění pro produkt IBM MQ Console pomocí IAM produktu IBM Cloud Pak*

Oprávnění pro produkt IBM MQ Console jsou spravována prostřednictvím administrativního rozbočovače produktu IBM Cloud Pak , nikoli však IBM Cloud Pak for Integration Platform Navigator. IBM MQ nepoužívá oprávnění "Automation" poskytnutá serverem IBM Cloud Pak for Integration, ale místo toho používá základní oprávnění povolená produktem IBM Cloud Pak Identity and Access Manager (IAM).

Postup

1. Otevřete administrativní konzolu produktu IBM Cloud Pak .

   V produktu IBM Cloud Pak for Integration Platform UIklepněte na přepínač Cloud Pak (ikona 9-tečka) v pravém horním rohu panelu nástrojů a poté klepněte na panel **IBM Cloud PAk Administrativa** .
2. V navigační nabídce v levém horním rohu vyberte volbu **Identita a přístup**, poté vyberte volbu **Týmy a ID služeb**.
3. Vytvořte tým a pak do něj přidejte uživatele.
   a) Vyberte volbu **Vytvořit tým**.
   b) Zadejte název týmu, poté vyberte doménu zabezpečení pro uživatele, které chcete spravovat.
   c) Vyhledejte uživatele.

      Tito uživatelé musí existovat již ve vašem poskytovateli identity.
   d) Když najdete každého uživatele, dejte jim roli. Musí se jednat o "Administrátor" nebo "Administrátor klastrů", abyste mohli spravovat produkt IBM MQ pomocí produktu IBM MQ Console.
4. Přidejte každého uživatele do oboru názvů.
   a) Vyberte tým, který chcete upravit.
   b) Vyberte volbu **Prostředky** > **Spravovat prostředky**.
   c) Vyberte obory názvů, které má tento tým spravovat. Může se jednat o jakékoli prostory jmen ve správci front.

OpenShift CP4I Monitorování při použití produktu IBM MQ Operator

Správci front spravovaní produktem IBM MQ Operator mohou produkovat metriky kompatibilní s Prometheus.

Tyto metriky můžete zobrazit pomocí zásobníku monitorování Red Hat OpenShift Container Platform (OCP). Otevřete kartu **Metriky** v produktu OCPa poté klepněte na volbu **Sledovat** > **Metriky**. Metriky správce front jsou standardně povoleny, ale lze je vypnout nastavením hodnoty **`.spec.metrics.enabled`** na hodnotu `false`.

Prometheus je databáze časových řad a generátor vyhodnocení pravidla pro metriky. Kontejnery produktu IBM MQ vystavují koncový bod metrik, na který se může dotazovat Prometheus. Metriky jsou generovány z témat systému MQ pro monitorování a trasování aktivity.

Red Hat OpenShift Container Platform zahrnuje předkonfigurovaný, předinstalovaný zásobník a zásobník monitorování samoobslužné aktualizace používající server Prometheus. Zásobník monitorování Red Hat OpenShift Container Platform je třeba konfigurovat pro monitorování uživatelem definovaných projektů. Další informace viz Povolení monitorování pro uživatelem definované projekty. IBM MQ Operator vytvoří `ServiceMonitor`, když vytvoříte `QueueManager` s povolenými metrikami, které pak může operátor Prometheus zjistit.

Ve starších verzích IBM Cloud Pak for Integration je možné také místo toho použít službu Monitorování platformy IBM Cloud k poskytnutí serveru Prometheus.

OpenShift CP4I *Metriky publikované při použití produktu IBM MQ Operator*

Kontejnery správce front mohou publikovat metriky kompatibilní s Red Hat OpenShift Monitoring.

Metric	Typ	Popis
ibmmq_qmgr_commit_total	counter	Počet potvrzení
ibmmq_qmgr_cpu_load_fifteen_minute_average_percentage	gauge	Zatížení procesoru - průměr za patnáct minut
ibmmq_qmgr_cpu_load_five_minute_average_percentage	gauge	Zatížení procesoru - průměr za pět minut
ibmmq_qmgr_cpu_load_one_minute_average_percentage	gauge	Zatížení procesoru - průměr za jednu minutu
ibmmq_qmgr_destructive_get_bytes_total	counter	Celkový počet bajtů destruktivních operací get pro interval
ibmmq_qmgr_destructive_get_total	counter	Celkový počet destruktivních operací get pro interval
ibmmq_qmgr_durable_subscription_alter_total	counter	Počet změn trvalého odběru
ibmmq_qmgr_durable_subscription_create_total	counter	Počet vytvoření trvalého odběru
ibmmq_qmgr_durable_subscription_delete_total	counter	Počet odstranění trvalého odběru
ibmmq_qmgr_durable_subscription_resume_total	counter	Počet obnovení trvalého odběru
ibmmq_qmgr_errors_file_system_free_space_percentage	gauge	Systém souborů chyb MQ - volné místo
ibmmq_qmgr_errors_file_system_in_use_bytes	gauge	Systém souborů chyb MQ - počet používaných bajtů
ibmmq_qmgr_expired_message_total	counter	Počet zpráv s vypršenou platností
ibmmq_qmgr_failed_browse_total	counter	Počet nezdařených procházení
ibmmq_qmgr_failed_mqcb_total	counter	Počet nezdařených operací MQCB
ibmmq_qmgr_failed_mqclose_total	counter	Počet nezdařených operací MQCLOSE
ibmmq_qmgr_failed_mqconn_mqconnx_total	counter	Počet nezdařených operací MQCONN/MQCONNX

Metric	Typ	Popis
ibmmq_qmgr_failed_mqget_total	counter	Počet nezdařených operací MQGET
ibmmq_qmgr_failed_mqinq_total	counter	Počet nezdařených operací MQINQ
ibmmq_qmgr_failed_mqopen_total	counter	Počet nezdařených operací MQOPEN
ibmmq_qmgr_failed_mqput1_total	counter	Počet nezdařených operací MQPUT1
ibmmq_qmgr_failed_mqput_total	counter	Počet nezdařených operací MQPUT
ibmmq_qmgr_failed_mqset_total	counter	Počet nezdařených operací MQSET
ibmmq_qmgr_failed_mqsubrq_total	counter	Počet nezdařených operací MQSUBRQ
ibmmq_qmgr_failed_subscription_create_alter_resume_total	counter	Počet nezdařených vytvoření/změn/obnovení odběru
ibmmq_qmgr_failed_subscription_delete_total	counter	Počet nezdařených odstranění odběru
ibmmq_qmgr_failed_topic_mqput_mqput1_total	counter	Počet nezdařených operací MQPUT/MQPUT1 tématu
ibmmq_qmgr_fdc_files	gauge	Počet souborů FDC MQ
ibmmq_qmgr_log_file_system_in_use_bytes	gauge	Systém souborů protokolu - počet používaných bajtů
ibmmq_qmgr_log_file_system_max_bytes	gauge	Systém souborů protokolu - maximální počet bajtů
ibmmq_qmgr_log_in_use_bytes	gauge	Protokol - počet používaných bajtů
ibmmq_qmgr_log_logical_written_bytes_total	counter	Protokol - počet logicky zapsaných bajtů
ibmmq_qmgr_log_max_bytes	gauge	Protokol - maximální počet bajtů
ibmmq_qmgr_log_physical_written_bytes_total	counter	Protokol - počet fyzicky zapsaných bajtů
ibmmq_qmgr_log_primary_space_in_use_percentage	gauge	Protokol - aktuální používaný primární prostor

Metric	Typ	Popis
ibmmq_qmgr_log_workload_primary_space_utilization_percentage	gauge	Protokol - využití primárního prostoru pracovní zátěže
ibmmq_qmgr_log_write_latency_seconds	gauge	Protokol - latence zápisu
ibmmq_qmgr_log_write_size_bytes	gauge	Protokol - velikost zápisu
ibmmq_qmgr_mqcb_total	counter	Počet operací MQCB
ibmmq_qmgr_mqclose_total	counter	Počet operací MQCLOSE
ibmmq_qmgr_mqconn_mqconnx_total	counter	Počet operací MQCONN/MQCONNX
ibmmq_qmgr_mqctl_total	counter	Počet operací MQCTL
ibmmq_qmgr_mqdisc_total	counter	Počet operací MQDISC
ibmmq_qmgr_mqinq_total	counter	Počet operací MQINQ
ibmmq_qmgr_mqopen_total	counter	Počet operací MQOPEN
ibmmq_qmgr_mqput_mqput1_bytes_total	counter	Celkový počet bajtů operací MQPUT/MQPUT1 pro interval
ibmmq_qmgr_mqput_mqput1_total	counter	Celkový počet operací MQPUT/MQPUT1 pro interval
ibmmq_qmgr_mqset_total	counter	Počet operací MQSET
ibmmq_qmgr_mqstat_total	counter	Počet operací MQSTAT
ibmmq_qmgr_mqsubrq_total	counter	Počet operací MQSUBRQ
ibmmq_qmgr_non_durable_subscription_create_total	counter	Počet vytvoření dočasného odběru
ibmmq_qmgr_non_durable_subscription_delete_total	counter	Počet odstranění dočasného odběru
ibmmq_qmgr_non_persistent_message_browse_bytes_total	counter	Počet bajtů procházení dočasných zpráv
ibmmq_qmgr_non_persistent_message_browse_total	counter	Počet procházení dočasných zpráv

Metric	Typ	Popis
ibmmq_qmgr_non_persistent_message_destructive_get_total	counter	Počet dočasných zpráv destruktivních operací get
ibmmq_qmgr_non_persistent_message_get_bytes_total	counter	Počet bajtů získaných dočasných zpráv
ibmmq_qmgr_non_persistent_message_mqput1_total	counter	Počet dočasných zpráv operací MQPUT1
ibmmq_qmgr_non_persistent_message_mqput_total	counter	Počet dočasných zpráv operací MQPUT
ibmmq_qmgr_non_persistent_message_put_bytes_total	counter	Vložení dočasných zpráv - počet bajtů
ibmmq_qmgr_non_persistent_topic_mqput_mqput1_total	counter	Dočasné - počet operací MQPUT/MQPUT1 tématu
ibmmq_qmgr_persistent_message_browse_bytes_total	counter	Počet bajtů procházení trvalých zpráv
ibmmq_qmgr_persistent_message_browse_total	counter	Počet procházení trvalých zpráv
ibmmq_qmgr_persistent_message_destructive_get_total	counter	Počet trvalých zpráv destruktivních operací get
ibmmq_qmgr_persistent_message_get_bytes_total	counter	Počet bajtů získaných trvalých zpráv
ibmmq_qmgr_persistent_message_mqput1_total	counter	Počet trvalých zpráv operací MQPUT1
ibmmq_qmgr_persistent_message_mqput_total	counter	Počet trvalých zpráv operací MQPUT
ibmmq_qmgr_persistent_message_put_bytes_total	counter	Vložení trvalých zpráv - počet bajtů
ibmmq_qmgr_persistent_topic_mqput_mqput1_total	counter	Trvalé - počet operací MQPUT/MQPUT1 tématu
ibmmq_qmgr_published_to_subscribers_bytes_total	counter	Publikování odběratelům - počet bajtů

Metric	Typ	Popis
ibmmq_qmgr_published_to_subscribers_message_total	counter	Publikování odběratelům - počet zpráv
ibmmq_qmgr_purged_queue_total	counter	Počet vyprázdnění fronty
ibmmq_qmgr_queue_manager_file_system_free_space_percentage	gauge	Systém souborů správce front - volné místo
ibmmq_qmgr_queue_manager_file_system_in_use_bytes	gauge	Systém souborů správce front - počet používaných bajtů
ibmmq_qmgr_ram_free_percentage	gauge	Procentní část volné paměti RAM
ibmmq_qmgr_ram_usage_estimate_for_queue_manager_bytes	gauge	Celkový počet bajtů paměti RAM - odhad pro správce front
ibmmq_qmgr_rollback_total	counter	Počet odvolání
ibmmq_qmgr_system_cpu_time_estimate_for_queue_manager_percentage	gauge	Systémový čas procesoru - odhad procentní části pro správce front
ibmmq_qmgr_system_cpu_time_percentage	gauge	Procentní část systémového času procesoru
ibmmq_qmgr_topic_mqput_mqput1_total	counter	Celkový počet operací MQPUT/MQPUT1 tématu pro interval
ibmmq_qmgr_topic_put_bytes_total	counter	Celkový počet vložených bajtů tématu pro interval
ibmmq_qmgr_trace_file_system_free_space_percentage	gauge	Systém souborů trasování MQ - volné místo
ibmmq_qmgr_trace_file_system_in_use_bytes	gauge	Systém souborů trasování MQ - počet používaných bajtů
ibmmq_qmgr_user_cpu_time_estimate_for_queue_manager_percentage	gauge	Uživatelský čas procesoru - odhad procentní části pro správce front
ibmmq_qmgr_user_cpu_time_percentage	gauge	Procentní část uživatelského času procesoru

V 9.2.2 CP4I CD Zobrazení stavu správců front nativních HA pro certifikované kontejnery produktu IBM MQ

V případě certifikovaných kontejnerů produktu IBM MQ můžete zobrazit stav nativních instancí vysoké dostupnosti spuštěním příkazu **dspmq** v rámci jedné z spuštěných funkcí Pods.

Informace o této úloze

Důležité:

Chcete-li zobrazit provozní stav instance správce front, můžete použít příkaz **dspmq** v jednom ze spuštěných podů. Vrácené informace závisí na tom, zda je instance aktivní nebo zda je to replika. Informace poskytnuté aktivní instancí jsou konečné, informace z replikovaných uzlů mohou být zastaralé.

Můžete provést následující akce:

- Zobrazit, zda je instance správce front v aktuálním uzlu aktivní nebo zda je to replika.
- Zobrazit provozní stav nativní vysoké dostupnosti instance v aktuálním uzlu.
- Zobrazit provozní stav všech tří instancí v konfiguraci nativní vysoké dostupnosti.

Následující stavová pole se používají k hlášení stavu konfigurace nativní vysoké dostupnosti:

ROLE
Určuje aktuální roli instance a je jednou z hodnot `Active`, `Replica` nebo `Unknown`.

INSTANCE
Název poskytnutý pro tuto instanci správce front, když byl vytvořen pomocí volby **-lr** příkazu **crtmqm**.

INSYNC
Určuje, zda je instance v případě potřeby schopna převzít funkci aktivní instance.

QUORUM
Hlásí stav kvora ve formátu *počet_synchronizovaných_instancí*/*počet_nakonfigurovaných_instancí*.

REPLADDR
Adresa replikace instance správce front.

CONNACTV
Označuje, zda je uzel připojen k aktivní instanci.

BACKLOG
Označuje, kolik kB instance překročila.

CONNINST
Označuje, zda je pojmenovaná instance připojena k této instanci.

ALTDATE
Označuje datum, kdy byly tyto informace naposledy aktualizovány (prázdné, pokud dosud nebyly aktualizovány).

ALTTIME
Označuje čas poslední aktualizace těchto informací (prázdné, pokud dosud nebyla aktualizována).

Procedura

- Najděte si lusky, které jsou součástí vašeho správce front.

```
oc get pod --selector app.kubernetes.io/instance=nativeha-qm
```

- Spusťte produkt `dspmq` v jednom z podů

```
oc exec -t Pod dspmq
```

```
oc rsh Pod
```

pro interaktivní shell, kde můžete spustit produkt dspmq přímo.

- Chcete-li určit, zda je instance správce front spuštěna jako aktivní instance nebo jako replika:

```
oc exec -t Pod dspmq -o status -m QMgrName
```

Aktivní instance správce front s názvem BOB bude vykazovat následující stav:

```
QMNAME(BOB)                STATUS(Running)
```

Instance repliky správce front s názvem BOB bude vykazovat následující stav:

```
QMNAME(BOB)                STATUS(Replica)
```

Neaktivní instance bude hlásit následující stav:

```
QMNAME(BOB)                STATUS(Ended Immediately)
```

- Chcete-li určit provozní stav nativní vysoké dostupnosti instance v uvedeném modulu:

```
oc exec -t Pod dspmq -o nativeha -m QMgrName
```

Aktivní instance správce front s názvem BOB může vykazovat následující stav:

```
QMNAME(BOB)              ROLE(Active) INSTANCE(inst1) INSYNC(Yes) QUORUM(3/3)
```

Instance repliky správce front s názvem BOB může vykazovat následující stav:

```
QMNAME(BOB)              ROLE(Replica) INSTANCE(inst2) INSYNC(Yes) QUORUM(2/3)
```

Neaktivní instance správce front s názvem BOB může vykazovat následující stav:

```
QMNAME(BOB)              ROLE(Unknown) INSTANCE(inst3) INSYNC(no) QUORUM(0/3)
```

- Chcete-li určit provozní stav nativní vysoké dostupnosti všech instancí v konfiguraci nativní vysoké dostupnosti:

```
oc exec -t Pod dspmq -o nativeha -x -m QMgrName
```

Pokud tento příkaz zadáte na uzlu, na kterém je spuštěna aktivní instance správce front BOB, můžete obdržet následující stav:

```
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) ALTDATE(2022-01-12) ALTTIME(12.03.44)
 INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
 INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
```

Pokud tento příkaz zadáte na uzlu, na kterém je spuštěna instance repliky správce front BOB, můžete obdržet následující stav, který znamená, že jedna z replik zaostává:

```
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) ALTDATE(2022-01-12) ALTTIME(12.03.44)
 INSTANCE(inst1) ROLE(Active)  REPLADDR(9.20.123.45) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
 INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(No)  BACKLOG(435)
CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
```

Pokud tento příkaz zadáte na uzlu, kde je spuštěna neaktivní instance správce front BOB, můžete obdržet následující stav:

```
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) ALTDATE() ALTTIME()
 INSTANCE(inst2) ROLE(Unknown) REPLADDR(9.20.123.46) CONNACTV(Unknown) INSYNC(Unknown)
```

```
BACKLOG(Unknown) CONNINST(No) ALTDATE() ALTTIME()
 INSTANCE(inst3) ROLE(Unknown) REPLADDR(9.20.123.47) CONNACTV(No) INSYNC(Unknown)
BACKLOG(Unknown) CONNINST(No) ALTDATE() ALTTIME()
```

Pokud zadáte příkaz, když se instance ještě domlouvají, která je aktivní a které jsou repliky, obdržíte následující stav:

```
QMNAME(BOB)               STATUS(Negotiating)
```

Související odkazy

dspmq (display queue managers) command

“Příklad: Konfigurace správce front nativní vysoké dostupnosti” na stránce 93

Tento příklad ukazuje, jak implementovat správce front pomocí funkce nativní vysoké dostupnosti do Red Hat OpenShift Container Platform (OCP) pomocí IBM MQ Operator.

OpenShift CP4I Zálohování a obnova konfigurace správce front pomocí rozhraní Red Hat OpenShift CLI

Záloha konfigurace správce front vám může pomoci při znovusestavení správce front z jeho definic v případě, že dojde ke ztrátě konfigurace správce front. Tento postup nezálohuje data protokolu správce front. Vzhledem k přechodné povaze zpráv je pravděpodobné, že historická data protokolu budou v době obnovy bezvýznamná.

Než začnete

Přihlaste se do svého klastru pomocí **`cloudctl login`** (pro IBM Cloud Pak for Integration) nebo **`oc login`**.

Procedura

- Zazálohujte konfiguraci správce front.

  Příkaz **dmpmqcfg** můžete použít k vypsání paměti konfigurace správce front IBM MQ.

  a) Získejte název podu pro správce front.
  Můžete například spustit následující příkaz, kde *název_správce_front* je název vašeho prostředku `QueueManager`:

  ```
  oc get pods --selector app.kubernetes.io/name=ibm-mq,app.kubernetes.io/
  instance=queue_manager_name
  ```

  b) Spusťte příkaz **dmpmqcfg** na podu, směrujte výstup do souboru na svém lokálním počítači.

  Příkaz **dmpmqcfg** je výstupem konfigurace MQSC správce front.

  ```
  oc exec -it pod_name -- dmpmqcfg > backup.mqsc
  ```

- Obnovte konfiguraci správce front.

  Po provedení procedury zálohy uvedené v předchozím kroku byste měli mít soubor `backup.mqsc` obsahující konfiguraci správce front. Konfiguraci můžete obnovit tak, že tento soubor použijete pro nového správce front.

  a) Získejte název podu pro správce front.
  Můžete například spustit následující příkaz, kde *název_správce_front* je název vašeho prostředku `QueueManager`:

  ```
  oc get pods --selector app.kubernetes.io/name=ibm-mq,app.kubernetes.io/
  instance=queue_manager_name
  ```

  b) Spusťte příkaz **`runmqsc`** na podu, směrovaný do obsahu souboru `backup.mqsc`.

```
oc exec -i pod_name -- runmqsc < backup.mqsc
```

OpenShift CP4I Odstraňování problémů s produktem IBM MQ Operator

Pokud máte problémy s produktem IBM MQ Operator, mohou vám zde popsané metody pomoci při jejich diagnostice a řešení.

Procedura

- “Odstraňování problémů: Získání přístupu k datům správce front” na stránce 123

OpenShift CP4I Odstraňování problémů: Získání přístupu k datům správce front

Pomocí nástroje PVC inspector získáte přístup k souborům v PVC správce front, kde nelze vytvořit vzdálený shell pro sekci správce front. Důvodem může být skutečnost, že sekce je ve stavu **`Error`** nebo **`CrashLoopBackOff`** . Tento nástroj je navržen pro použití se správci front implementovanými produktem IBM MQ Operator.

Než začnete

Chcete-li použít nástroj PVC inspektor. musíte mít přístup k oboru názvů správce front.

Informace o této úloze

Při odstraňování problémů můžete přistupovat k datům uloženým v PVC (Persistent Volume Claims) přidružených k danému správci front. Chcete-li to provést, použijte nástroj pro připojení PVC k sadě podů inspektora. Poté můžete získat vzdálený shell do libovolného podu inspektora pro čtení souborů.

V závislosti na typu nasazení se vytvoří jedna až tři sekce inspektora. Svazky specifické pro danou sekci správce front Native-HA nebo Multi-Instance jsou k dispozici v přidružené sekci inspektoru PVC. Sdílené svazky jsou k dispozici na všech inspektorech. Název sekce inspektora obsahuje název přidružené sekce správce front.

Postup

1. Stáhněte nástroj PVC inspector produktu MQ .

   Nástroj je k dispozici zde: https://github.com/ibm-messaging/mq-pvc-tool.
2. Ujistěte se, že jste přihlášeni do svého klastru.
3. Zjistěte název správce front a obor názvů, ve kterém je spuštěn správce front.
4. Spusťte nástroj Inspektor pro vašeho správce front.
   a) Spusťte následující příkaz a zadejte název správce front a jeho obor názvů.

   ```
   ./pvc-tool.sh queue_manager_name queue_manager_namespace_name
   ```

   b) Po dokončení nástroje spusťte následující příkaz, abyste zobrazili vytvářené sekce inspektora.

   ```
   oc get pods
   ```
5. Zobrazte soubory připojené k panelu inspektora.
   a) Každá sekce inspektora PVC je přidružena k podu správce front, takže může existovat více podů inspektora. Přistupte k jedné z těchto sekcí spuštěním následujícího příkazu:

   ```
   oc rsh pvc-inspector-pod-name
   ```

   Jste umístěni do adresáře obsahujícího připojené adresáře PVC.
   b) Otevřete vzdálený shell do podu spuštěním následujícího příkazu:

```
ls
```

c) Můžete vidět adresáře se stejným názvem jako připojené PVC. Přistupte k souborům v PVC správců front procházením těchto adresářů. Chcete-li zobrazit seznam PVC, spusťte následující příkaz mimo relaci vzdáleného shellu:

```
oc get pvc
```

d) Vyčistěte sekce vytvořené nástrojem spuštěním následujícího příkazu:

```
'oc delete pods -l tool=mq-pvc-inspector
```

OpenShift CP4I Odkaz rozhraní API pro IBM MQ Operator

Produkt IBM MQ poskytuje operátor Kubernetes poskytující nativní integraci s platformou Red Hat OpenShift Container Platform.

OpenShift CP4I Odkaz rozhraní API pro mq.ibm.com/v1beta1

Rozhraní v1beta1 API lze použít k vytvoření a správě prostředků správce front.

OpenShift CP4I CD EUS *Odkaz na licenci pro mq.ibm.com/v1beta1*

Aktuální verze licencí

Pole `spec.license.license` musí obsahovat identifikátor licence pro licenci, kterou přijímáte. Platné hodnoty:

Hodnota `spec.license.license`	**Hodnota `spec.license.use`**	**Informace o licenci**	**Použitelné verze IBM MQ**
`L-RJON-C7QG3S`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration 2021.4.1	9.2.4 nebo 9.2.5
`L-RJON-C7QFZX`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration Limited Edition 2021.4.1	9.2.4 nebo 9.2.5
`L-RJON-C5CSNH`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration 2021.3.1	9.2.3 nebo 9.2.4
`L-RJON-C5CSM2`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration Limited Edition 2021.3.1	9.2.3 nebo 9.2.4
`L-RJON-BZFQU2`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration 2021.2.1	9.2.3
`L-RJON-BZFQSB`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration Limited Edition 2021.2.1	9.2.3
`L-RJON-BUVMQX`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration 2020.4.1	9.2.0 EUS nebo 9.2.1
`L-RJON-BUVMYB`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration Limited Edition 2020.4.1	9.2.0 EUS nebo 9.2.1
`L-APIG-BZDDDY`	`Production`	IBM MQ Advanced a IBM MQ Advanced pro Non-Production Environment 9.2 - 07/2021	9.2.3, 9.2.4 nebo 9.2.5

Hodnota `spec.license.license`	Hodnota `spec.license.use`	Informace o licenci	Použitelné verze IBM MQ
`L-APIG-BYHCL7`	`Development`	IBM MQ Advanced for Developers (bez záruky) V9.2 -07/2021	9.2.3, 9.2.4 nebo 9.2.5
`L-APIG-BVJJB3`	`Production`	IBM MQ Advanced a 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 nebo 9.2.1
`L-APIG-BMKG5H`	`Development`	IBM MQ Advanced for Developers (bez záruky) V9.2	9.2.0 CD, 9.2.1 nebo 9.2.2

Všimněte si, že je určena *verze* licence, což není vždy stejné jako verze produktu IBM MQ.

Starší verze licencí

Pole `spec.license.license` musí obsahovat identifikátor licence pro licenci, kterou přijímáte. Platné hodnoty:

Hodnota `spec.license.license`	Hodnota `spec.license.use`	Informace o licenci	Použitelné verze IBM MQ
`L-RJON-BXUPZ2`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration 2021.1.1	9.2.2
`L-RJON-BXUQ34`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration Limited Edition 2021.1.1	9.2.2
`L-RJON-BYRMYW`	`NonProduction`	IBM Cloud Pak for Integration Eval-Demo 2021.1.1. Úvodní vydání pro použití pouze s Nativní vysokou dostupností s IBM MQ Operator 1.5.	9.2.2
`L-RJON-BQPGWD`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration 2020.3.1	9.2.0 CD
`L-RJON-BN7PN3`	`Production` nebo `NonProduction`	IBM Cloud Pak for Integration 2020.2.1	9.1.5 nebo 9.2.0 CD
`L-RJON-BPHL2Y`	`Production` nebo `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 (bez záruky) V9.1 -04/2020	9.1.5

Všimněte si, že je určena *verze* licence, což není vždy stejné jako verze produktu IBM MQ.

Odkaz rozhraní API pro správce front (mq.ibm.com/v1beta1)

Správce front

Správce front je server IBM MQ, který poskytuje služby front a publikování/odebírání pro aplikace.

Pole	Popis
`apiVersion` string	APIVersion definuje schéma opatřené verzí této reprezentace objektu. Servery by měly převést rozpoznaná schémata na nejnovější interní hodnotu a mohou odmítnout nerozpoznané hodnoty. Další informace: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources.
`kind` string	Kind je hodnota řetězce představující prostředek REST, který tento objekt reprezentuje. Servery mohou toto odvodit z koncového bodu, na který klient odesílá požadavky. Nelze aktualizovat. Bez mezer mezi slovy a s velkými počátečními písmeny (CamelCase). Další informace: https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds.
`metadata`	
`spec` QueueManagerSpec	Požadovaný stav správce front.
`status` QueueManagerStatus	Pozorovaný stav správce front.

.spec

Požadovaný stav správce front.

Zobrazí se v:

- “Správce front” na stránce 125

Pole	Popis
`affinity`	Standardní pravidla afinity Kubernetes. Další informace viz https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.17/#affinity-v1-core.
`annotations` Anotace	Pole anotací slouží jako předávací pro anotace typu Pod. Uživatelé mohou do tohoto pole přidat libovolnou anotaci a použít ji na Pod. Zde uvedené anotace přepíší výchozí anotace, jsou-li k dispozici. Vyžaduje Operator MQ 1.3.0 nebo novější.
Pole `imagePullSecrets` LocalObjectReference	Volitelný seznam odkazů na tajné údaje ve stejném oboru názvů, které mají být použity pro stažení libovolného z obrazů používaných tímto správcem front. Je-li tato možnost určena, budou tyto tajné údaje předány jednotlivým stahujícím (puller) implementacím typu, aby je použily. Například v případě dockeru jsou uznány pouze tajné údaje typu DockerConfig. Další informace viz https://kubernetes.io/docs/concepts/containers/images#specifying-imagepullsecrets-on-a-pod.
`labels` Štítky	Pole štítků slouží jako předávací pro štítky typu Pod. Uživatelé mohou do tohoto pole přidat libovolný štítek a použít jej na Pod. Zde uvedené štítky přepíší výchozí štítky, jsou-li k dispozici. Vyžaduje Operator MQ 1.3.0 nebo novější.
`license` License	Nastavení, která řídí převzetí licence a které metriky licencí se mají používat.
`pki` PKI	Nastavení Public Key Infrastructure pro definování klíčů a certifikátů pro použití se zabezpečením Transport Layer Security (TLS) nebo MQ Advanced Message Security (AMS).
`queueManager` QueueManagerConfig	Nastavení pro kontejner správce front a základního správce front.

Pole	Popis
`securityContext` SecurityContext	Nastavení zabezpečení, které má být přidáno do kontextu securityContext podu správce front.
`template` Template	Rozšířené vytváření šablon pro prostředky Kubernetes. Šablona umožňuje uživatelům potlačit způsob, jakým produkt IBM MQ generuje základní prostředky typu Kubernetes, jako např. StatefulSet, Pods a Services. Tento postup je určen pouze pro pokročilé uživatele, protože má potenciál narušit normální provoz produktu MQ, pokud je použit nesprávně. Všechny hodnoty zadané kdekoli jinde v prostředku správce front budou přepsány nastaveními v šabloně.
`terminationGracePeriod Seconds` integer	Volitelná doba trvání v sekundách, které Pod potřebuje k řádnému ukončení. Hodnota musí být nezáporné celé číslo. Hodnota nula označuje okamžité odstranění. Cílový čas, v němž se pokouší správce front provést ukončení, eskaluje fáze odpojení aplikace. V případě potřeby jsou nezbytné úkoly údržby správce front přerušeny. Výchozí hodnota je 30 sekund.
`tracing` TracingConfig	Nastavení pro integraci trasování s produktem Cloud Pak for Integration Operations Dashboard.
`version` string	Nastavení, které řídí, jaká verze produktu MQ bude použita (povinné). Například: řetězec `9.1.5.0-r2` by specifikoval MQ verze 9.1.5.0 s druhou revizí kontejnerového obrazu. Opravy specifické pro kontejner jsou často používány v revizích, jako např. opravy v základním obrazu.
`web` WebServerConfig	Nastavení pro webový server MQ.

.spec.annotations

Pole anotací slouží jako předávací pro anotace typu Pod. Uživatelé mohou do tohoto pole přidat libovolnou anotaci a použít ji na Pod. Zde uvedené anotace přepíší výchozí anotace, jsou-li k dispozici. Vyžaduje Operator MQ 1.3.0 nebo novější.

Zobrazí se v:

- ".spec" na stránce 126

.spec.imagePullSecrets

LocalObjectReference obsahuje dostatek informací, aby bylo možné umístit odkazovaný objekt do stejného oboru názvů.

Zobrazí se v:

- ".spec" na stránce 126

Pole	Popis
`name` string	Název odkazujícího objektu. Další informace: https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names ÚKOL: Přidejte další užitečná pole. apiVersion, kind, uid?.

.spec.labels

Pole štítků slouží jako předávací pro štítky typu Pod. Uživatelé mohou do tohoto pole přidat libovolný štítek a použít jej na Pod. Zde uvedené štítky přepíší výchozí štítky, jsou-li k dispozici. Vyžaduje Operator MQ 1.3.0 nebo novější.

Zobrazí se v:

- ".spec" na stránce 126

.spec.license

Nastavení, která řídí převzetí licence a které metriky licencí se mají používat.

Zobrazí se v:

- ".spec" na stránce 126

Pole	Popis
`accept` boolean	Zda přijímáte licenci přidruženou k tomuto softwaru (povinné), či nikoli.
`license` string	Identifikátor licence, kterou přijímáte. Musí se jednat o správný identifikátor licence pro vámi používanou verzi produktu MQ. Platné hodnoty viz http://ibm.biz/BdqvCF.
`metric` string	Nastavení, které určuje, která metrika licence se má použít. Např. `ProcessorValueUnit`, `VirtualProcessorCore` nebo `ManagedVirtualServer`. Standardně se používá `ProcessorValueUnit` při použití licence MQ a `VirtualProcessorCore` při použití licence Cloud Pak for Integration.
`use` string	Nastavení, které řídí, jak se bude software používat, kde licence podporuje více použití. Platné hodnoty viz http://ibm.biz/BdqvCF.

.spec.pki

Nastavení Public Key Infrastructure pro definování klíčů a certifikátů pro použití se zabezpečením Transport Layer Security (TLS) nebo MQ Advanced Message Security (AMS).

Zobrazí se v:

- ".spec" na stránce 126

Pole	Popis
Pole `keys` PKISource	Soukromé klíče, které mají být přidány do úložiště klíčů správce front.
Pole `trust` PKISource	Certifikáty pro přidání do úložiště klíčů správce front.

.spec.pki.keys

PKISsource definuje zdroj informací o Public Key Infrastructure, jako např. klíče nebo certifikáty.

Zobrazí se v:

- ".spec.pki" na stránce 128

Pole	Popis
`name` string	Name se používá jako štítek pro klíč nebo certifikát. Musí se jednat o alfanumerický řetězec s malými písmeny.
`secret` Secret	Zadejte klíč pomocí tajného údaje Kybernetes.

.spec.pki.keys.secret

Zadejte klíč pomocí tajného údaje Kybernetes.

Zobrazí se v:

- ".spec.pki.keys" na stránce 128

Pole	Popis
Pole `items`	Klíče uvnitř tajného údaje Kubernetes, které mají být přidány do kontejneru správce front.
`secretName` string	Název tajného údaje Kubernetes.

.spec.pki.trust

PKISsource definuje zdroj informací o Public Key Infrastructure, jako např. klíče nebo certifikáty.

Zobrazí se v:

- ".spec.pki" na stránce 128

Pole	Popis
`name` string	Name se používá jako štítek pro klíč nebo certifikát. Musí se jednat o alfanumerický řetězec s malými písmeny.
`secret` Secret	Zadejte klíč pomocí tajného údaje Kybernetes.

.spec.pki.trust.secret

Zadejte klíč pomocí tajného údaje Kybernetes.

Zobrazí se v:

- ".spec.pki.trust" na stránce 129

Pole	Popis
Pole `items`	Klíče uvnitř tajného údaje Kubernetes, které mají být přidány do kontejneru správce front.
`secretName` string	Název tajného údaje Kubernetes.

.spec.queueManager

Nastavení pro kontejner správce front a základního správce front.

Zobrazí se v:

- ".spec" na stránce 126

Pole	Popis
`availability` Availability	Nastavení dostupnosti pro správce front, například zda má být použita dvojice aktivní-pohotovostní, či nikoli nebo nativní vysoká dostupnost.
`debug` boolean	Zda protokolovat zprávy ladění z kódu specifického pro kontejner do protokolu kontejneru, či nikoli. Výchozí hodnota je false.
`image` string	Kontejnerový obraz, který bude použit.
`imagePullPolicy` string	Nastavení, které se řídí, když se kubelet pokusí stáhnout uvedený obraz. Výchozí hodnota je `IfNotPresent`.
Pole `ini` INISource	Nastavení pro dodávání souborů INI pro správce front. Vyžaduje MQ Operator 1.1.0 nebo vyšší.
`livenessProbe` QueueManagerLivenessProbe	Nastavení, která řídí sondu živosti.

Pole	Popis
`logFormat` string	Který formát protokolu má být použit pro tento kontejner. Použijte `JSON` pro protokoly formátované JSON z kontejneru. Použijte `Basic` pro textově formátované zprávy. Výchozí hodnota je `Basic`.
`metrics` QueueManagerMetrics	Nastavení pro metriky ve stylu Prometheus.
Pole `mqsc` MQSCSource	Nastavení pro dodávání MQSC pro správce front. Vyžaduje MQ Operator 1.1.0 nebo vyšší.
`name` string	Název základního správce front MQ, pokud je odlišný od metadata.name. Toto pole použijte, pokud chcete, aby název správce front, který neodpovídá pravidlům Kubernetes, obsahoval názvy (například název, který obsahuje velká písmena).
`readinessProbe` QueueManagerReadinessProbe	Nastavení, které řídí sondu připravenosti.
`resources` Resources	Nastavení, která řídí požadavky na prostředky.
`route` Trasa	Nastavení pro trasu správce front. Vyžaduje Operator MQ 1.4.0 nebo novější.
`startupProbe` StartupProbe	Nastavení, která řídí spouštěcí sondu. Vztahuje se pouze na implementace MultiInstance a NativeHA. Vyžaduje Operator MQ 1.5.0 nebo novější.
`storage` QueueManagerStorage	Nastavení úložiště pro řízení použití trvalých svazků a úložných tříd správce front.

.spec.queueManager.availability

Nastavení dostupnosti pro správce front, například zda má být použita dvojice aktivní-pohotovostní, či nikoli nebo nativní vysoká dostupnost.

Zobrazí se v:

- “.spec.queueManager” na stránce 129

Pole	Popis
`tls` Tls	Volitelné nastavení TLS pro konfiguraci zabezpečené komunikace mezi replikami NativeHA. Vyžaduje Operator MQ 1.5.0 nebo novější.
`type` string	Typ dostupnosti, který se má použít. Použijte `SingleInstance` pro jeden Pod, který bude automaticky restartován (v některých případech) službou Kubernetes. Použijte `MultiInstance` pro dvojici podů, z nichž jeden je `aktivní` správce front a druhý z nich je pohotovostní. Použijte `NativeHA` pro replikaci nativní vysoké dostupnosti (vyžaduje MQ Operator 1.5.0 nebo vyšší). Výchozí hodnota je `SingleInstance`. Další podrobnosti viz http://ibm.biz/BdqAQa.
Řetezec `updateStrategy`	Strategie aktualizace, která má být použita pro správce front MultiInstance a NativeHA. Pomocí volby `RollingUpdate` můžete povolit automatické průběžné aktualizace, kdykoli se změní konfigurace správce front. Chcete-li zakázat automatické průběžné aktualizace, použijte volbu `OnDelete`. Změny správců front budou použity pouze v případě, že budou odstraněny pouze pody (včetně odstranění podů spouštěných externími faktory). Výchozí hodnota je `RollingUpdate`. Vyžaduje MQ Operator 1.6.0 nebo vyšší.

.spec.queueManager.availability.tls

Volitelné nastavení TLS pro konfiguraci zabezpečené komunikace mezi replikami NativeHA. Vyžaduje Operator MQ 1.5.0 nebo novější.

Zobrazí se v:

- ".spec.queueManager.availability" na stránce 130

Pole	Popis
`cipherSpec` string	Název specifikace CipherSpec pro zabezpečení NativeHA TLS.
`secretName` string	Název tajného údaje Kubernetes.

.spec.queueManager.ini

Zdroj konfiguračních souborů INI.

Zobrazí se v:

- ".spec.queueManager" na stránce 129

Pole	Popis
`configMap` ConfigMapINISource	ConfigMap reprezentuje Kubernetes ConfigMap obsahující informace INI.
`secret` SecretINISource	Secret reprezentuje tajný údaj Kubernetes obsahující informace INI.

.spec.queueManager.ini.configMap

ConfigMap reprezentuje Kubernetes ConfigMap obsahující informace INI.

Zobrazí se v:

- ".spec.queueManager.ini" na stránce 131

Pole	Popis
Pole `items`	Klíče uvnitř zdroje Kubernetes, které by měly být aplikovány.
`name` string	Název zdroje Kubernetes.

.spec.queueManager.ini.secret

Secret reprezentuje tajný údaj Kubernetes obsahující informace INI.

Zobrazí se v:

- ".spec.queueManager.ini" na stránce 131

Pole	Popis
Pole `items`	Klíče uvnitř zdroje Kubernetes, které by měly být aplikovány.
`name` string	Název zdroje Kubernetes.

.spec.queueManager.livenessProbe

Nastavení, která řídí sondu živosti.

Zobrazí se v:

- ".spec.queueManager" na stránce 129

Pole	Popis
`failureThreshold` integer	Minimální počet po sobě jdoucích selhání pro sondu, aby to bylo považováno po úspěšném dokončení za nezdar. Výchozí hodnota je 1.
`initialDelaySeconds` integer	Počet sekund po spuštění kontejneru, než je sonda zahájena. Výchozí hodnota je 90 sekund pro instanci SingleInstance. Výchozí hodnota je 0 sekund pro implementace MultiInstance a NativeHA. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.
`periodSeconds` integer	Jak často (v sekundách) provést sondu. Výchozí hodnota je 10 sekund.
`successThreshold` integer	Minimální počet po sobě jdoucích úspěšných dokončení pro sondu, aby to bylo považováno po nezdaru za úspěch. Výchozí hodnota je 1.
`timeoutSeconds` integer	Počet sekund, po jejichž uplynutí dojde k překročení časového limitu sondy. Výchozí hodnota je 5 sekund. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.

.spec.queueManager.metrics

Nastavení pro metriky ve stylu Prometheus.

Zobrazí se v:

- “.spec.queueManager” na stránce 129

Pole	Popis
`enabled` boolean	Zda se má povolit koncový bod metrik kompatibilních s Prometheus, či nikoli. Výchozí hodnota je true.

.spec.queueManager.mqsc

Zdroj konfiguračních souborů MQSC.

Zobrazí se v:

- “.spec.queueManager” na stránce 129

Pole	Popis
`configMap` ConfigMapMQSCSource	ConfigMap reprezentuje Kubernetes ConfigMap obsahující informace MQSC.
`secret` SecretMQSCSource	Secret reprezentuje tajný údaj Kubernetes obsahující informace MQSC.

.spec.queueManager.mqsc.configMap

ConfigMap reprezentuje Kubernetes ConfigMap obsahující informace MQSC.

Zobrazí se v:

- “.spec.queueManager.mqsc” na stránce 132

Pole	Popis
Pole `items`	Klíče uvnitř zdroje Kubernetes, které by měly být aplikovány.
`name` string	Název zdroje Kubernetes.

.spec.queueManager.mqsc.secret

Secret reprezentuje tajný údaj Kubernetes obsahující informace MQSC.

Zobrazí se v:

- ".spec.queueManager.mqsc" na stránce 132

Pole	Popis
Pole `items`	Klíče uvnitř zdroje Kubernetes, které by měly být aplikovány.
`name` string	Název zdroje Kubernetes.

.spec.queueManager.readinessProbe

Nastavení, které řídí sondu připravenosti.

Zobrazí se v:

- ".spec.queueManager" na stránce 129

Pole	Popis
`failureThreshold` integer	Minimální počet po sobě jdoucích selhání pro sondu, aby to bylo považováno po úspěšném dokončení za nezdar. Výchozí hodnota je 1.
`initialDelaySeconds` integer	Počet sekund po spuštění kontejneru, než je sonda zahájena. Výchozí hodnota je 10 sekund pro instanci SingleInstance. Výchozí hodnota je 0 sekund pro implementace MultiInstance a NativeHA. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.
`periodSeconds` integer	Jak často (v sekundách) provést sondu. Výchozí hodnota je 5 sekund.
`successThreshold` integer	Minimální počet po sobě jdoucích úspěšných dokončení pro sondu, aby to bylo považováno po nezdaru za úspěch. Výchozí hodnota je 1.
`timeoutSeconds` integer	Počet sekund, po jejichž uplynutí dojde k překročení časového limitu sondy. Výchozí hodnota je 3 sekundy. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.

.spec.queueManager.resources

Nastavení, která řídí požadavky na prostředky.

Zobrazí se v:

- ".spec.queueManager" na stránce 129

Pole	Popis
`limits` Limits	Nastavení CPU & paměti.
`requests` Requests	Nastavení CPU & paměti.

.spec.queueManager.resources.limits

Nastavení CPU & paměti.

Zobrazí se v:

- ".spec.queueManager.resources" na stránce 133

Pole	Popis
cpu	
memory	

.spec.queueManager.resources.requests

Nastavení CPU & paměti.

Zobrazí se v:

- ".spec.queueManager.resources" na stránce 133

Pole	Popis
cpu	
memory	

.spec.queueManager.route

Nastavení pro trasu správce front. Vyžaduje Operator MQ 1.4.0 nebo novější.

Zobrazí se v:

- ".spec.queueManager" na stránce 129

Pole	Popis
enabled boolean	Zda povolit nebo zakázat trasu. Výchozí hodnota je true.

.spec.queueManager.startupProbe

Nastavení, která řídí spouštěcí sondu. Vztahuje se pouze na implementace MultiInstance a NativeHA. Vyžaduje Operator MQ 1.5.0 nebo novější.

Zobrazí se v:

- ".spec.queueManager" na stránce 129

Pole	Popis
failureThreshold integer	Minimální počet po sobě jdoucích selhání pro sondu, aby to bylo považováno za nezdar. Výchozí hodnota je 60.
initialDelaySeconds integer	Počet sekund po spuštění kontejneru, než je sonda zahájena. Výchozí hodnota je 0 sekund. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.
periodSeconds integer	Jak často (v sekundách) provést sondu. Výchozí hodnota je 5 sekund.
successThreshold integer	Minimální počet po sobě jdoucích úspěšných dokončení pro sondu, aby to bylo považováno za úspěch. Výchozí hodnota je 1.
timeoutSeconds integer	Počet sekund, po jejichž uplynutí dojde k překročení časového limitu sondy. Výchozí hodnota je 5 sekund. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.

.spec.queueManager.storage

Nastavení úložiště pro řízení použití trvalých svazků a úložných tříd správce front.

Zobrazí se v:

- “.spec.queueManager” na stránce 129

Pole	Popis
Řetězec `defaultClass`	Paměťová třída, která má být standardně použita pro všechny trvalé svazky tohoto správce front. Specifické trvalé svazky mohou definovat vlastní paměťovou třídu, která přepíše toto výchozí nastavení paměťové třídy. Je-li `type of availability` `SingleInstance` nebo `NativeHA`, může být paměťová třída typu type ReadWriteOnce nebo ReadWriteMany. Je-li `type of availability` `MultiInstance`, musí být paměťová třída typu ReadWriteMany.
`defaultDeleteClaim` boolean	Určuje, zda mají být odstraněny všechny svazky při odstranění správce front. Specifické trvalé svazky mohou definovat svou vlastní hodnotu pro deleteClaim, což přepíše toto nastavení defaultDeleteClaim. Výchozí hodnota je false.
`persistedData` QueueManagerOptionalVolume	Podrobnosti o PersistentVolume pro trvalá data pro produkt MQ, včetně konfigurace, front a zpráv. Povinné při použití správce front s více instancemi.
`queueManager` QueueManagerVolume	Výchozí svazek PersistentVolume pro veškerá data běžně pod `/var/mqm`. Bude obsahovat všechna trvalá data a protokoly zotavení, pokud nejsou určeny žádné jiné svazky.
`recoveryLogs` QueueManagerOptionalVolume	Podrobnosti o trvalých svazcích pro protokoly zotavení MQ. Povinné při použití správce front s více instancemi.

.spec.queueManager.storage.persistedData

Podrobnosti o PersistentVolume pro trvalá data pro produkt MQ, včetně konfigurace, front a zpráv. Povinné při použití správce front s více instancemi.

Zobrazí se v:

- “.spec.queueManager.storage” na stránce 134

Pole	Popis
`class` string	Paměťová třída, která se má použít pro tento svazek. Platné pouze v případě, že `type` je `persistent-claim`. Je-li `type of availability` `SingleInstance` nebo `NativeHA`, může být paměťová třída typu type ReadWriteOnce nebo ReadWriteMany. Je-li `type of availability` `MultiInstance`, musí být paměťová třída typu ReadWriteMany.
`deleteClaim` boolean	Určuje, zda má být tento svazek odstraněn při odstranění správce front.
`enabled` boolean	Určuje, zda má být tento svazek povolen jako samostatný svazek, nebo umístěn ve výchozím svazku `queueManager`. Výchozí hodnota je false.
`size` string	Velikost svazku PersistentVolume, který má být předán objektu Kubernetes, včetně jednotek SI.Size of the PersistentVolume to pass to Kubernetes, including SI units Platné pouze v případě, že `type` je `persistent-claim`. Například `2Gi`. Výchozí hodnota je `2Gi`.
`sizeLimit` string	Limit velikosti při použití svazku `ephemeral`. Soubory jsou stále zapisovány do dočasného adresáře, takže můžete tuto volbu použít k omezení velikosti. Platné pouze v případě, že `type` je `ephemeral`.
`type` string	Typ svazku, který se má použít. Chcete-li použít dočasné úložiště, vyberte `ephemeral`, chcete-li použít trvalý svazek, vyberte `ephemeral`. Výchozí hodnota je `persistent-claim`.

.spec.queueManager.storage.queueManager

Výchozí svazek PersistentVolume pro veškerá data běžně pod `/var/mqm`. Bude obsahovat všechna trvalá data a protokoly zotavení, pokud nejsou určeny žádné jiné svazky.

Zobrazí se v:

- ".spec.queueManager.storage" na stránce 134

Pole	Popis
`class` string	Paměťová třída, která se má použít pro tento svazek. Platné pouze v případě, že `type` je `persistent-claim`. Je-li `type of availability` `SingleInstance` nebo `NativeHA`, může být paměťová třída typu type ReadWriteOnce nebo ReadWriteMany. Je-li `type of availability` `MultiInstance`, musí být paměťová třída typu ReadWriteMany.
`deleteClaim` boolean	Určuje, zda má být tento svazek odstraněn při odstranění správce front.
`size` string	Velikost svazku PersistentVolume, který má být předán objektu Kubernetes, včetně jednotek SI.Size of the PersistentVolume to pass to Kubernetes, including SI units Platné pouze v případě, že `type` je `persistent-claim`. Například `2Gi`. Výchozí hodnota je `2Gi`.
`sizeLimit` string	Limit velikosti při použití svazku `ephemeral`. Soubory jsou stále zapisovány do dočasného adresáře, takže můžete tuto volbu použít k omezení velikosti. Platné pouze v případě, že `type` je `ephemeral`.
`type` string	Typ svazku, který se má použít. Chcete-li použít dočasné úložiště, vyberte `ephemeral`, chcete-li použít trvalý svazek, vyberte `ephemeral`. Výchozí hodnota je `persistent-claim`.

.spec.queueManager.storage.recoveryLogs

Podrobnosti o trvalých svazcích pro protokoly zotavení MQ. Povinné při použití správce front s více instancemi.

Zobrazí se v:

- ".spec.queueManager.storage" na stránce 134

Pole	Popis
`class` string	Paměťová třída, která se má použít pro tento svazek. Platné pouze v případě, že `type` je `persistent-claim`. Je-li `type of availability` `SingleInstance` nebo `NativeHA`, může být paměťová třída typu type ReadWriteOnce nebo ReadWriteMany. Je-li `type of availability` `MultiInstance`, musí být paměťová třída typu ReadWriteMany.
`deleteClaim` boolean	Určuje, zda má být tento svazek odstraněn při odstranění správce front.
`enabled` boolean	Určuje, zda má být tento svazek povolen jako samostatný svazek, nebo umístěn ve výchozím svazku `queueManager`. Výchozí hodnota je false.
`size` string	Velikost svazku PersistentVolume, který má být předán objektu Kubernetes, včetně jednotek SI.Size of the PersistentVolume to pass to Kubernetes, including SI units Platné pouze v případě, že `type` je `persistent-claim`. Například `2Gi`. Výchozí hodnota je `2Gi`.
`sizeLimit` string	Limit velikosti při použití svazku `ephemeral`. Soubory jsou stále zapisovány do dočasného adresáře, takže můžete tuto volbu použít k omezení velikosti. Platné pouze v případě, že `type` je `ephemeral`.

Pole	Popis
`type` string	Typ svazku, který se má použít. Chcete-li použít dočasné úložiště, vyberte `ephemeral`, chcete-li použít trvalý svazek, vyberte `ephemeral`. Výchozí hodnota je `persistent-claim`.

.spec.securityContext

Nastavení zabezpečení, které má být přidáno do kontextu securityContext podu správce front.

Zobrazí se v:

- ".spec" na stránce 126

Pole	Popis
`fsGroup` integer	Speciální doplňková skupina, která se vztahuje na všechny kontejnery v podu. Některé typy svazků umožňují Kubelet změnit vlastnictví tohoto svazku, které má být vlastněno tímto podem: 1. Vlastnící GID bude skupina FSGroup 2. Bit setgid je nastaven (nové soubory vytvořené ve svazku budou vlastněny skupinou FSGroup) 3. Bity oprávnění jsou OR d with rw-rw---- Pokud nejsou nastaveny, Kubelet neupraví vlastnictví a oprávnění žádné svazku.
`initVolumeAsRoot` boolean	To ovlivňuje securityContext použitý kontejnerem, který inicializuje PersistentVolume. Nastavte tuto hodnotu na `true`, jestliže používáte poskytovatele úložiště, který vyžaduje, abyste byli kořenovým uživatelem pro přístup k nově zajišťovaným svazkům. Nastavení této hodnoty na `true` ovlivňuje, který objekt SCC (Security Context Constraints) můžete použít, a správce front se nemusí spustit, pokud nemáte autorizaci k použití objektu SCC, který umožňuje kořenovému uživateli. Výchozí hodnota je false. Další informace viz https://docs.openshift.com/container-platform/latest/authentication/managing-security-context-constraints.html.
Pole `supplementalGroups`	Seznam skupin aplikovaných na první proces spuštěný v každém kontejneru kromě primárního GID kontejneru. Není-li zadán, nebudou žádné skupiny přidány do žádného kontejneru.

.spec.template

Rozšířené vytváření šablon pro prostředky Kubernetes. Šablona umožňuje uživatelům potlačit způsob, jakým produkt IBM MQ generuje základní prostředky typu Kubernetes, jako např. StatefulSet, Pods a Services. Tento postup je určen pouze pro pokročilé uživatele, protože má potenciál narušit normální provoz produktu MQ, pokud je použit nesprávně. Všechny hodnoty zadané kdekoli jinde v prostředku správce front budou přepsány nastaveními v šabloně.

Zobrazí se v:

- ".spec" na stránce 126

Pole	Popis
pod	Potlačení pro šablonu použitou pro Pod. Viz https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.17/#podspec-v1-core.

.spec.tracing

Nastavení pro integraci trasování s produktem Cloud Pak for Integration Operations Dashboard.

Zobrazí se v:

- ".spec" na stránce 126

Pole	Popis
`agent` TracingAgent	Pouze v produktu Cloud Pak for Integration můžete konfigurovat nastavení pro volitelného agenta trasování.
`collector` TracingCollector	Pouze v produktu Cloud Pak for Integration můžete konfigurovat nastavení pro volitelný kolektor trasování.
`enabled` boolean	Zda se má povolit integrace s produktem Cloud Pak for Integration Operations Dashboard přes trasování, či nikoli. Výchozí hodnota je false.
`namespace` string	Obor názvů, kde je nainstalován produkt Cloud Pak for Integration Operations Dashboard.

.spec.tracing.agent

Pouze v produktu Cloud Pak for Integration můžete konfigurovat nastavení pro volitelného agenta trasování.

Zobrazí se v:

- ".spec.tracing" na stránce 137

Pole	Popis
`image` string	Kontejnerový obraz, který bude použit.
`imagePullPolicy` string	Nastavení, které se řídí, když se kubelet pokusí stáhnout uvedený obraz. Výchozí hodnota je `IfNotPresent`.
`livenessProbe` TracingProbe	Nastavení, která řídí sondu živosti.
`readinessProbe` TracingProbe	Nastavení, které řídí sondu připravenosti.

.spec.tracing.agent.livenessProbe

Nastavení, která řídí sondu živosti.

Zobrazí se v:

- ".spec.tracing.agent" na stránce 138

Pole	Popis
`failureThreshold` integer	Minimální počet po sobě jdoucích selhání pro sondu, aby to bylo považováno po úspěšném dokončení za nezdar. Výchozí hodnota je 1.
`initialDelaySeconds` integer	Počet sekund po spuštění kontejneru, než budou zahájeny sondy živosti. Výchozí hodnota je 10 sekund. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.
`periodSeconds` integer	Jak často (v sekundách) provést sondu. Výchozí hodnota je 10 sekund.
`successThreshold` integer	Minimální počet po sobě jdoucích úspěšných dokončení pro sondu, aby to bylo považováno po nezdaru za úspěch. Výchozí hodnota je 1.
`timeoutSeconds` integer	Počet sekund, po jejichž uplynutí dojde k překročení časového limitu sondy. Výchozí hodnota je 2 sekundy. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.

.spec.tracing.agent.readinessProbe

Nastavení, které řídí sondu připravenosti.

Zobrazí se v:

- “.spec.tracing.agent” na stránce 138

Pole	Popis
`failureThreshold` integer	Minimální počet po sobě jdoucích selhání pro sondu, aby to bylo považováno po úspěšném dokončení za nezdar. Výchozí hodnota je 1.
`initialDelaySeconds` integer	Počet sekund po spuštění kontejneru, než budou zahájeny sondy živosti. Výchozí hodnota je 10 sekund. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.
`periodSeconds` integer	Jak často (v sekundách) provést sondu. Výchozí hodnota je 10 sekund.
`successThreshold` integer	Minimální počet po sobě jdoucích úspěšných dokončení pro sondu, aby to bylo považováno po nezdaru za úspěch. Výchozí hodnota je 1.
`timeoutSeconds` integer	Počet sekund, po jejichž uplynutí dojde k překročení časového limitu sondy. Výchozí hodnota je 2 sekundy. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.

.spec.tracing.collector

Pouze v produktu Cloud Pak for Integration můžete konfigurovat nastavení pro volitelný kolektor trasování.

Zobrazí se v:

- “.spec.tracing” na stránce 137

Pole	Popis
`image` string	Kontejnerový obraz, který bude použit.
`imagePullPolicy` string	Nastavení, které se řídí, když se kubelet pokusí stáhnout uvedený obraz. Výchozí hodnota je `IfNotPresent`.
`livenessProbe` TracingProbe	Nastavení, která řídí sondu živosti.
`readinessProbe` TracingProbe	Nastavení, které řídí sondu připravenosti.

.spec.tracing.collector.livenessProbe

Nastavení, která řídí sondu živosti.

Zobrazí se v:

- “.spec.tracing.collector” na stránce 139

Pole	Popis
`failureThreshold` integer	Minimální počet po sobě jdoucích selhání pro sondu, aby to bylo považováno po úspěšném dokončení za nezdar. Výchozí hodnota je 1.
`initialDelaySeconds` integer	Počet sekund po spuštění kontejneru, než budou zahájeny sondy živosti. Výchozí hodnota je 10 sekund. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.
`periodSeconds` integer	Jak často (v sekundách) provést sondu. Výchozí hodnota je 10 sekund.

Pole	Popis
`successThreshold` integer	Minimální počet po sobě jdoucích úspěšných dokončení pro sondu, aby to bylo považováno po nezdaru za úspěch. Výchozí hodnota je 1.
`timeoutSeconds` integer	Počet sekund, po jejichž uplynutí dojde k překročení časového limitu sondy. Výchozí hodnota je 2 sekundy. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.

.spec.tracing.collector.readinessProbe

Nastavení, které řídí sondu připravenosti.

Zobrazí se v:

- ".spec.tracing.collector" na stránce 139

Pole	Popis
`failureThreshold` integer	Minimální počet po sobě jdoucích selhání pro sondu, aby to bylo považováno po úspěšném dokončení za nezdar. Výchozí hodnota je 1.
`initialDelaySeconds` integer	Počet sekund po spuštění kontejneru, než budou zahájeny sondy živosti. Výchozí hodnota je 10 sekund. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.
`periodSeconds` integer	Jak často (v sekundách) provést sondu. Výchozí hodnota je 10 sekund.
`successThreshold` integer	Minimální počet po sobě jdoucích úspěšných dokončení pro sondu, aby to bylo považováno po nezdaru za úspěch. Výchozí hodnota je 1.
`timeoutSeconds` integer	Počet sekund, po jejichž uplynutí dojde k překročení časového limitu sondy. Výchozí hodnota je 2 sekundy. Další informace: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-probes.

.spec.web

Nastavení pro webový server MQ.

Zobrazí se v:

- ".spec" na stránce 126

Pole	Popis
`enabled` boolean	Zda povolit nebo zakázat webový server. Výchozí hodnota je false.

.status

Pozorovaný stav správce front.

Zobrazí se v:

- "Správce front" na stránce 125

Pole	Popis
`adminUiUrl` string	Adresa URL pro uživatelské rozhraní administrace.
`availability` Availability	Stav dostupnosti správce front.
Pole `conditions` QueueManagerStatusCondition	Podmínky reprezentují nejnovější dostupná pozorování stavu správce front.

Pole	Popis
Pole `endpoints` QueueManagerStatusEndpoint	Informace v koncových bodech, které tento správce front vystavuje, jako např. koncové body rozhraní API nebo uživatelské rozhraní.
`name` string	Název správce front.
`phase` string	Fáze stavu správce front.
`versions` QueueManagerStatusVersion	Verze používaného produktu MQ a další verze dostupné z produktu IBM Entitled Registry.

.status.availability

Stav dostupnosti správce front.

Zobrazí se v:

- “.status” na stránce 140

Pole	Popis
`initialQuorumEstablished` boolean	Určuje, zda bylo počáteční kvorum ustanoveno pro NativeHA.

.status.conditions

QueueManagerStatusCondition definuje podmínky správce front.

Zobrazí se v:

- “.status” na stránce 140

Pole	Popis
`lastTransitionTime` string	Poslední čas, kdy podmínka přešla z jednoho stavu do druhého.
`message` string	Zpráva čitelná pro člověka označující podrobnosti o posledním přechodu.
`reason` string	Důvod posledního přechodu tohoto stavu.
`status` string	Stav podmínky.
`type` string	Typ podmínky.

.status.endpoints

QueueManagerStatusEndpoint definuje koncové body správce front.

Zobrazí se v:

- “.status” na stránce 140

Pole	Popis
`name` string	Název koncového bodu.
`type` string	Typ koncového bodu, například „UI“ pro koncový bod uživatelského rozhraní, „API“ pro koncový bod rozhraní API, „OpenAPI“ pro dokumentaci rozhraní API.
`uri` string	Identifikátor URI pro koncový bod.

.status.versions

Verze používaného produktu MQ a další verze dostupné z produktu IBM Entitled Registry.

Zobrazí se v:

- ".status" na stránce 140

Pole	Popis
`available` QueueManagerStatusVersionAvailable	Další verze produktu MQ dostupné z produktu IBM Entitled Registry.
`reconciled` string	Používá se specifická verze produktu IBM MQ. Je-li zadán vlastní obraz, pak se nemusí shodovat s aktuálně používanou verzí produktu MQ.

.status.versions.available

Další verze produktu MQ dostupné z produktu IBM Entitled Registry.

Zobrazí se v:

- ".status.versions" na stránce 142

Pole	Popis
Pole `channels`	Kanály, které jsou k dispozici pro automatickou aktualizaci verze MQ.
Pole `versions` Versions	Specifické verze produktu MQ, které jsou k dispozici.

.status.versions.available.versions

QueueManagerStatusVersion definuje verzi produktu MQ.

Zobrazí se v:

- ".status.versions.available" na stránce 142

Pole	Popis
`name` string	`Název` verze pro verzi správce front. Toto jsou platné hodnoty pro pole `spec.version`.

OpenShift CP4I *Stavové podmínky pro správce front (mq.ibm.com/v1beta1)*

Pole **`status.conditions`** se aktualizují tak, aby odrážely podmínku prostředku `QueueManager`. Obecně podmínky popisují nestandardní situace. Správce front ve zdravém, připraveném stavu nemá žádné podmínky **`Error`** (chyba) nebo **`Pending`** (nevyřízené). Může mít nějaké doporučovací podmínky typu **`Warning`** (varování).

Podpora podmínek byla zavedena v produktu IBM MQ Operator 1.2.

Pro prostředek `QueueManager` jsou definovány tyto podmínky:

Tabulka 1. Stavové podmínky správce front			
Komponenta	**Typ podmínky**	**Kód příčiny**	**Varovná zpráva**
QueueManager[7]	Nevyřízeno	Creating	Probíhá implementace správce front MQ
	Nevyřízeno	OidcPending	Správce front MQ čeká na registraci klienta OIDC
	Chyba	Nezdar	Implementace správce front MQ se nezdařila
	Varování	UnsupportedVersion	[8]Operand byl instalován operátorem, který není verzí *<ocp_version>* podporován. Tento operand není podporován.
	Varování	EUSSupport	[9]Operand EUS *<mq_version>* byl nainstalován, je ale spravován operátorem, který nelze kvalifikovat pro dobu rozšířené podpory. Tento operand nelze kvalifikovat pro dobu rozšířené podpory.
	Varování	EUSSupport	[10]Byl nainstalován operand EUS*<mq_version>*, ale OCP verze 4*<ocp_version>* nekvalifikuje pro dobu rozšířené podpory. Tento operand nelze kvalifikovat pro dobu rozšířené podpory.
	Varování	EUSSupport	[11]Byl nainstalován operand EUS *<mq_version>*, ale OCP verze *<ocp_version>* nekvalifikuje pro dobu rozšířené podpory. Tento operand je podporován podle pravidelného vydání CD.

[7] Podmínky `Creating` a `Failed` monitorují celkový průběh implementace správce front. Pokud je povoleno použití licence IBM Cloud Pak for Integration webové konzoly MQ, potom podmínka `OidcPending` protokoluje stav správce front při čekání na dokončení registrace klienta OIDC s IAM.
[8] Operator 1.4.0 a novější
[9] Operator 1.4.0 a novější
[10] Operator 1.4.0 a novější
[11] Pouze Operator 1.3.0

Tabulka 1. Stavové podmínky správce front (pokračování)			
Komponenta	**Typ podmínky**	**Kód příčiny**	**Varovná zpráva**
Pod[12]	Nevyřízeno	PodPending	Probíhá implementace Pod pro správce front MQ
	Chyba	PodFailed	Probíhá implementace Pod pro správce front MQ
Úložný prostor[13]	Nevyřízeno	StoragePending	Probíhá zajišťování úložiště pro správce front MQ
	Varování	StorageEphemeral	Použití dočasného úložiště pro produkčního správce front MQ
	Chyba	StorageFailed	Úložiště pro správce front MQ se nezdařilo zajistit

Multi Sestavení vlastního kontejneru IBM MQ a kódu implementace

Vyviňte svůj vlastní kontejner. Jedná se o nejflexibilnější řešení kontejneru, které ale od vás vyžaduje značné dovednosti v konfiguraci kontejnerů a abyste "vlastnili" výsledný kontejner.

Než začnete

Než začnete vyvíjet svůj vlastní kontejner, zvažte, zda nemůžete místo toho využít některý z předpřipravených zabalených kontejnerů od IBM. Viz IBM MQ v kontejnerech

Informace o této úloze

Když zabalíte IBM MQ jako kontejnerový obraz, můžete rychle a snadno implementovat změny v aplikaci tak, aby mohly být implementovány v testovacím a přechodovém systému. To může být významným přínosem pro průběžné doručování ve vašem podniku.

Procedura

- “Plánování vlastního obrazu správce front IBM MQ pomocí kontejneru” na stránce 145
- “Sestavení ukázkového kontejnerového obrazu správce front produktu IBM MQ” na stránce 145
- “Spuštění lokálních aplikací vazby v samostatných kontejnerech” na stránce 148

Související pojmy

IBM MQ v kontejnerech

[12] Podmínky Pod monitorují stav podů během implementace správce front. Když se zobrazí podmínka `PodFailed`, pak bude celková podmínka správce front rovněž nastavena na `Failed`.

[13] Podmínky úložiště monitorují průběh (podmínka `StoragePending`) požadavků na vytvoření svazků pro trvalé úložiště a hlásí zpět chyby vazby a další selhání. Jestliže dojde k jakékoli chybě během zajišťování úložiště, bude podmínka `StorageFailed` přidána do seznamu podmínek a celková podmínka správce front bude rovněž nastavena na `Failed`.

Multi Plánování vlastního obrazu správce front IBM MQ pomocí kontejneru

Při spuštění správce front produktu IBM MQ v kontejneru je třeba vzít v úvahu několik požadavků. Ukázkový kontejnerový obraz nabízí způsob, jak tyto požadavky zpracovat, ale chcete-li použít vlastní obraz, je třeba zvážit, jak jsou tyto požadavky zpracovávány.

Řízení procesu

Když spustíte kontejner, v podstatě spouštíte jeden proces (PID 1 uvnitř kontejneru), který může později vyvolat podřízené procesy.

Pokud hlavní proces skončí, běhové prostředí kontejneru zastaví kontejner. Správce front produktu IBM MQ vyžaduje, aby bylo na pozadí spuštěno více procesů.

Z tohoto důvodu se musíte ujistit, že váš hlavní proces zůstane aktivní, dokud bude spuštěn správce front. Dobrým zvykem je kontrolovat z tohoto procesu, zda je správce front aktivní, například prostřednictvím administrativních dotazů.

Naplnění `/var/mqm`

Kontejnery musí být nakonfigurovány s `/var/mqm` jako svazkem.

Provedete-li to, bude adresář svazku při prvním spuštění kontejneru prázdný. Tento adresář je obvykle naplněn v době instalace, ale instalace a běhové prostředí jsou oddělená prostředí při použití kontejneru.

Chcete-li tento problém vyřešit při spuštění kontejneru, můžete použít příkaz **crtmqdir** k naplnění `/var/mqm` při prvním spuštění.

Zabezpečení kontejneru

Aby byly minimalizovány nároky na zabezpečení běhového prostředí, jsou ukázkové kontejnerové obrazy nainstalovány s použitím rozbalitelné instalace produktu IBM MQ. Tím je zajištěno, že nejsou nastaveny žádné bity `setuid` a že kontejner nemusí ani používat eskalaci oprávnění. Některé systémy kontejnerů definují, která ID uživatelů se mohou používat. Rozbalitelná instalace nečiní žádné předpoklady o dostupných uživatelích operačního systému.

Multi Sestavení ukázkového kontejnerového obrazu správce front produktu IBM MQ

Tyto informace použijte k sestavení ukázkového kontejnerového obrazu pro spuštění správce front IBM MQ v kontejneru.

Informace o této úloze

Za prvé sestavíte základní obraz obsahující systém souborů Red Hat Universal Base Image a čistou instalaci produktu IBM MQ.

Za druhé sestavíte nad základní další vrstvu kontejnerového obrazu, která přidává nějakou konfiguraci produktu IBM MQ, aby bylo umožněno základní zabezpečení ID uživatele a hesla.

Nakonec spustíte kontejner tak, aby používal tento obraz jako svůj systém souborů, s obsahem `/var/mqm` poskytovaným svazkem kontejneru na systému souborů hostitele.

Procedura

- Informace, jak sestavit ukázkový kontejnerový obraz pro spuštění správce front IBM MQ v kontejneru viz následující dílčí témata:
  - “Sestavení ukázkového obrazu základního správce front produktu IBM MQ” na stránce 146

– “Sestavení ukázkového obrazu nakonfigurovaného správce front IBM MQ” na stránce 146

Multi Sestavení ukázkového obrazu základního správce front produktu IBM MQ

Abyste mohli používat produkt IBM MQ ve svém vlastním kontejnerovém obrazu, musíte nejprve sestavit základní obraz s čistou instalací produktu IBM MQ. Následující postup ukazuje, jak sestavit ukázkový základní obraz pomocí ukázkového kódu hostovaného na serveru GitHub.

Procedura

- Použijte soubory make dodané v úložišti mq-container GitHub k sestavení produkčního kontejnerového obrazu.

  Postupujte podle pokynů v části Sestavení kontejnerového obrazu v GitHub. Pokud plánujete konfigurovat zabezpečený přístup pomocí Red Hat OpenShift Container Platform "restricted" Security Context Constraint (SCC), musíte použít balík 'No-Install' IBM MQ .

Výsledky

Nyní máte nainstalovaný základní kontejnerový obraz s nainstalovaným produktem IBM MQ.

Nyní jste připraveni sestavit ukázkový nakonfigurovaný obraz správce front IBM MQ.

Multi Sestavení ukázkového obrazu nakonfigurovaného správce front IBM MQ

Jakmile sestavíte generický kontejnerový obraz základního produktu IBM MQ, musíte použít vlastní konfiguraci, abyste umožnili bezpečný přístup. Chcete-li tak učinit, vytvořte vlastní vrstvu kontejnerového obrazu s použitím generického obrazu jako nadřízeného prvku.

Než začnete

V 9.2.0 Tato úloha předpokládá, že při sestavení ukázkového základního obrazu správce front IBM MQ jste použili balík "No-Install" IBM MQ. Jinak nemůžete konfigurovat zabezpečený přístup pomocí Red Hat OpenShift Container Platform "omezeného" objektu Security Context Constraint (SCC). "Omezený" objekt SCC, který se používá ve výchozím nastavení, používá náhodná ID uživatelů a zabraňuje eskalaci oprávnění změnou na jiného uživatele. Tradiční instalační program produktu IBM MQ založený na balících RPM se spoléhá na uživatele a skupinu `mqm` a také používá bity `setuid` na spustitelných programech. Když v produktu IBM MQ 9.2 použijete balík "No-Install" IBM MQ, není k dispozici žádný uživatel `mqm` ani skupina `mqm`.

Postup

1. Vytvořte nový adresář a přidejte soubor s názvem `config.mqsc` s následujícím obsahem:

```
DEFINE QLOCAL(EXAMPLE.QUEUE.1) REPLACE
```

   Mějte na zřeteli, že předchozí příklad používá jednoduché ověření ID uživatele a hesla. Nicméně můžete použít jakoukoli konfiguraci zabezpečení, kterou vyžaduje váš podnik.

2. Vytvořte soubor s názvem `Dockerfile` s následujícím obsahem:

```
FROM mq
COPY config.mqsc /etc/mqm/
```

3. Sestavte vlastní kontejnerový obraz pomocí následujícího příkazu:

```
docker build -t mymq .
```

kde „`.`“ je adresář obsahující dva soubory, které jste právě vytvořili.

Docker potom vytvoří dočasný kontejner pomocí tohoto obrazu a spustí zbývající příkazy.

Poznámka: V systému Red Hat Enterprise Linux (RHEL) můžete použít příkaz **`docker`** (RHEL V7) nebo **podman** (RHEL V7 nebo RHEL V8). V systému Linux bude nutné spustit příkazy **`docker`** pomocí příkazu **sudo** na začátku příkazu, abyste získali dodatečná oprávnění.

4. Spusťte nový upravený obraz a vytvořte nový kontejner s obrazem disku, který jste právě vytvořili.

   Vaše nová vrstva obrazu neurčovala žádný konkrétní příkaz ke spuštění, takže byl zděděn z nadřízeného obrazu. Vstupní bod nadřízeného prvku (kód je k dispozici v GitHub):

   - Vytvoří správce front.
   - Spustí správce front.
   - Vytvoří výchozí modul listener.
   - Poté spustí všechny příkazy MQSC z `/etc/mqm/config.mqsc`.

   Chcete-li spustit nový upravený obraz, zadejte následující příkazy:

```
docker run \
  --env LICENSE=accept \
  --env MQ_QMGR_NAME=QM1 \
  --volume /var/example:/var/mqm \
  --publish 1414:1414 \
  --detach \
  mymq
```

   Kde:

   První parametr env
   Předává proměnnou prostředí do kontejneru, který potvrzuje vaše přijetí licence pro IBM IBM WebSphere MQ. Můžete také nastavit proměnnou LICENSE pro zobrazení licence.

   Další podrobnosti viz informace o licenci IBM MQ v licencích IBM MQ.

   Druhý parametr env
   Nastaví název správce front, který používáte.

   Parametr svazku
   Říká kontejneru, že jakékoli zápisy MQ do `/var/mqm` by měly být skutečně zapsány do `/var/example` na hostiteli.

   Tato volba znamená, že lze kontejner snadno odstranit později a přesto zachovat veškerá trvalá data. Tato volba také usnadňuje zobrazení souborů protokolu.

   Parametr publikování
   Mapuje porty na hostitelském systému do portů v kontejneru. Kontejner se standardně spouští s vlastní interní adresou IP, což znamená, že musíte specificky mapovat všechny porty, které chcete vystavit.

   V tomto příkladu to znamená mapování portu 1414 na hostiteli na port 1414 v kontejneru.

   Parametr odpojení
   Spustí kontejner na pozadí.

Výsledky

Sestavili jste nakonfigurovaný kontejnerový obraz a můžete jej zobrazit pomocí příkazu **`docker ps`**. Procesy produktu IBM MQ spuštěné ve vašem kontejneru si můžete zobrazit pomocí příkazu **`docker top`**.

Upozornění:

Protokoly kontejneru si můžete zobrazit pomocí příkazu **`docker logs ${CONTAINER_ID}`**.

Jak pokračovat dále

- Pokud se kontejner nezobrazí, když použijete příkaz **docker ps**, mohlo dojít k nezdaru kontejneru. Kontejnery se selháním se zobrazují pomocí příkazu **docker ps -a** .
- Použijete-li příkaz **docker ps -a**, zobrazí se ID kontejneru. Toto ID bylo také vytištěno, jste zadali příkaz **docker run**.
- Protokoly kontejneru si můžete zobrazit pomocí příkazu **docker logs ${CONTAINER_ID}**.

Multi Spuštění lokálních aplikací vazby v samostatných kontejnerech

Díky sdílení oboru názvů procesu mezi kontejnery v Docker můžete spouštět aplikace, které vyžadují připojení lokální vazby k produktu IBM MQ v samostatných kontejnerech ze správce front IBM MQ.

Informace o této úloze

Tato funkce je podporována ve správcích front IBM MQ 9.0.3 a novějších.

Musíte dodržovat následující omezení:

- Musíte sdílet obor názvů PID kontejnerů pomocí argumentu `--pid`.
- Musíte sdílet obor názvů IPC kontejnerů pomocí argumentu `--ipc`.
- Musíte buď:
  1. Sdílet obor názvů UTS kontejnerů s hostitelem pomocí argumentu `--uts`, nebo
  2. zajistit, že kontejnery budou mít stejný název hostitele pomocí argumentu `-h` nebo `--hostname`.
- Datový adresář IBM MQ je třeba připojit do svazku, který je k dispozici pro všechny kontejnery v adresáři `/var/mqm`

Tuto funkčnost můžete vyzkoušet provedením následujících kroků v systému Linux, na kterém je již nainstalován Docker.

Následující příklad používá ukázkový kontejnerový obraz IBM MQ. Podrobnosti o tomto obrazu viz Github.

Postup

1. Vytvořte dočasný adresář, který bude fungovat jako váš svazek, zadáním následujícího příkazu:

```
mkdir /tmp/dockerVolume
```

2. Vytvořte správce front (`QM1`) v kontejneru, s názvem `sharedNamespace`, zadáním následujícího příkazu:

```
docker run -d -e LICENSE=accept -e MQ_QMGR_NAME=QM1 --volume /tmp/dockerVol:/mnt/mqm
--uts host --name sharedNamespace ibmcom/mq
```

3. Spusťte druhý kontejner s názvem `secondaryContainer`, který je založen na produktu `ibmcom/mq`, ale nevytvářejte správce front, zadáním následujícího příkazu:

```
docker run --entrypoint /bin/bash --volumes-from sharedNamespace --pid
container:sharedNamespace --ipc container:sharedNamespace --uts host --name
secondaryContainer -it --detach ibmcom/mq
```

4. Spusťte příkaz **dspmq** ve druhém kontejneru, abyste viděli stav obou správců front, zadáním následujícího příkazu:

```
docker exec secondaryContainer dspmq
```

5. Spusťte následující příkaz ke zpracování příkazů MQSC pro správce front spuštěného na jiném kontejneru:

```
docker exec -it secondaryContainer runmqsc QM1
```

Výsledky

Nyní máte lokální aplikace spuštěné v samostatných kontejnerech a můžete úspěšně spouštět příkazy jako **dspmq**, **amqsput**, **amqsget** a **runmqsc** jako lokální vazby ke správci front QM1 ze sekundárního kontejneru.

Pokud se nezobrazí očekávaný výsledek, přečtěte si další informace v "Odstraňování problémů s aplikacemi oboru názvů" na stránce 149.

Multi Odstraňování problémů s aplikacemi oboru názvů

Při používání sdílených oborů názvů musíte zajistit sdílení všech oborů názvů (IPC, PID a UTS/hostname) a připojených svazků, jinak vaše aplikace nebudou fungovat.

Seznam omezení, která musíte dodržovat, viz "Spuštění lokálních aplikací vazby v samostatných kontejnerech" na stránce 148.

Pokud vaše aplikace nesplňuje všechna uvedená omezení, můžete se setkat s problémy při spuštění kontejneru, ale funkčnost, kterou očekáváte, nebude fungovat.

Následující seznam popisuje některé běžné příčiny a chování, které se pravděpodobně zobrazí, pokud jste zapomněli splnit jedno z omezení.

- Pokud zapomenete sdílet buď obor názvů (UTS/PID/IPC), nebo název hostitele kontejnerů a poté svazek připojíte, bude kontejner schopen zobrazit správce front, ale nebude se správcem front spolupracovat.
  - V případě příkazů **dspmq** uvidíte následující:

```
docker exec container dspmq

QMNAME(QM1)                       STATUS(Status not available)
```

  - V případě příkazů **runmqsc** nebo jiných příkazů, které se pokusí připojit ke správci front, pravděpodobně obdržíte chybovou zprávu AMQ8146:

```
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
```

- Jestliže sdílíte všechny požadované obory názvů, ale nepřipojíte sdílený svazek k adresáři `/var/mqm` a máte platnou cestu k datům IBM MQ, pak vaše příkazy také přijímají chybové zprávy AMQ8146.

  Příkaz **dspmq** však není vůbec schopen zobrazit vašeho správce front, místo toho vrací prázdnou odezvu:

```
docker exec container dspmq
```

- Jestliže sdílíte všechny požadované obory názvů, ale nepřipojíte sdílený svazek k adresáři `/var/mqm` a máte platnou cestu k datům IBM MQ (nebo datovou cestu IBM MQ), zobrazí se různé chyby, protože cesta k datům je klíčovou komponentou instalace produktu IBM MQ. Bez cesty k datům produkt IBM MQ nemůže fungovat.

  Pokud spustíte kterýkoli z následujících příkazů a uvidíte odezvy podobné těm, které jsou zobrazeny v těchto příkladech, měli byste ověřit, zda jste připojili adresář nebo vytvořili datový adresář IBM MQ:

```
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 dltmqm QM1
AMQ7002: An error occurred manipulating a file.

docker exec container strmqweb
<file path>/mqrc.c[1152]
lpiObtainQMDetails --> 545261715
```

CP4I Vytvoření nativní skupiny HA při vytváření vlastních kontejnerů

Chcete-li vytvořit nativní skupinu HA, musíte vytvořit, nakonfigurovat a spustit tři správce front.

Informace o této úloze

Doporučená metoda pro vytvoření nativního řešení HA je pomocí operátoru IBM MQ (viz Nativní HA). Pokud vytváříte vlastní kontejnery, můžete postupovat podle těchto pokynů.

Chcete-li vytvořit nativní skupinu HA, vytvořte tři správce front ve třech uzlech s jejich typem protokolu nastaveným na hodnotu `log replication`. Pak upravíte soubor `qm.ini` pro každého správce front, chcete-li přidat podrobnosti o připojení pro každý ze tří uzlů tak, aby mohly replikovat data protokolu mezi sebou.

Pak musíte spustit všechny tři správce front, aby mohli zkontrolovat, zda všechny tři instance mohou navzájem komunikovat, a určit, která z nich bude aktivní instancí a které budou repliky.

Postup

1. Na každém ze tří uzlů vytvořte správce front s určením typu protokolu repliky protokolu a zadáním jedinečného názvu pro každou instanci protokolu. Každý správce front má stejný název:

   ```
   crtmqm -lr instance_name qmname
   ```

   Příklad:

   ```
   node 1> crtmqm -lr qm1_inst1 qm1

   node 2> crtmqm -lr qm1_inst2 qm1

   node 3> crtmqm -lr qm1_inst3 qm1
   ```

2. Při úspěšném vytvoření každého správce front je do konfiguračního souboru správce front `qm.ini`přidán další oddíl s názvem `NativeHALocalInstance` . Atribut `Name` se přidá do oddílu s uvedením dodaného názvu instance.

   Do oddílu `NativeHALocalInstance` v souboru `qm.ini` můžete volitelně přidat následující atributy:

   `KeyRepository`
   Umístění úložiště klíčů, které obsahuje digitální certifikát, který má být použit pro ochranu přenosu replikace protokolu. Umístění je uvedeno ve formátu kmene, tj. zahrnuje úplnou cestu a název souboru bez přípony. Je-li atribut stanza `KeyRepository` vynechán, data replikace protokolu se mezi instancemi vyměňují v prostém textu.

   `CertificateLabel`
   Označení certifikátu identifikující digitální certifikát, který má být použit pro ochranu přenosu replikace protokolu. Je-li `KeyRepository` zadán, ale `CertificateLabel` je vynechán, použije se standardní hodnota `ibmwebspheremq`*`queue_manager`* .

CipherSpec
Specifikace MQ CipherSpec , která má být použita k ochraně přenosu replikace protokolu. Je-li tento atribut stanzy zadán, musí být zadán také příznak KeyRepository . Je-li KeyRepository zadán, ale CipherSpec je vynechán, použije se standardní hodnota ANY .

LocalAddress
Adresa rozhraní lokálního síťového rozhraní, která přijímá provoz replikace protokolu. Je-li tento atribut stanzy zadán, identifikuje lokální síťové rozhraní a/nebo port ve formátu "[adresa] [(port)]". Síťová adresa může být zadána jako název hostitele, IPv4 desítková tečková konvence nebo hexadecimální formát IPv6 . Je-li tento atribut vynechán, pokusí se správce front o svázání se všemi síťovými rozhraními, použije port uvedený ve stanze ReplicationAddress ve stanze NativeHAInstances odpovídající názvu lokální instance.

HeartbeatInterval
Interval prezenčního signálu definuje, jak často, v milisekundách, aktivní instance správce front nativní vysoké dostupnosti odešle synchronizaci sítě. Platný rozsah hodnoty intervalu prezenčního signálu je 500 (0,5 s) do 60000 (1 min), hodnota mimo tento rozsah způsobí, že se správce front nespustí. Je-li tento atribut vynechán, použije se výchozí hodnota 5000 (5 s). Každá instance musí používat stejný interval prezenčního signálu.

HeartbeatTimeout
Časový limit prezenčního signálu definuje, jak dlouho bude instance repliky správce front nativní vysoké dostupnosti čekat, než se rozhodne, že aktivní instance nereaguje. Platný rozsah hodnoty časového limitu intervalu prezenčního signálu je 500 (0,5 s) do 120000 (2 min). Hodnota časového limitu prezenčního signálu musí být větší než nebo rovna intervalu prezenčního signálu.

Neplatná hodnota způsobí, že se správce front nespustí. Je-li tento atribut vynechán, čeká replika 2 x HeartbeatInterval před spuštěním procesu pro výběr nové aktivní instance. Každá instance musí používat stejný časový limit prezenčního signálu.

RetryInterval
Interval opakování definuje, jak často by se měl správce front nativní vysoké dostupnosti, v milisekundách, opakovat nezdařený odkaz na replikaci. Platný rozsah intervalu opakování je 500 (0,5 s) do 120000 (2 min). Je-li tento atribut vynechán, čeká replika 2 x HeartbeatInterval před zopakováním nezdařeného odkazu na replikaci.

3. Upravte soubor qm.ini pro každého správce front a přidejte podrobnosti o připojení. Přidáte tři sekce NativeHAInstance , jednu pro každou instanci správce front ve skupině s nativním vysokou dostupností (včetně lokální instance). Přidejte následující atributy:

   Název
   Určete název instance, který jste použili při vytvoření instance správce front.

   ReplicationAddress
   Zadejte název hostitele, IPv4 desítkovou tečkovou notaci nebo adresu v hexadecimálním formátu IPv6 instance. Adresu můžete uvést jako název hostitele, IPv4 desítkovou tečkovou adresu nebo hexadecimální adresu formátu IPv6 . Replikační adresa musí být rozlišitelná a směrovatelná z každé instance ve skupině. Číslo portu, které má být použito pro replikaci protokolu, musí být uvedeno v hranatých závorkách, například:

   ```
   ReplicationAddress=host1.example.com(4444)
   ```

   Poznámka: Stanzy NativeHAInstance jsou identické na všech instancích a mohly by být poskytnuty pomocí automatické konfigurace (**crtmqm -ii**).

4. Spusťte každou ze tří instancí:

   ```
   strmqm QMgrName
   ```

   Když jsou instance spuštěny, komunikují, aby zkontrolovali, zda jsou spuštěny všechny tři instance, pak se rozhodnout, která z těchto tří instancí je aktivní instancí, zatímco ostatní dvě instance jsou nadále spuštěny jako repliky.

Příklad

Následující příklad ukazuje sekci souboru `qm.ini` uvádějící požadované nativní podrobnosti HA pro jednu ze tří instancí:

```
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 Faktory ovlivňující provádění vlastní průběžné aktualizace správce front nativní vysoké dostupnosti

Jakákoli aktualizace verze produktu IBM MQ nebo specifikace podu pro správce front nativní vysoké dostupnosti bude vyžadovat provedení průběžné aktualizace instancí správce front. Produkt IBM MQ Operator ji automaticky zpracuje, ale pokud sestavujete vlastní kód implementace, pak jsou zde některé důležité aspekty.

Poznámka: Ukázka Helmova grafu obsahuje skript shellu k provedení průběžné aktualizace, ale tento skript **není** vhodný pro provozní použití, protože se nezabývá otázkami v tomto tématu.

V produktu Kubernetes se prostředky `StatefulSet` používají ke správě seřazených aktualizací při spuštění a průběžných aktualizací. Součástí procedury spuštění je spuštění každého podu jednotlivě, vyčkání na jeho připravenost a pak přesun na další pod. To nebude fungovat pro nativní HA, protože všechny pody musí být spuštěny, aby mohly spustit volby vůdce. Proto musí být pole `.spec.podManagementPolicy` v produktu `StatefulSet` nastaveno na `Parallel`. To také znamená, že budou všechny pody aktualizovány také paralelně, což je zvláště nežádoucí. Z tohoto důvodu by měl produkt `StatefulSet` také použít strategii aktualizace `OnDelete`.

Neschopnost používat kód průběžné aktualizace `StatefulSet` vyžaduje potřebu vlastního kódu průběžné aktualizace, který by měl zvážit následující:

- Postup běžné průběžné aktualizace
- Minimalizace výpadku aktualizací podů v nejlepším pořadí
- Zpracování změn ve stavu klastru
- Ošetření chyb
- Práce s problémy časování

Postup běžné průběžné aktualizace

Kód průběžné aktualizace by měl čekat na každou instanci, aby se zobrazil stav `REPLICA` z `dspmq`. To znamená, že instance provedla určitou úroveň spuštění (je například spuštěn kontejner a jsou spuštěny procesy MQ), ale zatím nutně nemusela vést konverzaci s ostatními instancemi. Například: Pod A se restartuje a jakmile je ve stavu `REPLICA` , pod B se restartuje. Jakmile je spuštěn Pod B s novou konfigurací, měl by být schopen komunikovat s Podem A a může formovat kvorum, a buď A, nebo B se stane novou aktivní instancí.

Je proto užitečné mít zpoždění poté, co každý pod dosáhne stavu `REPLICA`, aby mu bylo umožněno připojit se ke svým protějškům a ustavit kvorum.

Minimalizace výpadku aktualizací podů v nejlepším pořadí

Kód průběžné aktualizace by měl odstranit pody jeden po druhém, počínaje pody, které jsou ve známém chybovém stavu, a poté jakékoli pody, které se nespustily úspěšně. Aktivní pod správce front by měl být obecně aktualizován naposledy.

Je také důležité pozastavit odstranění podů, pokud poslední aktualizace vedla k tomu, že pod vstoupil do známého chybového stavu. Tím se zabrání provedení přerušené aktualizace ve všech podech. K tomu může dojít například v případě, když je pod aktualizován pro použití nového obrazu kontejneru, který není přístupný (nebo obsahuje překlep).

Zpracování změn ve stavu klastru

Kód průběžné aktualizace musí vhodně reagovat na změny v reálném čase ve stavu klastru. Jeden z podů správce front může být například vypovězen kvůli opětnému zavedení uzlu nebo kvůli tlaku uzlu. Je možné, že nemusí být vypovězený pod ihned znovu naplánován, pokud je klastr zaneprázdněn. V takovém případě by měl být kód průběžné aktualizace před restartováním jakýchkoli jiných podů čekat odpovídajícím způsobem.

Ošetření chyb

Kód průběžní aktualizace musí být odolný vůči selháním při volání rozhraní API Kubernetes a jiného neočekávaného chování klastru.

Kromě toho musí být samotný kód průběžné aktualizace tolerantní k restartování. Průběžná aktualizace může být spuštěna dlouho a může být nutné restartovat kód.

Práce s problémy časování

Kód průběžné aktualizace potřebuje zkontrolovat revize aktualizace podu, aby mohl zajistit, že je pod restartován. Tím se vyvarujete problémů s časováním, kdy může pod označovat, že je "Spuštěný", ale ve skutečnosti ještě není ukončený.

Související pojmy

“Zvolení, jak se má produkt IBM MQ používat v kontejnerech” na stránce 5
Existuje více voleb pro použití produktu IBM MQ v kontejnerech: můžete zvolit použití IBM MQ Operator, který používá předem seskupené kontejnerové obrazy nebo můžete sestavit vlastní obrazy a kód implementace.

CP4I Zobrazení stavu správců front nativních HA pro kontejnery přizpůsobené sestavení

V případě kontejnerů s přizpůsobením obsahu můžete zobrazit stav nativních instancí vysoké dostupnosti pomocí příkazu **dspmq** .

Informace o této úloze

Chcete-li zobrazit provozní stav instance správce front v uzlu, můžete použít příkaz **dspmq** . Vrácené informace závisí na tom, zda je instance aktivní nebo zda je to replika. Informace poskytnuté aktivní instancí jsou konečné, informace z replikovaných uzlů mohou být zastaralé.

Můžete provést následující akce:

- Zobrazit, zda je instance správce front v aktuálním uzlu aktivní nebo zda je to replika.
- Zobrazit provozní stav nativní vysoké dostupnosti instance v aktuálním uzlu.
- Zobrazit provozní stav všech tří instancí v konfiguraci nativní vysoké dostupnosti.

Následující stavová pole se používají k hlášení stavu konfigurace nativní vysoké dostupnosti:

ROLE
Určuje aktuální roli instance a je jednou z hodnot `Active`, `Replica` nebo `Unknown`.

INSTANCE
Název poskytnutý pro tuto instanci správce front, když byl vytvořen pomocí volby **`-lr`** příkazu **`crtmqm`**.

INSYNC
Určuje, zda je instance v případě potřeby schopna převzít funkci aktivní instance.

QUORUM
Hlásí stav kvora ve formátu *počet_synchronizovaných_instancí*/*počet_nakonfigurovaných_instancí*.

REPLADDR
Adresa replikace instance správce front.

CONNACTV
Označuje, zda je uzel připojen k aktivní instanci.

BACKLOG
Označuje, kolik kB instance překročila.

CONNINST
Označuje, zda je pojmenovaná instance připojena k této instanci.

ALTDATE
Označuje datum, kdy byly tyto informace naposledy aktualizovány (prázdné, pokud dosud nebyly aktualizovány).

ALTTIME
Označuje čas poslední aktualizace těchto informací (prázdné, pokud dosud nebyla aktualizována).

Procedura

- Chcete-li určit, zda je instance správce front spuštěna jako aktivní instance nebo jako replika:

  ```
  dspmq -o status -m QMgrName
  ```

  Aktivní instance správce front s názvem BOB bude vykazovat následující stav:

  ```
  QMNAME(BOB)                STATUS(Running)
  ```

  Instance repliky správce front s názvem BOB bude vykazovat následující stav:

  ```
  QMNAME(BOB)                STATUS(Replica)
  ```

  Neaktivní instance bude hlásit následující stav:

  ```
  QMNAME(BOB)                STATUS(Ended Immediately)
  ```

- Chcete-li určit nativní provozní stav HA instance na aktuálním uzlu, postupujte takto:

  ```
  dspmq -o nativeha -m QMgrName
  ```

  Aktivní instance správce front s názvem BOB může vykazovat následující stav:

  ```
  QMNAME(BOB)              ROLE(Active) INSTANCE(inst1) INSYNC(Yes) QUORUM(3/3)
  ```

  Instance repliky správce front s názvem BOB může vykazovat následující stav:

  ```
  QMNAME(BOB)              ROLE(Replica) INSTANCE(inst2) INSYNC(Yes) QUORUM(2/3)
  ```

  Neaktivní instance správce front s názvem BOB může vykazovat následující stav:

  ```
  QMNAME(BOB)              ROLE(Unknown) INSTANCE(inst3) INSYNC(no) QUORUM(0/3)
  ```

- Chcete-li určit provozní stav nativní vysoké dostupnosti všech instancí v konfiguraci nativní vysoké dostupnosti:

  ```
  dspmq -o nativeha -x -m QMgrName
  ```

  Pokud tento příkaz zadáte na uzlu, na kterém je spuštěna aktivní instance správce front BOB, můžete obdržet následující stav:

  ```
  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) ALTDATE(2022-01-12) ALTTIME(12.03.44)
   INSTANCE(inst2) ROLE(Replica) REPLADDR(9.20.123.46) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
  CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
   INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
  CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
  ```

  Pokud tento příkaz zadáte na uzlu, na kterém je spuštěna instance repliky správce front BOB, můžete obdržet následující stav, který znamená, že jedna z replik zaostává:

  ```
  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) ALTDATE(2022-01-12) ALTTIME(12.03.44)
   INSTANCE(inst1) ROLE(Active)  REPLADDR(9.20.123.45) CONNACTV(Yes) INSYNC(Yes) BACKLOG(0)
  CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
   INSTANCE(inst3) ROLE(Replica) REPLADDR(9.20.123.47) CONNACTV(Yes) INSYNC(No)  BACKLOG(435)
  CONNINST(Yes) ALTDATE(2022-01-12) ALTTIME(12.03.44)
  ```

  Pokud tento příkaz zadáte na uzlu, kde je spuštěna neaktivní instance správce front BOB, můžete obdržet následující stav:

  ```
  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) ALTDATE() ALTTIME()
   INSTANCE(inst2) ROLE(Unknown) REPLADDR(9.20.123.46) CONNACTV(Unknown) INSYNC(Unknown)
  BACKLOG(Unknown) CONNINST(No) ALTDATE() ALTTIME()
   INSTANCE(inst3) ROLE(Unknown) REPLADDR(9.20.123.47) CONNACTV(No) INSYNC(Unknown)
  BACKLOG(Unknown) CONNINST(No) ALTDATE() ALTTIME()
  ```

  Pokud zadáte příkaz, když se instance ještě domlouvají, která je aktivní a které jsou repliky, obdržíte následující stav:

  ```
  QMNAME(BOB)              STATUS(Negotiating)
  ```

Související odkazy
dspmq

CP4I Ukončení nativních správců front HA

Příkaz **endmqm** můžete použít k ukončení aktivního nebo replikovaného správce front, který je součástí nativní skupiny HA.

Procedura

- Chcete-li ukončit aktivní instanci správce front, přečtěte si téma Ukončení nativních správců front HA v sekci Konfigurace této dokumentace.

Poznámky

Tyto informace byly vyvinuty pro produkty a služby poskytované v USA.

Společnost IBM nemusí nabízet produkty, služby nebo funkce uvedené v tomto dokumentu v jiných zemích. Informace o produktech a službách, které jsou ve vaší oblasti aktuálně dostupné, získáte od místního zástupce společnosti IBM. Odkazy na produkty, programy nebo služby společnosti IBM v této publikaci nejsou míněny jako vyjádření nutnosti použití pouze uvedených produktů, programů či služeb společnosti IBM. Místo toho lze použít jakýkoli funkčně ekvivalentní produkt, program nebo službu, které neporušují žádná práva k duševnímu vlastnictví IBM. Ověření funkčnosti produktu, programu nebo služby pocházející od jiného výrobce je však povinností uživatele.

Společnost IBM může vlastnit patenty nebo nevyřízené žádosti o patenty zahrnující předměty popsané v tomto dokumentu. Vlastnictví tohoto dokumentu neposkytuje licenci k těmto patentům. Dotazy týkající se licencí můžete posílat písemně na adresu:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Odpovědi na dotazy týkající se licencí pro dvoubajtové znakové sady (DBCS) získáte od oddělení IBM Intellectual Property Department ve vaší zemi, nebo tyto dotazy můžete zasílat písemně na adresu:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

Následující odstavec se netýká Velké Británie nebo kterékoliv jiné země, kde taková opatření odporují místním zákonům: SPOLEČNOST INTERNATIONAL BUSINESS MACHINES CORPORATION TUTO PUBLIKACI POSKYTUJE TAKOVOU, "JAKÁ JE" , BEZ JAKÝCHKOLIV ZÁRUK, VYJÁDŘENÝCH VÝSLOVNĚ NEBO VYPLÝVAJÍCÍCH Z OKOLNOSTÍ, VČETNĚ, A TO ZEJMÉNA, ZÁRUK NEPORUŠENÍ PRÁV TŘETÍCH STRAN, PRODEJNOSTI NEBO VHODNOSTI PRO URČITÝ ÚČEL VYPLÝVAJÍCÍCH Z OKOLNOSTÍ. Některé právní řády u určitých transakcí nepřipouštějí vyloučení záruk výslovně vyjádřených nebo vyplývajících z okolností, a proto se na vás toto omezení nemusí vztahovat.

Uvedené údaje mohou obsahovat technické nepřesnosti nebo typografické chyby. Údaje zde uvedené jsou pravidelně upravovány a tyto změny budou zahrnuty v nových vydáních této publikace. Společnost IBM může kdykoli bez upozornění provádět vylepšení nebo změny v produktech či programech popsaných v této publikaci.

Veškeré uvedené odkazy na webové stránky, které nespravuje společnost IBM, jsou uváděny pouze pro referenci a v žádném případě neslouží jako záruka funkčnosti těchto webů. Materiály uvedené na tomto webu nejsou součástí materiálů pro tento produkt IBM a použití uvedených stránek je pouze na vlastní nebezpečí.

Společnost IBM může použít nebo distribuovat jakékoli informace, které jí sdělíte, libovolným způsobem, který společnost považuje za odpovídající, bez vyžádání vašeho svolení.

Vlastníci licence k tomuto programu, kteří chtějí získat informace o možnostech (i) výměny informací s nezávisle vytvořenými programy a jinými programy (včetně tohoto) a (ii) oboustranného využití vyměňovaných informací, mohou kontaktovat informační středisko na adrese:

IBM Corporation
Koordinátor spolupráce softwaru, oddělení 49XA
148 00 Praha 4-Chodby

148 00 Praha 4-Chodov
U.S.A.

Poskytnutí takových informací může být podmíněno dodržením určitých podmínek a požadavků zahrnujících v některých případech uhrazení stanoveného poplatku.

IBM poskytuje licencovaný program popsaný v těchto informacích a veškeré dostupné licencované materiály na základě podmínek smlouvy IBM Customer Agreement, IBM International Program License Agreement nebo jiné ekvivalentní smlouvy mezi námi.

Jakékoli údaje o výkonnosti obsažené v této publikaci byly zjištěny v řízeném prostředí. Výsledky získané v jakémkoli jiném operačním prostředí se proto mohou výrazně lišit. Některá měření mohla být prováděna na vývojových verzích systémů a není zaručeno, že tato měření budou stejná i na běžně dostupných systémech. Některá měření mohla být navíc odhadnuta pomocí extrapolace. Skutečné výsledky mohou být jiné. Čtenáři tohoto dokumentu by měli zjistit použitelné údaje pro své specifické prostředí.

Informace týkající se produktů jiných výrobců pocházejí od dodavatelů těchto produktů, z jejich veřejných oznámení nebo z jiných veřejně dostupných zdrojů. Společnost IBM tyto produkty netestovala a nemůže potvrdit správný výkon, kompatibilitu ani žádné jiné výroky týkající se produktů jiných výrobců než IBM. Otázky týkající se kompatibility produktů jiných výrobců by měly být směrovány dodavatelům těchto produktů.

Veškerá tvrzení týkající se budoucího směru vývoje nebo záměrů společnosti IBM se mohou bez upozornění změnit nebo mohou být zrušena a reprezentují pouze cíle a plány společnosti.

Tyto údaje obsahují příklady dat a sestav používaných v běžných obchodních operacích. Aby byla představa úplná, používají se v příkladech jména osob a názvy společností, značek a produktů. Všechna tato jména a názvy jsou fiktivní a jejich podobnost se jmény, názvy a adresami používanými ve skutečnosti je zcela náhodná.

LICENČNÍ INFORMACE:

Tyto informace obsahují ukázkové aplikační programy ve zdrojovém jazyce ilustrující programovací techniky na různých operačních platformách. Tyto ukázkové programy můžete bez závazků vůči společnosti IBM jakýmkoli způsobem kopírovat, měnit a distribuovat za účelem vývoje, používání, odbytu či distribuce aplikačních programů odpovídajících rozhraní API pro operační platformu, pro kterou byly ukázkové programy napsány. Tyto příklady nebyly plně testovány za všech podmínek. Společnost IBM proto nemůže zaručit spolehlivost, upotřebitelnost nebo funkčnost těchto programů.

Při prohlížení těchto dokumentů v elektronické podobě se nemusí zobrazit všechny fotografie a barevné ilustrace.

Informace o programovacím rozhraní

Informace programátorských rozhraní, jsou-li poskytovány, jsou určeny k tomu, aby vám pomohly vytvořit aplikační software pro použití s tímto programem.

Tato příručka obsahuje informace o zamýšlených programovacích rozhraních, které umožňují zákazníkům psát programy za účelem získání služeb produktu WebSphere MQ.

Tyto informace však mohou obsahovat i diagnostické údaje a informace o úpravách a ladění. Informace o diagnostice, úpravách a vyladění jsou poskytovány jako podpora ladění softwarových aplikací.

Důležité: Nepoužívejte tyto informace o diagnostice, úpravách a ladění jako programátorské rozhraní, protože se mohou měnit.

Ochranné známky

IBM, logo IBM , ibm.comjsou ochranné známky společnosti IBM Corporation, registrované v mnoha jurisdikcích po celém světě. Aktuální seznam ochranných známek IBM je k dispozici na webu na stránce "Copyright and trademark information"www.ibm.com/legal/copytrade.shtml. Ostatní názvy produktů a služeb mohou být ochrannými známkami společnosti IBM nebo jiných společností.

Microsoft a Windows jsou ochranné známky společnosti Microsoft Corporation ve Spojených státech a případně v dalších jiných zemích.

UNIX je registrovaná ochranná známka skupiny The Open Group ve Spojených státech a případně v dalších jiných zemích.

Linux je registrovaná ochranná známka Linuse Torvaldse ve Spojených státech a případně v dalších jiných zemích.

Tento produkt obsahuje software vyvinutý v rámci projektu Eclipse Project (https://www.eclipse.org/).

Java a všechny ochranné známky a loga založené na termínu Java jsou ochranné známky nebo registrované ochranné známky společnosti Oracle anebo příbuzných společností.

Číslo položky:

(1P) P/N: