

介紹RED HAT AMQ 7

紅帽AMQ 7.2

功能和組件概述

法律聲明

抽象

本文檔重點介紹了Red Hat AMQ 7的功能和組件。它還演示了此產品版本中解決的常見用例和設計模式。

第1章關於Red Hat AMQ 7

Red Hat AMQ為Internet規模的應用程序提供快速，輕量級和安全的消息傳遞。AMQ Broker支持多種協議和快速消息持久性。AMQ Interconnect利用AMQP協議在整個網絡中分發和擴展您的消息資源。AMQ客戶端為多種語言和平台提供了一套消息傳遞API。

將AMQ組件視為工具箱中的工具。它們可以一起使用或單獨使用來構建和維護您的消息傳遞應用程序，而AMQP是工具箱中將它們綁定在一起的粘合劑。AMQ組件共享一個公共管理控制台，因此您可以從單個界面管理它們。

1.1。主要特點

AMQ使開發人員能夠構建快速，可靠且易於管理的消息傳遞應用程序。

互聯網規模的消息傳遞

AMQ包含構建高級多數據中心消息傳遞網絡的工具。它可以在無縫消息傳遞結構中連接客戶端，代理和獨立服務。

頂級安全性和性能

AMQ提供現代SSL / TLS加密和可擴展SASL身份驗證。AMQ提供快速，高容量的消息傳遞和領先的JMS性能。

廣泛的平台和語言支持

AMQ適用於多種語言和操作系統，因此您的各種應用程序組件可以進行通信。AMQ支持C ++，Java，JavaScript，Python，Ruby和.NET應用程序，以及基於Linux，Windows和JVM的環境。

專注於標準

AMQ實現了Java JMS 1.1和2.0 API規範。其組件支持ISO標準AMQP 1.0和MQTT消息傳遞協議，以及STOMP和WebSocket。

集中管理

使用AMQ，您可以從單個管理界面管理所有AMQ組件。您可以使用JMX或REST接口以編程方式管理服務器。

第2章組件概述

Red Hat AMQ由[AMQ Broker](#)，[AMQ客戶端](#)和一種名為[AMQ Interconnect](#)的新型消息服務器組成。它們協同工作以實現分佈式客戶端應用程序之間 [AMQ控制台](#)是用於監視和管理AMQ部署的通用接口。

- [AMQ經紀人](#)
- [AMQ客戶](#)
- [AMQ互連](#)
- [AMQ控制台](#)

2.1。AMQ經紀人

AMQ Broker是一個功能齊全，面向消息的中間件代理。它提供高級尋址和排隊，快速消息持久性和高可用性。AMQ Broker支持多種協議和操作環境，使您可以使用現有資產。AMQ Broker支持與紅帽JBoss企業應用平台的集成。

有關更多信息，請參閱[使用AMQ Broker](#)。

2.2。AMQ互連

AMQ Interconnect在支持AMQP的端點之間提供靈活的消息路由，包括客戶端，代理和獨立服務。通過與AMQ互連路由器網絡的單一連接，客戶端可以與連接到網絡的任何其他端點交換消息。

AMQ互連不使用主從集群來實現高可用性。它通常部署在具有冗餘網絡路徑的多個路由器的拓撲中，用於提供可靠的連接。AMQ Interconnect可以在整個網絡中分發消息傳遞工作負載，並以極低的延遲實現新的擴展級別。

有關更多信息，請參閱[使用AMQ互連](#)。

2.3。AMQ客戶

AMQ客戶端是適用於多種語言和平台的AMQP 1.0消息傳遞API的集合。它包括JMS 2.0支持和新的事件驅動API，以實現與現有应用程序的集成。

- [使用AMQ JMS客戶端](#)
- [使用AMQ C ++客戶端](#)
- [使用AMQ JavaScript客戶端](#)
- [使用AMQ .NET客戶端](#)
- [使用AMQ Python客戶端](#)
- [使用AMQ Ruby Client](#)

2.4。AMQ控制台

AMQ控制台是一個用於在一個方便的界面中管理AMQ代理和路由器的工具。從AMQ控制台，您可以管理地址，隊列，連接和其他管理對象。

有關更多信息，請參閱[使用AMQ控制台](#)。

2.5。組件兼容性

下表列出了AMQ組件支持的語言，平台和協議。請注意，支持相同協議的任何組件都可以互操作，即使它們的語言和平台不同。例如，AMQ Python可以與AMQ JMS通信。

表2.1。AMQ組件兼容性

零件	語言	平台	協議
AMQ經紀人	-	JVM	AMQP 1.0， MQTT， OpenWire， STOMP，核心 協議
AMQ互連	-	Linux的	AMQP 1.0
AMQ C ++	C ++	Linux， Windows	AMQP 1.0
AMQ JavaScript	JavaScript的	Node.js，瀏覽 器	AMQP 1.0
AMQ JMS	Java的	JVM	AMQP 1.0
AMQ .NET	C #	。淨	AMQP 1.0
AMQ Python	蟒蛇	Linux的	AMQP 1.0
AMQ Ruby	紅寶石	Linux的	AMQP 1.0
AMQ核心協議 JMS	Java的	JVM	核心協議
AMQ OpenWire JMS	Java的	JVM	OpenWire

有關更多信息，請參閱[Red Hat AMQ 7支持的配置](#)。

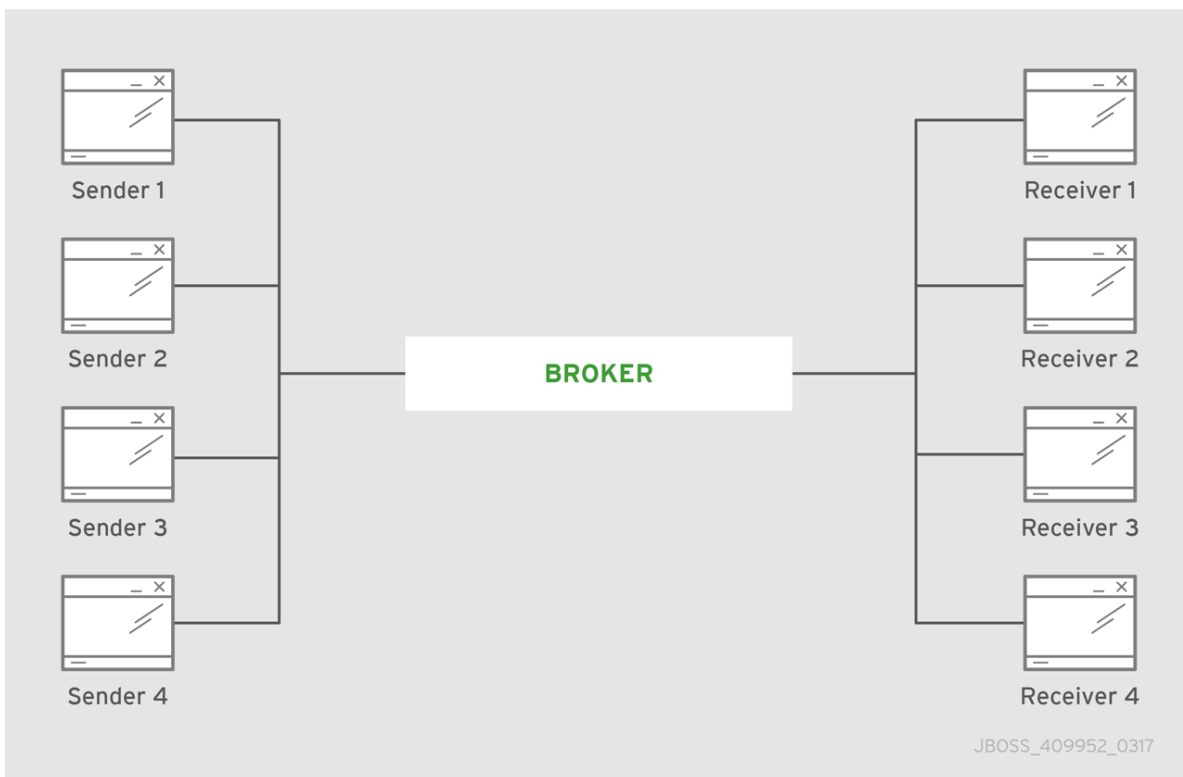
第3章常見部署模式

Red Hat AMQ 7可以在各種拓撲中設置。以下是您可以使用AMQ組件實現的一些常見部署模式。

3.1。中央經紀人

中央經紀人模式相對容易設置和維護。它也相對強大。路由通常是本地路由，因為無論添加多少個節點，代理及其客戶端始終在彼此的一個網絡躍點內。這種模式也稱為**集線器和輻條**，中央代理作為集線器，客戶端稱為輻條。

圖3.1。中央經紀人模式

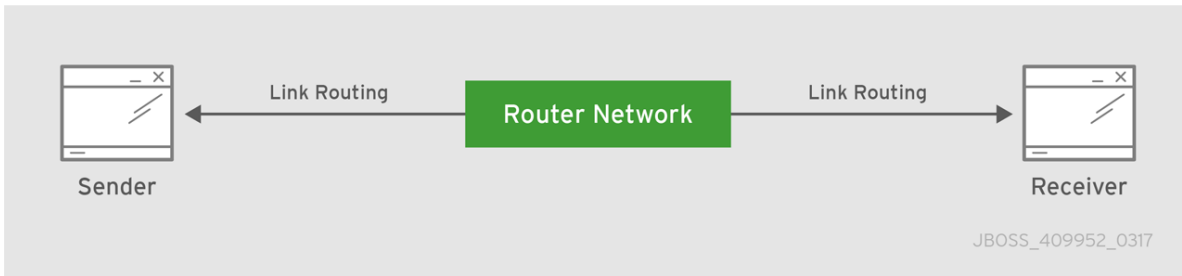


唯一關鍵的元素是中央代理節點。您的維護工作的重點是保持這個經紀人可供其客戶使用。

3.2。路由消息傳遞

將消息路由到遠程目標時，代理會將它們存儲在本地隊列中，然後再將它們轉發到目標。但是，有時應用程序需要實時發送請求和響應消息，並且擁有代理存儲和轉發消息的成本太高。使用AMQ，您可以使用路由器代替代理來避免此類成本。與代理不同，路由器在將消息轉發到目標之前不會存儲消息。相反，它作為輕量級管道工作，並直接連接兩個端點。

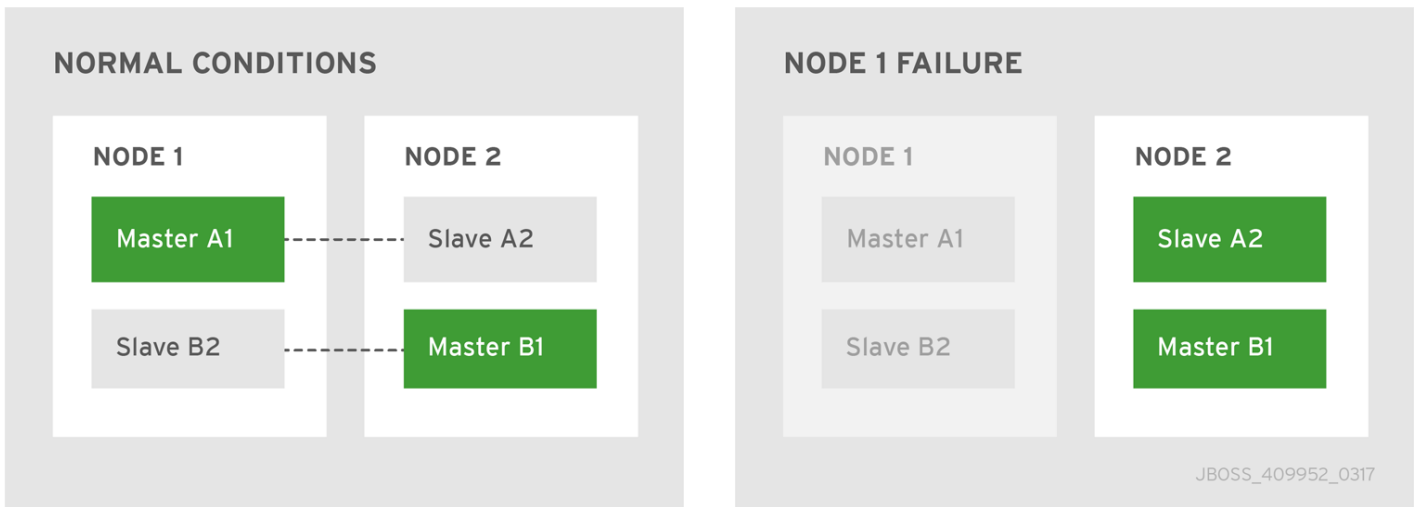
圖3.2。無破解路由消息傳遞模式



3.3。高度可用的經紀人

要確保為其客戶端提供代理，請部署高可用性（HA）主從對以創建備份組。例如，您可以在兩個節點上部署兩個主從組。這樣的部署將為每個活動代理提供備份，如下圖所示。

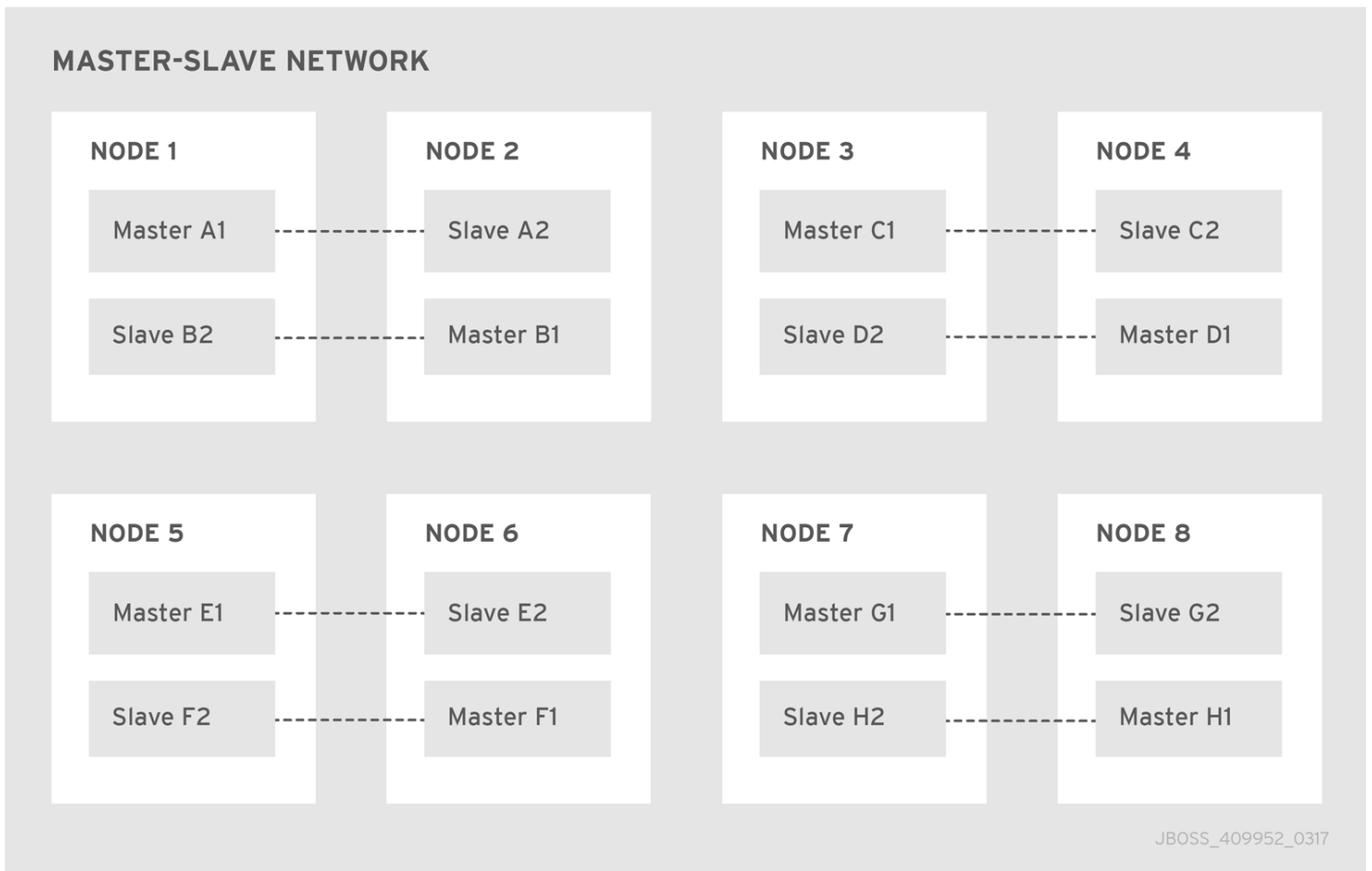
圖3.3。主從配對



在正常操作條件下，每個節點上都有一個主代理處於活動狀態，該節點可以是物理服務器或虛擬機。如果一個節點發生故障，另一個節點上的從站將接管。結果是兩個活動的代理駐留在同一個健康節點上。

通過部署主從對，您可以擴展此類備份組的整個網絡。此類型的較大部署對於跨多個代理分發消息處理負載非常有用。下圖中的代理網絡由分佈在八個節點上的八個主從組組成。

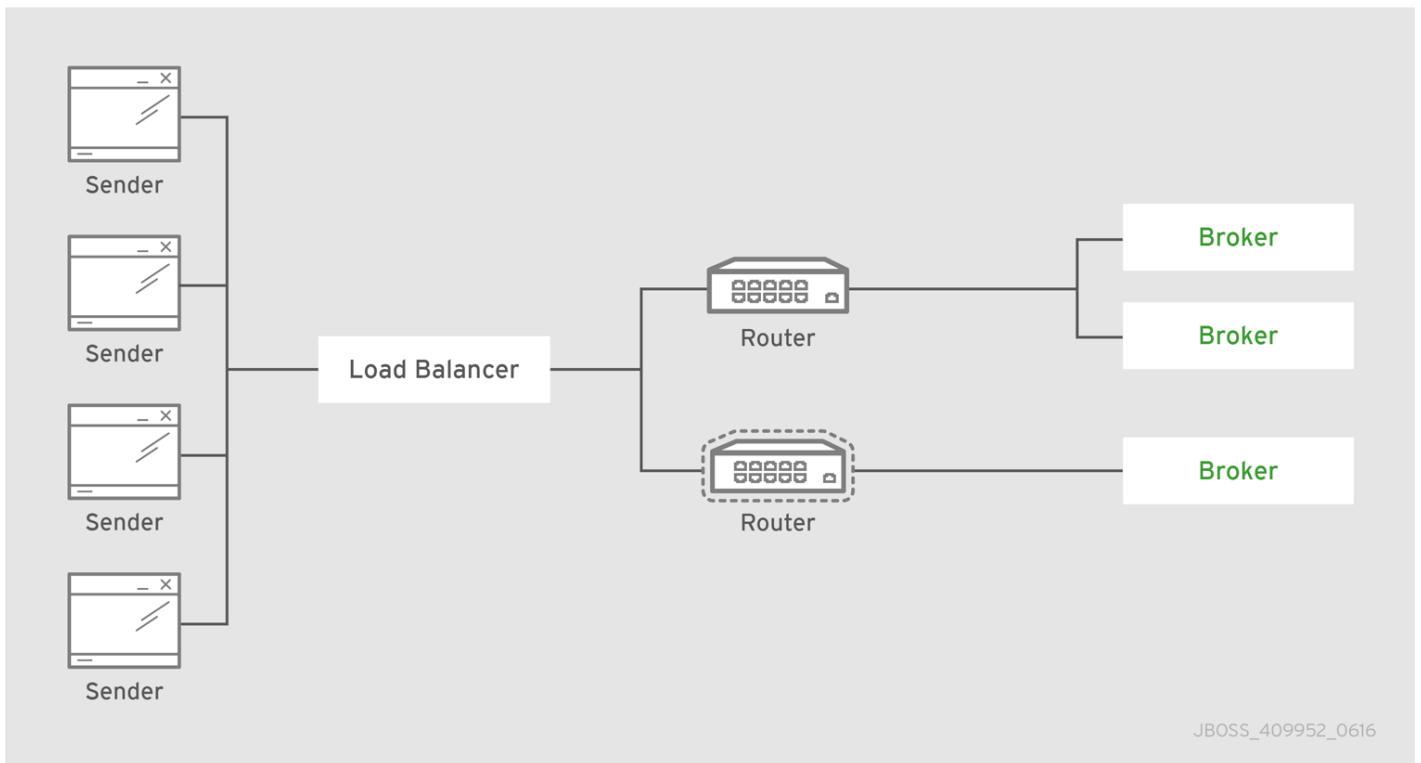
圖3.4。主從網絡



3.4。負載均衡器後面的路由器對

在負載均衡器後面部署兩個路由器可為單數據中心部署提供高可用性，彈性和增強的可擴展性。端點與負載均衡器支持的已知URL建立連接。接下來，負載均衡器在路由器之間傳播傳入連接，以便分配連接和消息傳遞負載。如果其中一個路由器出現故障，連接到它的端點將重新連接到剩餘的活動路由器。

圖3.5。負載均衡器後面的路由器對

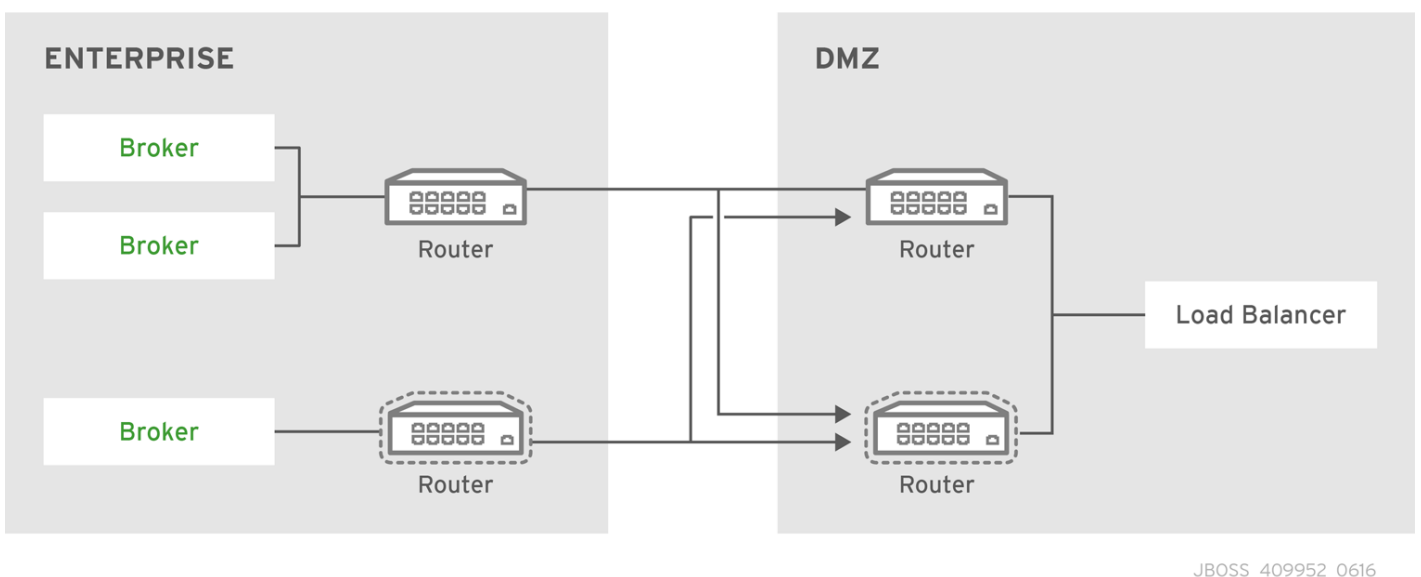


為了獲得更高的可擴展性，您可以使用更多的路由器，例如三個或四個。每個路由器直接連接到所有其他路由器。

3.5。DMZ中的路由器對

在此部署體系結構中，路由器網絡在外部客戶端和支持企業應用程序的代理之間提供了一層保護和隔離。

圖3.6。DMZ中的路由器對



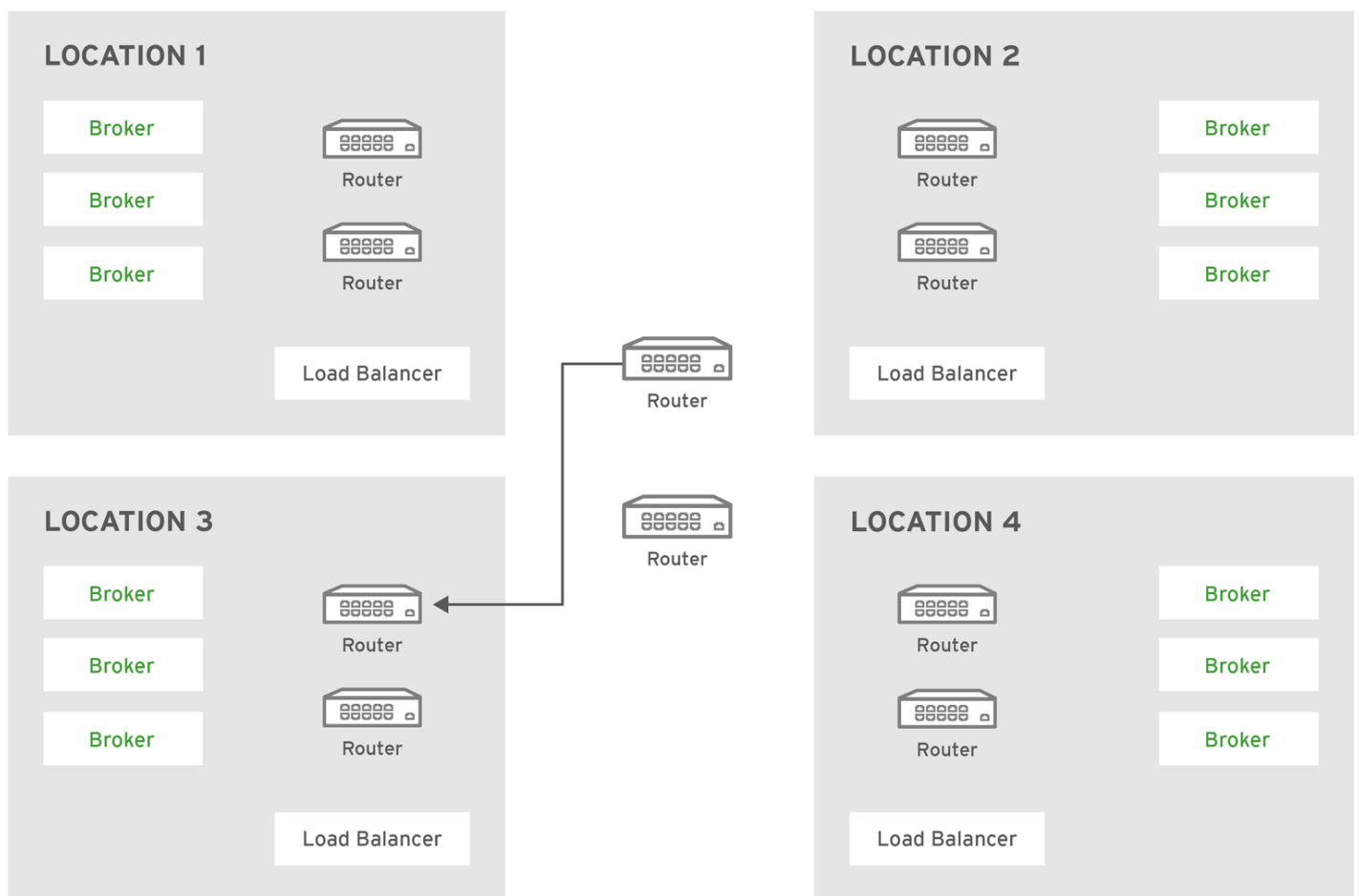
有關DMZ拓撲的重要說明：

- 部署中的連接的安全性與用於外部客戶端的安全性是分開的。例如，您的部署可能使用專用證書頒發機構（CA）進行內部安全性，向每個路由器和代理髮出x.509證書以進行身份驗證，但外部用戶可能使用不同的公共CA。
- 為了安全起見，企業和DMZ之間的路由器間連接總是從企業到DMZ建立。因此，不允許從外部進入企業的连接。然而，AMQP協議在建立連接之後實現雙向通信。

3.6。不同數據中心的路由器對

您可以在跨多個位置的AMQ組件部署中使用更複雜的拓撲。例如，您可以在四個位置中的每個位置部署一對負載平衡路由器。您可能在中心包括兩個骨幹路由器，以在所有位置之間提供冗餘連接。下圖是跨多個位置的示例部署。

圖3.7。多個互連路由器



JBoss_409952_0616

修訂於2018-07-16 16:08:02 EDT

法律聲明

版權所有©2018 Red Hat，Inc。

本文檔中的文字和插圖由Red Hat根據知識共享署名 - 相同方式共享3.0 Unported許可（“CC-BY-SA”）許可。有關CC-BY-SA的說明，請訪問<http://creativecommons.org/licenses/by-sa/3.0/>。根據CC-BY-SA，如果您分發此文檔或對其進行修改，則必須提供原始版本的URL。

作為本文件的許可方，Red Hat放棄在適用法律允許的最大範圍內強制執行並同意不主張CC-BY-SA第4d節的權利。

Red Hat，Red Hat Enterprise Linux，Shadowman徽標，JBoss，OpenShift，Fedora，Infinity徽標和RHCE是Red Hat，Inc。在美國和其他國家/地區註冊的商標。

Linux的®是Linus Torvalds在美國和其他國家的註冊商標。

的Java®是Oracle和/或其附屬公司的註冊商標。

XFS®是Silicon Graphics International Corp.或其子公司在美國和/或其他國家/地區的商標。

MySQL®是MySQL AB在美國，歐盟和其他國家/地區的註冊商標。

Node.js®是Joyent的官方商標。Red Hat Software Collections與官方Joyent Node.js開源或商業項目沒有正式關聯或認可。

在OpenStack的®字標和OpenStack的標識是註冊商標/服務商標或註冊商標/在的OpenStack基金會的服務商標，在美國和其他國家，並與OpenStack的基金會的許可使用。我們與OpenStack Foundation或OpenStack社區沒有任何關聯，認可或贊助。

所有其他商標均為其各自所有者的財產。