



APUSIC
固若长城
睿比世界

安裝手冊

金蝶Apusic分布式消息队列V2.0.6_for_kafka

版權所有 © 深圳市金蝶天燕雲計算股份有限公司2026。保留所有權利。

版權聲明

本文件所涉及的軟件著作權、版權等知識產權已依法進行了註冊，由金蝶天燕雲計算股份有限公司合法擁有。受《中華人民共和國著作權法》《計算機軟件保護條例》《知識產權保護條例》和相關國際版權條約、法律、法規以及其它知識產權法律和條約的保護。未經授權許可，不得非法使用。

免責聲明

本文件包含的版權資訊由金蝶天燕雲計算股份有限公司合法擁有，受法律的保護，金蝶天燕雲計算股份有限公司對本文件可能涉及到的非金蝶天燕雲計算股份有限公司的資訊不承擔任何責任。在法律允許的範圍內，您可以查閱並僅能夠在《中華人民共和國著作權法》規定的合法範圍內複製和打印本文件。任何單位和個人未經金蝶天燕雲計算股份有限公司書面授權許可，不得使用、修改、再發佈本文件的任何部分和內容，否則將被視為侵權，金蝶天燕雲計算股份有限公司有依法追究其責任的權利。

本文件如有更新，不另行通知。對本文件中的問題您可向金蝶天燕雲計算股份有限公司告知或查詢。未經本公司明確授予的任何權利均予保留。

商標聲明

 是深圳市金蝶天燕雲計算股份有限公司向中華人民共和國國家商標局申請註冊的註冊商標，註冊商標專用權由金蝶天燕合法擁有，受法律保護。未經金蝶天燕的書面許可，任何單位及個人不得以任何方式或理由對該商標的任何部分進行使用、複製、修改、傳播、抄錄或與其它產品捆綁使用銷售。凡侵犯金蝶天燕商標權的，金蝶天燕將依法追究其法律責任。本文件提及的其他所有商標或註冊商標，由各自的所有人擁有。

目錄

- 1 前言
 - 1.1 適用對象
 - 1.2 相關文檔
 - 1.3 技術支援
- 2 概述
- 3 部署架構圖
 - 3.1 單機部署架構
 - 3.2 叢集部署架構
 - 3.3 混合部署架構
- 4 安裝前準備
 - 4.1 系統要求
 - 4.2 連接埠檢查
 - 4.3 關閉防火牆（測試環境）
- 5 單機部署
 - 5.1 1. 建立部署目錄
 - 5.2 2. 下載並解壓縮軟體包
 - 5.3 3. 目錄結構說明
 - 5.4 4. 配置修改
 - 5.5 5. 授權許可
 - 5.6 6. 啟動服務
 - 5.6.1 6.1 啟動 ADCC for zk
 - 5.6.2 6.2 啟動 Broker
 - 5.7 7. 服務驗證
 - 5.7.1 7.1 檢查行程狀態
 - 5.7.2 7.2 檢查連接埠監聽
 - 5.7.3 7.3 功能測試
 - 5.8 8. 停止服務
- 6 叢集部署
 - 6.1 1. 部署規劃
 - 6.2 2. 所有節點執行：基礎安裝
 - 6.3 3. 放置授權檔案
 - 6.4 4. 配置 ADCC for zk 叢集
 - 6.4.1 4.1 修改 ADCC for zk 設定檔
 - 6.4.2 4.2 建立 MyID 檔案
 - 6.5 5. 配置 Broker 叢集

- 6.5.1 5.1 Node 1 配置 (192.168.1.10)
- 6.5.2 5.2 Node 2 配置 (192.168.1.11)
- 6.5.3 5.3 Node 3 配置 (192.168.1.12)
- 6.6 6. 啟動叢集
 - 6.6.1 6.1 第一步：啟動所有節點的 ADCC for zk
 - 6.6.2 6.2 第二步：啟動所有節點的 Broker
- 6.7 7. 驗證與測試
 - 6.7.1 7.1 驗證 ADCC for zk 叢集狀態
 - 6.7.2 7.2 驗證 Broker 叢集中繼資料
 - 6.7.3 7.3 功能測試
- 6.8 8. 停止服務
- 7 服務化部署 (systemd)
 - 7.1 註冊系統服務
 - 7.2 服務管理命令
- 8 常見問題
 - 8.1 安裝問題
 - 8.2 啟動問題
- 9 回滾方案
 - 9.1 單機回滾
 - 9.2 叢集回滾
 - 9.3 配置回滾
- 10 安全加固
 - 10.1 1. 存取控制
 - 10.1.1 1.1 防火牆配置
 - 10.1.2 1.2 停用不必要的協定
 - 10.2 2. 認證配置
 - 10.2.1 2.1 SASL/SCRAM 認證
 - 10.2.2 2.2 建立安全使用者
 - 10.3 3. 授權配置
 - 10.3.1 3.1 配置 ACL
 - 10.4 4. SSL/TLS 加密
 - 10.4.1 4.1 產生憑證
 - 10.4.2 4.2 配置 SSL
 - 10.5 5. 稽核日誌
 - 10.6 6. 作業系統安全
 - 10.6.1 6.1 檔案權限
 - 10.6.2 6.2 停用 root 執行

• 11 技術支援

1 前言

本文檔為金蝶Apusic分散式訊息佇列for Kafka（Apusic Distributed Message Queue，簡稱：ADMQ for Kafka）的安裝手冊，提供產品各種部署模式（含單機、叢集）在各作業系統、容器環境的安裝部署過程指導以及相關配置說明。

1.1 適用對象

本文檔適用於IT資訊化業務負責人、研發經理、軟體專案經理、軟體架構師、運維工程師。

1.2 相關文檔

瞭解更多ADMQ for Kafka產品相關的資訊，請參閱以下ADMQ for Kafka產品手冊文檔集：

序號	手冊文檔	說明
1	金蝶Apusic分散式訊息佇列for Kafka 快速使用手冊	簡單介紹了如何快速上手使用ADMQ for Kafka。
2	金蝶Apusic分散式訊息佇列for Kafka 安裝手冊	詳細介紹如何在各作業系統上安裝ADMQ for Kafka，以及ADMQ for Kafka服務啟停等操作。
3	金蝶Apusic分散式訊息佇列for Kafka 訊息引擎使用者手冊	詳細介紹 ADMQ for Kafka 訊息引擎相關功能的使用、配置、管理及配套工具的使用方法。
4	金蝶Apusic分散式訊息佇列for Kafka 管控臺使用者手冊	詳細介紹ADMQ for Kafka管控臺相關功能的使用和操作說明。
5	金蝶Apusic分散式訊息佇列for Kafka 開發手冊	詳細介紹基於各開發語言進行ADMQ for Kafka用戶端應用開發的說明。
6	金蝶Apusic分散式訊息佇列for Kafka 遷移手冊	詳細介紹從Kafka遷移到ADMQ for Kafka的說明。
7	金蝶Apusic分散式訊息佇列for Kafka 運維手冊	詳細介紹ADMQ for Kafka的監控、運維、安全加固等運維說明。
8	金蝶Apusic分散式訊息佇列for Kafka 效能最佳化手冊	詳細介紹ADMQ for Kafka效能調優的說明。

1.3 技術支援

ADMQ for Kafka產品提供全面的技術支援服務，您可以透過以下方式獲得技術支援：

- 網址：www.apusic.com
- 電話：400-855-5800
- 信箱：support@apusic.com

- 金蝶雲社群: <https://vip.kingdee.com/?productId=73&productLineId=14&lang=zh-CN>

您在取得技術支援時，請提供如下資訊：

1. 您的姓名
2. 公司資訊與聯絡方式
3. 作業系統及其版本
4. 產品版本號
5. 出現異常及錯誤的日誌、截圖等詳細資訊

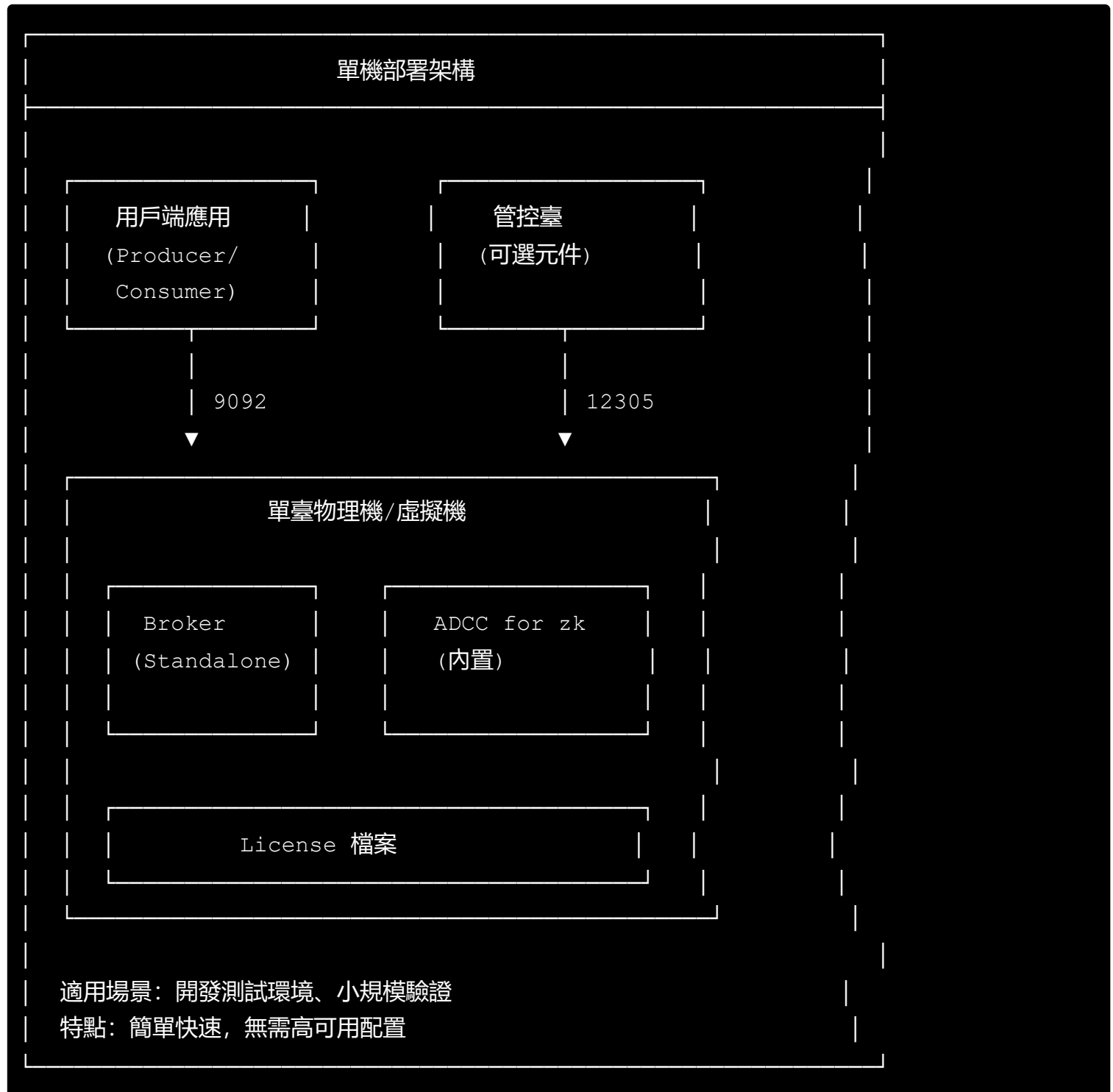
ADMQ for Kafka支援透過管控臺部署，請參考《金蝶Apusic分散式訊息佇列V2.0.6_統一管控臺使用者手冊.pdf》，【快速入門】章節。

2 概述

ADMQ for Kafka包含核心引擎、分散式協調、管控臺三個部分，其中核心引擎受license控制，需要在安裝時申請並匯入license。管控臺為可選部件，如需可視化管控時，可單獨安裝管控臺。

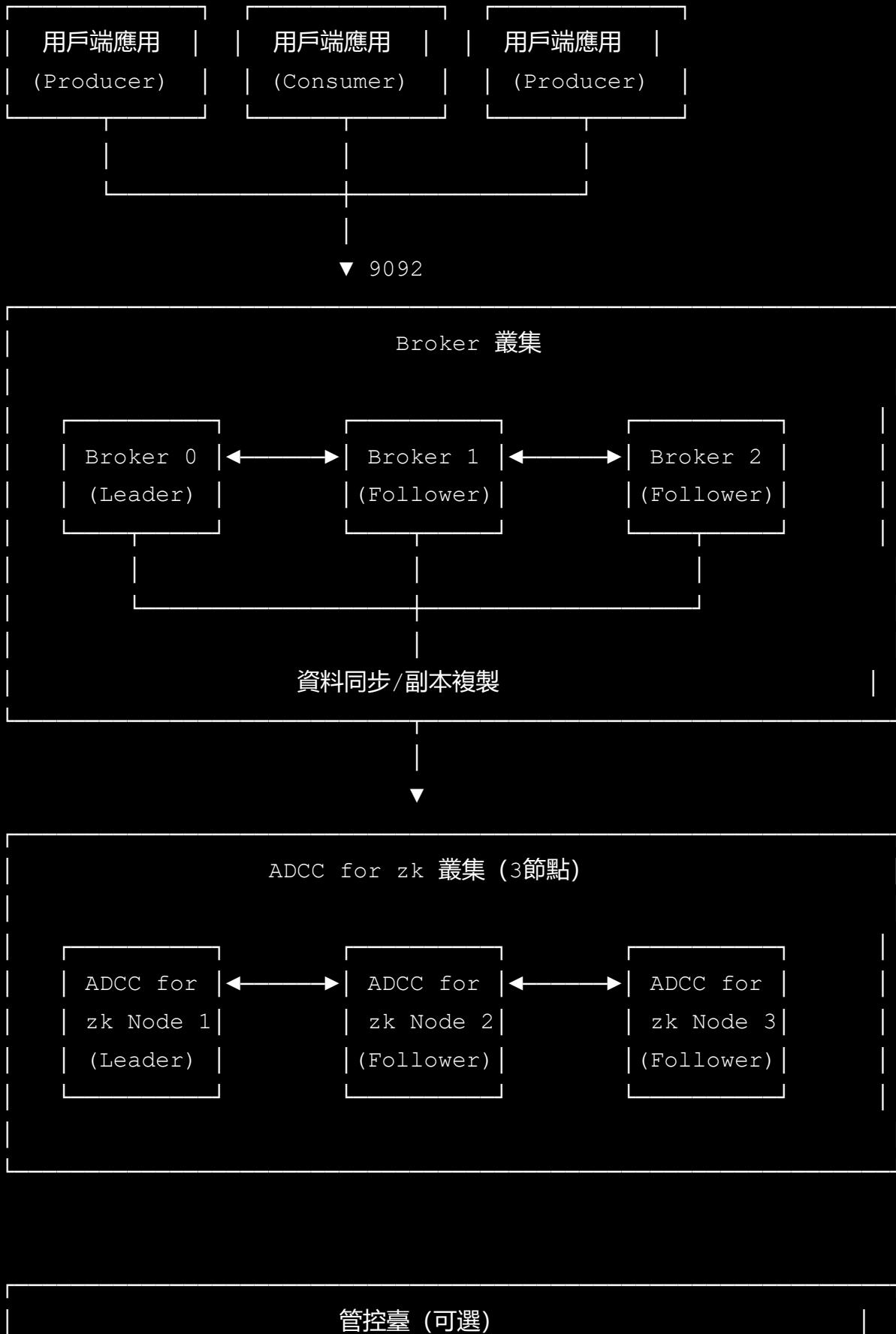
3 部署架構圖

3.1 單機部署架構



3.2 叢集部署架構

叢集部署架構



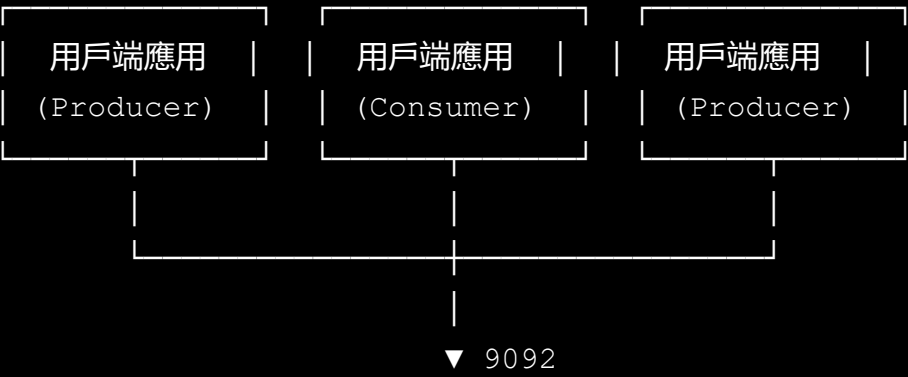
連接埠：12305

適用場景：生產環境、高可用業務場景

特點：多副本冗餘、故障自動轉移、水平擴展

3.3 混合部署架構

混合部署架構



管控臺（可選，獨立部署）

連接埠：12305

適用場景：資源受限的生產環境、中小型叢集

特點：節省伺服器資源，ZK與Broker共用節點，部署成本更低

注意：需確保節點資源充足（CPU/記憶體/磁碟），避免資源競爭

4 安裝前準備

4.1 系統要求

元件	最低要求	建議配置
作業系統	Linux (CentOS 7+, Ubuntu 18.04+, 麒麟 V10, 統信 UOS)	-
CPU	4 核	8 核及以上
記憶體	8 GB	16 GB 及以上
磁碟空間	100 GB	500 GB 及以上 SSD
JDK	JDK 1.8+ (內置 OpenJDK 11)	JDK 11/17

4.2 連接埠檢查

確保以下連接埠可用：

連接埠	服務	說明
9092	Broker	生產者和消費者連線
2181	ADCC for zk	ADCC for zk用戶端連線
2888	ADCC for zk	Leader 和 Follower 通訊
3888	ADCC for zk	選舉通訊
12305	管控臺	Web 介面 (HTTP)

4.3 關閉防火牆（測試環境）

```
# CentOS/RHEL
systemctl stop firewalld
systemctl disable firewalld

# Ubuntu
ufw disable
```

5 單機部署

5.1 1. 建立部署目錄

首先建立應用根目錄並進入：

```
mkdir -p /apusic
cd /apusic
```

5.2 2. 下載並解壓縮軟體包

可從產品光碟或聯絡銷售取得安裝包：

```
# 假設安裝包已複製到伺服器
tar zxvf ADMQ-V2.0.6.391-Kafka-20260319.tar.gz
```

5.3 3. 目錄結構說明

解壓縮完成後，目前目錄下將產生 `adm-q-kafka`。標準目錄結構如下：

目錄名	說明
bin	存放啟動、停止及管理指令稿
config	存放設定檔 (如 <code>kafka-standalone.conf</code>)
data	存放訊息資料及ADCC for zk資料
jdk	內置的 Java 執行環境，無需額外配置系統 JDK
logs	存放執行日誌 (Broker 日誌、ADCC for zk日誌等)
service	系統服務註冊相關檔案 (可選)

5.4 4. 配置修改

編輯單機模式設定檔，綁定指定的 IP 位址和連接埠：

```
vi config/kafka-standalone.conf
```

修改 `listeners` 配置項：

```
listeners=PLAINTEXT://192.168.1.10:9092
```

5.5 5. 授權許可

ADMQ for Kafka 產品需要 License 檔案才能正常執行：

1. **申請授權**：聯絡管理員或透過授權平台申請對應版本的 `license.xml` 檔案
2. **放置檔案**：將取得的 `license.xml` 檔案上傳至部署目錄根路徑

```
# 確認檔案位置
ls -l /apusic/admq-kafka/license.xml
```

5.6 6. 啟動服務

ADMQ for Kafka 單機模式依賴 ADCC for zk，需按順序啟動。

5.6.1 6.1 啟動 ADCC for zk

```
bin/admq-daemon start kafka zk
```

5.6.2 6.2 啟動 Broker

待 ADCC for zk 啟動成功後（通常等待 3-5 秒），啟動 Kafka 主服務：

```
bin/admq-daemon start kafka standalone
```

5.7 7. 服務驗證

5.7.1 7.1 檢查行程狀態

```
ps -ef | grep admq-kafka
```

正常狀態下應能看到兩個主要 Java 行程（一個 ADCC for zk，一個 Broker）。

5.7.2 7.2 檢查連接埠監聽

```
netstat -nltup | grep 9092
```

確認 9092 連接埠處於 LISTEN 狀態。

5.7.3 7.3 功能測試

```
# 建立測試 Topic
kafka/bin/kafka-topics.sh --create \
  --bootstrap-server 192.168.1.10:9092 \
  --topic test-topic \
  --partitions 3 \
  --replication-factor 1

# 生產訊息
kafka/bin/kafka-console-producer.sh \
  --bootstrap-server 192.168.1.10:9092 \
  --topic test-topic

# 消費訊息
kafka/bin/kafka-console-consumer.sh \
  --bootstrap-server 192.168.1.10:9092 \
  --topic test-topic \
  --from-beginning
```

5.8 8. 停止服務

操作	命令
停止 ADCC for zk	bin/admq-daemon stop kafka zk
停止 Broker	bin/admq-daemon stop kafka standalone

6 叢集部署

6.1 1. 部署規劃

假設部署 3 節點高可用叢集（ADCC for zk與 Broker 混合部署）：

節點	IP 位址	ADCC for zk MyID	Broker ID	ADCC for zk 連接埠	Broker 連接埠
Node 1	192.168.1.10	1	0	2181,2888,3888	9092
Node 2	192.168.1.11	2	1	2181,2888,3888	9092
Node 3	192.168.1.12	3	2	2181,2888,3888	9092

前置要求：

1. 確保三台機器時間同步（NTP）
2. 關閉防火牆或開放連接埠：2181, 2888, 3888, 9092

6.2 2. 所有節點執行：基礎安裝

在 Node 1, Node 2, Node 3 上分別執行：

```
mkdir -p /apusic
cd /apusic
tar zxvf ADMQ-V2.0.6.391-Kafka-20260319.tar.gz
cd admq-kafka
```

6.3 3. 放置授權檔案

將 `license.xml` 上傳至 `/apusic/admq-kafka/` 目錄（所有節點）。

6.4 4. 配置 ADCC for zk 叢集

6.4.1 4.1 修改 ADCC for zk 設定檔

編輯 `config/kafka-zk.conf`。在所有三個節點上，配置內容完全一致：

```
server.1=192.168.1.10:2888:3888:participant;2181
server.2=192.168.1.11:2888:3888:participant;2181
server.3=192.168.1.12:2888:3888:participant;2181
```

6.4.2 4.2 建立 MyID 檔案

Node 1 (192.168.1.10):

```
mkdir -p /apusic/admq-kafka/data/zk/data
mkdir -p /apusic/admq-kafka/data/zk/logs
echo "1" > /apusic/admq-kafka/data/zk/data/myid
```

Node 2 (192.168.1.11):

```
mkdir -p /apusic/admq-kafka/data/zk/data
mkdir -p /apusic/admq-kafka/data/zk/logs
echo "2" > /apusic/admq-kafka/data/zk/data/myid
```

Node 3 (192.168.1.12):

```
mkdir -p /apusic/admq-kafka/data/zk/data
mkdir -p /apusic/admq-kafka/data/zk/logs
echo "3" > /apusic/admq-kafka/data/zk/data/myid
```

6.5 5. 配置 Broker 叢集

編輯 `config/kafka-broker.conf` 。

6.5.1 5.1 Node 1 配置 (192.168.1.10)

```
broker.id=0
listeners=PLAINTEXT://192.168.1.10:9092
zookeeper.connect=192.168.1.10:2181,192.168.1.11:2181,192.168.1.12:2181
zookeeper.connection.timeout.ms=18000
```

6.5.2 5.2 Node 2 配置 (192.168.1.11)

```
broker.id=1
listeners=PLAINTEXT://192.168.1.11:9092
zookeeper.connect=192.168.1.10:2181,192.168.1.11:2181,192.168.1.12:2181
```

6.5.3 5.3 Node 3 配置 (192.168.1.12)

```
broker.id=2
listeners=PLAINTEXT://192.168.1.12:9092
zookeeper.connect=192.168.1.10:2181,192.168.1.11:2181,192.168.1.12:2181
```

6.6 6. 啟動叢集

嚴格遵循啟動順序：先 ADCC for zk 叢集，後 Broker 叢集。

6.6.1 6.1 第一步：啟動所有節點的 ADCC for zk

依次登入 Node 1, 2, 3 執行：

```
cd /apusic/admq-kafka
bin/admq-daemon start kafka zk
```

6.6.2 6.2 第二步：啟動所有節點的 Broker

依次登入 Node 1, 2, 3 執行：

```
cd /apusic/admq-kafka
bin/admq-daemon start kafka broker
```

6.7 7. 驗證與測試

6.7.1 7.1 驗證 ADCC for zk 叢集狀態

```
echo ruok | nc 192.168.1.10 2181
echo ruok | nc 192.168.1.11 2181
echo ruok | nc 192.168.1.12 2181
```

預期回傳 `imok`。

6.7.2 7.2 驗證 Broker 叢集中繼資料

```
kafka/bin/kafka-broker-api-versions.sh --bootstrap-server 192.168.1.10:9092
```

6.7.3 7.3 功能測試

1. 建立 Topic (3 副本)：

```
kafka/bin/kafka-topics.sh --create \
  --topic test-cluster \
  --bootstrap-server 192.168.1.10:9092 \
  --partitions 3 \
  --replication-factor 3
```

2. 檢視 Topic 詳情:

```
kafka/bin/kafka-topics.sh --describe \
  --topic test-cluster \
  --bootstrap-server 192.168.1.10:9092
```

3. 生產與消費測試:

```
# 生產者
kafka/bin/kafka-console-producer.sh \
  --topic test-cluster \
  --bootstrap-server 192.168.1.10:9092

# 消費者
kafka/bin/kafka-console-consumer.sh \
  --topic test-cluster \
  --from-beginning \
  --bootstrap-server 192.168.1.10:9092
```

6.8 8. 停止服務

```
# 先停 Kafka
bin/admq-daemon stop kafka broker
# 後停 ADCC for zk
bin/admq-daemon stop kafka zk
```

7 服務化部署 (systemd)

7.1 註冊系統服務

```
# 複製服務檔案
cp service/admq-kafka.service /etc/systemd/system/
cp service/admq-zookeeper.service /etc/systemd/system/

# 重新載入
systemctl daemon-reload

# 啟動服務
systemctl start admq-zookeeper
systemctl start admq-kafka

# 檢視狀態
systemctl status admq-kafka
```

7.2 服務管理命令

操作	命令
啟動 ADCC for zk	systemctl start admq-zookeeper
啟動 Broker	systemctl start admq-kafka
停止	systemctl stop admq-kafka
重新啟動	systemctl restart admq-kafka
狀態	systemctl status admq-kafka
開機自啟	systemctl enable admq-kafka

8 常見問題

8.1 安裝問題

問題	原因	解決方法
連接埠被佔用	9092 或 2181 連接埠已被佔用	檢查連接埠並釋放或修改配置連接埠
記憶體不足	伺服器記憶體低於最低要求	增加伺服器記憶體
磁碟空間不足	磁碟空間不足	清理磁碟或增加儲存
License 無效	License 檔案不正確或過期	申請有效 License 並重新放置

8.2 啟動問題

問題	原因	解決方法
ADCC for zk 啟動失敗	myid 檔案配置錯誤	檢查 myid 檔案內容
Broker 無法連線 ADCC for zk	ADCC for zk 位址配置錯誤	檢查 zookeeper.connect 配置
Broker ID 衝突	broker.id 重複	檢查並修改 broker.id

9 回滾方案

9.1 單機回滾

如需回滾到安裝前狀態：

```
# 1. 停止服務
bin/admq-daemon stop kafka standalone
bin/admq-daemon stop kafka zk

# 2. 備份並刪除資料目錄（謹慎操作）
mv /apusic/admq-kafka /apusic/admq-kafka.bak.$(date +%Y%m%d)

# 3. 如需恢復，重新解壓縮安裝包並配置
```

9.2 叢集回滾

```
# 1. 按順序停止所有節點服務
bin/admq-daemon stop kafka broker
bin/admq-daemon stop kafka zk

# 2. 備份各節點資料
mv /apusic/admq-kafka /apusic/admq-kafka.bak.$(date +%Y%m%d)

# 3. 清理 ADCC for zk 資料
cd /apusic/admq-kafka.bak.*/data/zk
cp -r data /backup/zk-data-$(date +%Y%m%d)

# 4. 重新部署並恢復配置
```

9.3 配置回滾

```
# 備份目前配置
cp config/kafka-broker.conf config/kafka-broker.conf.bak
cp config/kafka-zk.conf config/kafka-zk.conf.bak

# 恢復配置
```

```
cp config/kafka-broker.conf.bak config/kafka-broker.conf
cp config/kafka-zk.conf.bak config/kafka-zk.conf

# 重新啟動服務生效
bin/admq-daemon restart kafka broker
```

10 安全加固

10.1 1. 存取控制

10.1.1 1.1 防火牆配置

```
# 僅開放必要連接埠
iptables -A INPUT -p tcp --dport 9092 -s 192.168.1.0/24 -j ACCEPT
iptables -A INPUT -p tcp --dport 2181 -s 192.168.1.0/24 -j ACCEPT
iptables -A INPUT -p tcp --dport 12305 -s 192.168.1.0/24 -j ACCEPT
iptables -A INPUT -p tcp --dport 9092 -j DROP
iptables -A INPUT -p tcp --dport 2181 -j DROP
```

10.1.2 1.2 停用不必要的協定

```
# 僅啟用安全協定
listeners=SASL_SSL://0.0.0.0:9092
security.inter.broker.protocol=SASL_SSL
```

10.2 2. 認證配置

10.2.1 2.1 SASL/SCRAM 認證

```
# 啟用 SCRAM 認證
sasl.enabled.mechanisms=SCRAM-SHA-256,SCRAM-SHA-512
sasl.mechanism.inter.broker.protocol=SCRAM-SHA-512
```

10.2.2 2.2 建立安全使用者

```
# 建立管理員使用者
kafka-configs.sh --zookeeper localhost:2181 \
  --alter --add-config 'SCRAM-SHA-256=[password=admin-password]' \
  --entity-type users --entity-name admin

# 建立應用使用者
kafka-configs.sh --zookeeper localhost:2181 \
```

```
--alter --add-config 'SCRAM-SHA-256=[password=app-password]' \
--entity-type users --entity-name app-user
```

10.3 3. 授權配置

10.3.1 3.1 配置 ACL

授予生產者權限

```
kafka-acls.sh --authorizer-properties zookeeper.connect=localhost:2181 \
--add --allow-principal User:producer \
--operation Write --topic my-topic
```

授予消費者權限

```
kafka-acls.sh --authorizer-properties zookeeper.connect=localhost:2181 \
--add --allow-principal User:consumer \
--operation Read --topic my-topic \
--group my-consumer-group
```

拒絕非法存取

```
kafka-acls.sh --authorizer-properties zookeeper.connect=localhost:2181 \
--add --deny-principal User:Anonymous \
--operation All --topic '*'
```

10.4 4. SSL/TLS 加密

10.4.1 4.1 產生憑證

產生 CA 憑證

```
keytool -keystore kafka.server.keystore.jks \
--alias ca-cert -import -file ca-cert.crt
```

產生 Broker 憑證

```
keytool -keystore kafka.server.keystore.jks \
--alias broker-cert -validity 365 -genkey -keyalg RSA
```

匯入信任憑證

```
keytool -keystore kafka.server.truststore.jks \
--alias ca-cert -import -file ca-cert.crt
```

10.4.2 4.2 配置 SSL

```
# SSL 配置
listeners=SSL://0.0.0.0:9093
ssl.keystore.location=/path/to/kafka.server.keystore.jks
ssl.keystore.password=keystore-password
ssl.key.password=key-password
ssl.truststore.location=/path/to/kafka.server.truststore.jks
ssl.truststore.password=truststore-password
ssl.client.auth=required
ssl.enabled.protocols=TLSv1.2,TLSv1.3
```

10.5 5. 稽核日誌

```
# 啟用稽核日誌
log4j.logger.kafka.authorizer.logger=WARN, authorizerAppender
log4j.appender.authorizerAppender=org.apache.log4j.DailyRollingFileAppender
log4j.appender.authorizerAppender.File=${log.dir}/kafka-authorizer.log
log4j.appender.authorizerAppender.layout=org.apache.log4j.PatternLayout
log4j.appender.authorizerAppender.layout.ConversionPattern=[%d] %p %m (%c)%n
```

10.6 6. 作業系統安全

10.6.1 6.1 檔案權限

```
# 設定目錄權限
chmod 750 /apusic/admq-kafka
chmod 600 /apusic/admq-kafka/config/*.conf
chmod 600 /apusic/admq-kafka/license.xml

# 限制日誌存取
chmod 640 /apusic/admq-kafka/logs/*.log
```

10.6.2 6.2 停用 root 執行

```
# 建立專用使用者
useradd -r -s /bin/false admq-kafka
```

```
chown -R admq-kafka:admq-kafka /apusic/admq-kafka
```

```
# 使用 systemd 限制权限
```

```
# 在 admq-kafka.service 中新增:
```

```
User=admq-kafka
```

```
Group=admq-kafka
```

```
NoNewPrivileges=true
```

```
ProtectSystem=strict
```

```
ProtectHome=true
```

11 技術支援

如遇到無法解決的問題，請聯絡技術支援：

- 電話：400-855-5800
- 信箱：support@apusic.com

全國統一服務熱線
4008-555-800

金蝶天燕雲計算股份有限公司（簡稱“金蝶天燕”）成立於2000年，前身為“金蝶中間件公司”，是金蝶集團旗下基礎軟件平台提供商，雲計算國家標準制定企業，國家信創產業核心軟件企業。金蝶天燕是國家863重點研發計劃與核高基重大專項承接企業，也是“兩網一站四庫十二金”國家重點工程的基礎平台提供商，產品廣泛應用於政府、軍工、金融、能源等關鍵行業，累計服務客戶總數超過10萬家。



雲計算國家標準制定企業
金蝶集團旗下基礎軟件企業
資訊技術應用創新核心企業
官網：www.apusic.com

