



APUSIC
固若长城
睿比世界

遷移手冊

金蝶Apusic負載均衡器

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 相關手冊
 - 1.3 技術支援
- 2 遷移概述
- 3 支援遷移的版本
- 4 V2.0.5 遷移至V2.0.6
- 5 V2.0.2 敏捷版遷移至V2.0.6
- 6 Nginx遷移至ALB （使用ALB遷移工具）
 - 6.1 功能特性
 - 6.1.1 設定遷移
 - 6.1.2 SSL 憑證遷移
 - 6.1.3 靜態資源遷移
 - 6.1.4 日誌路徑遷移
 - 6.1.5 其他特性
 - 6.2 使用方法
 - 6.3 命令列選項
 - 6.4 使用範例
 - 6.5 遷移後的目錄結構
 - 6.6 注意事項
 - 6.6.1 路徑問題
 - 6.6.2 備份
 - 6.6.3 權限
 - 6.6.4 驗證
- 7 Nginx遷移至ALB （手動遷移）
 - 7.1 簡單直接使用nginx.conf檔案
- 8 常見問題
 - 8.1 遷移後管控台無法登入
 - 8.2 遷移後授權驗證失敗

1 前言

本手冊為金蝶Apusic負載均衡器軟體V2.0.6標準版的遷移手冊，為使用者介紹如何從ALB歷史版本遷移升級到V2.0.6標準版，幫助使用者順利完成版本升級。

1.1 適用對象

本手冊適用於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 相關手冊

了解更多ALB V2.0.6產品相關的資訊，請參閱以下ALB V2.0.6產品手冊文件集：

序號	手冊文件	說明
1	金蝶Apusic負載均衡器軟體 V2.0.6 發版說明	介紹了ALB 產品版本更新內容。
2	金蝶Apusic負載均衡器軟體 V2.0.6 快速使用手冊	簡單介紹了如何快速上手使用ALB。
3	金蝶Apusic負載均衡器軟體 V2.0.6 安裝手冊	詳細介紹如何在各作業系統上安裝ALB，以及ALB服務啟停操作，產品的註冊過程。
4	金蝶Apusic負載均衡器軟體 V2.0.6 代理功能手冊	詳細介紹 ALB 相關功能的使用、設定、管理及配套工具的使用方法。
5	金蝶Apusic負載均衡器軟體 V2.0.6 管控台使用者手冊	詳細介紹ALB管控台相關功能的使用和操作說明。
6	金蝶Apusic負載均衡器軟體 V2.0.6 開發手冊	詳細介紹ALB外部API、alb-cli命令列工具及AI Agent整合方法。
7	金蝶Apusic負載均衡器軟體 V2.0.6 遷移手冊	詳細介紹ALB歷史版本遷移升級到V2.0.6版本的說明。
8	金蝶Apusic負載均衡器軟體 V2.0.6 維運手冊	詳細介紹ALB的監控、維運、安全加固等維運說明。
9	金蝶Apusic負載均衡器軟體 V2.0.6 效能最佳化手冊	詳細介紹ALB效能調校的說明。

1.3 技術支援

ALB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
- 郵箱：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

2 遷移概述

ALB V2.0.6 標準版在 V2.0.5 標準版的基礎上進行了全面升級，主要包括：

- 管控台全新升級，支援更細粒度的三員權限管理和稽核功能。
- 安裝包目錄結構統一為 `ALB-V2.0.6-SE-amd64` / `ALB-V2.0.6-SE-arm64`。
- 控制台二進位統一為 `console/alb-console`，並採用新的使用者資料庫結構。
- 設定檔保持 Nginx 相容，歷史版本的 `alb.conf` 可直接複用。
- 授權檔案相容：`license.lic`、`license.xml` 和統一授權 `acls.properties` 均可繼續使用。

3 支援遷移的版本

來源版本	目標版本	說明
ALB V2.0.5 標準版	ALB V2.0.6 標準版	推薦遷移路徑，設定檔高度相容
ALB V2.0.2 標準版	相容	如果採用的是標準Nginx設定檔，可整體升級至V2.0.6
ALB V2.0.2 敏捷版	ALB V2.0.6 標準版	功能差異較大，建議按新安裝部署後遷移業務設定

4 V2.0.5 遷移至V2.0.6

V2.0.5可以不停機升級至V2.0.6，操作方式如下：

1. 備份現有環境

- 備份舊版本安裝目錄，例如：

```
cp -r /opt/ALB-V2.0.5-SE-amd64 /opt/ALB-V2.0.5-SE-amd64.bak
```

2. 上傳V2.0.6安裝包至/opt目錄並解壓

- 解壓安裝包

```
tar -zxvf ALB-V2.0.6-SE-*.tar.gz -C /opt/
```

3. 替換lua lib目錄

- 本遷移採用原地替換策略：在保留原有安裝目錄結構的前提下，將V2.0.6的新檔案逐步替換到V2.0.5目錄中。替換 `lua lib` 目錄：

```
mv /opt/ALB-V2.0.5-SE-amd64/lualib /opt/ALB-V2.0.5-SE-amd64/lualib.bak
cp -r /opt/ALB-V2.0.6-SE-amd64/lualib /opt/ALB-V2.0.5-SE-amd64/lualib
```

4. 替換alb可執行檔

- 替換 `sbin/alb`：

```
mv opt/ALB-V2.0.5-SE-amd64/sbin/alb opt/ALB-V2.0.5-SE-amd64/sbin/alb.bak
```

```
cp /opt/ALB-V2.0.6-SE-amd64/sbin/alb /opt/ALB-V2.0.5-SE-
amd64/sbin/alb
```

5. 替換啟動腳本

- 替換 bin 目錄啟動腳本：

```
cp /opt/ALB-V2.0.6-SE-amd64/bin/* /opt/ALB-V2.0.5-SE-
amd64/bin/
```

6. 發送重啟訊號

- 發送 USR2 訊號給 alb 主程序，以重新載入設定檔：

```
kill -USR2 $(cat /opt/ALB-V2.0.5-SE-amd64/logs/alb.pid)
# 查看是否存在新舊程序，即存在兩個alb master程序
ps -ef | grep alb | grep -v grep
# 查看logs目錄是否存在alb.pid.oldbin檔案，如果存在則新舊程序均存
在
```

7. 優雅停掉舊worker和master程序

```
# 優雅停止舊worker程序
kill -WINCH $(cat /opt/ALB-V2.0.5-SE-amd64/logs/alb.pid.oldbin)

# 優雅停止舊master程序
kill -QUIT `cat /opt/ALB-V2.0.5-SE-amd64/logs/alb.pid.oldbin`
```

8. 檢查是否只有新程序

- 僅存在一個alb master程序，則說明舊版本已停掉

9. 版本查看

```
./sbin/alb -v  
# 輸出結果為:  
alb version: alb/2.0.6-260615
```

10. 控制台停機升級

```
# 找到控制台程序，並殺掉  
ps aux | grep alb-console | grep -v grep | awk '{print $2}' |  
xargs kill -9  
# 備份舊控制台目錄  
mv /opt/ALB-V2.0.5-SE-amd64/console /opt/ALB-V2.0.5-SE-  
amd64/console.bak  
# 替換控制台目錄  
cp -r /opt/ALB-V2.0.6-SE-amd64/console /opt/ALB-V2.0.5-SE-  
amd64/console  
# 啟動控制台（使用絕對路徑，避免目錄錯誤）  
cd /opt/ALB-V2.0.5-SE-amd64/console && nohup ./alb-console >  
nohup.log 2>&1 &
```

5 V2.0.2 敏捷版遷移至V2.0.6

V2.0.2 敏捷版可以遷移至V2.0.6，操作方式如下：

1. 備份現有環境

- 備份舊版本安裝目錄，例如：

```
cp -r /opt/ALB-V2.0.2-AE-amd64 /opt/ALB-V2.0.2-AE-amd64.bak
```

2. 上傳V2.0.6標準版安裝包至/opt目錄並解壓

- 解壓安裝包（標準版安裝包可用於遷移替換二進位和腳本）

```
tar -zxvf ALB-V2.0.6-SE-*.tar.gz -C /opt/
```

3. 拷貝lualib目錄

- 先備份舊版lualib，再複製新版目錄：

```
mv /opt/ALB-V2.0.2-AE-amd64/lualib /opt/ALB-V2.0.2-AE-amd64/lualib.bak  
cp -r /opt/ALB-V2.0.6-SE-amd64/lualib /opt/ALB-V2.0.2-AE-amd64/lualib
```

4. 替換alb可執行檔

- 替換 `sbin/alb`：

```
mv /opt/ALB-V2.0.2-AE-amd64/sbin/alb /opt/ALB-V2.0.2-AE-amd64/sbin/alb.bak  
cp /opt/ALB-V2.0.6-SE-amd64/sbin/alb /opt/ALB-V2.0.2-AE-amd64/sbin/alb
```

5. 替換啟動腳本

- 替換 `bin` 目錄啟動腳本：

```
cp /opt/ALB-V2.0.6-SE-amd64/bin/* /opt/ALB-V2.0.2-AE-
amd64/bin/
```

6. 發送重啟訊號

- 發送 `USR2` 訊號給 `alb` 主程序，以重新載入設定檔：

```
kill -USR2 $(cat /opt/ALB-V2.0.2-AE-amd64/logs/alb.pid)
# 查看是否存在新舊程序，即存在兩個alb master程序
ps -ef | grep alb | grep -v grep
# 查看logs目錄是否存在alb.pid.oldbin檔案，如果存在則新舊程序均存在
```

7. 優雅停掉舊worker和master程序

```
# 優雅停止舊worker程序
kill -WINCH $(cat /opt/ALB-V2.0.2-AE-amd64/logs/alb.pid.oldbin)

# 優雅停止舊master程序
kill -QUIT `cat /opt/ALB-V2.0.2-AE-amd64/logs/alb.pid.oldbin`
```

8. 檢查是否只有新程序

- 僅存在一個alb master程序，則說明舊版本已停掉

9. 版本查看

```
./sbin/alb -v
# 輸出結果為：
alb version: alb/2.0.6-260615
```

10. 控制台停機升級

找到控制台程序，並殺掉

```
ps aux | grep alb-2.0-light-console | grep -v grep | awk  
'{print $2}' | xargs kill -9
```

備份舊控制台目錄

```
mv /opt/ALB-V2.0.2-AE-amd64/console /opt/ALB-V2.0.2-AE-  
amd64/console.bak
```

替換控制台目錄

```
cp -r /opt/ALB-V2.0.6-SE-amd64/console /opt/ALB-V2.0.2-AE-  
amd64/console
```

啟動控制台（使用絕對路徑，避免目錄錯誤）

```
cd /opt/ALB-V2.0.2-AE-amd64/console && nohup ./alb-console >  
nohup.log 2>&1 &
```

6 Nginx遷移至ALB（使用ALB遷移工具）

ALB 自帶nginx設定檔遷移工具，用於將 nginx 設定遷移為 ALB 設定。

ALB-Nginx遷移工具：`[ALB安裝路徑]/utils/nginx-migrate`

6.1 功能特性

6.1.1 設定遷移

- 掃描 nginx 主設定檔（nginx.conf）
- 解析並遷移所有 include 子設定檔
- 子設定遷移到 `conf/conf.d/` 目錄
- 自動更新主設定的 include 路徑為 `conf.d/*.conf`

6.1.2 SSL 憑證遷移

- 自動掃描 `ssl_certificate`、`ssl_certificate_key`、`ssl_trusted_certificate`
- 複製憑證檔案到 `conf/certs/` 目錄
- 更新設定中的憑證路徑為新路徑

6.1.3 靜態資源遷移

- 自動掃描 `root` 和 `alias` 指令
- 複製靜態資源到 `conf/html/` 目錄
- 更新設定中的靜態資源路徑為絕對路徑

6.1.4 日誌路徑遷移

- 自動遷移 `error_log`、`access_log`、`pid` 路徑
- 統一遷移到 `{ALBPath}/logs/` 目錄

6.1.5 其他特性

- 遷移前自動備份原設定
- 互動式確認（可跳過）
- 詳細日誌模式
- 設定驗證

6.2 使用方法

使用前提：

1. 同一台機器已經部署安裝好ALB。

2. nginx 設定檔所在目錄和 ALB 安裝目錄已存在。

```
# 自動偵測 nginx 設定和 ALB 路徑
./nginx-migrate

# 指定 nginx 設定和 ALB 目錄
./nginx-migrate --nginx-path /path/to/nginx.conf --alb-path
/path/to/ALB
```

6.3 命令列選項

選項	簡寫	說明	預設值
--nginx-path	-	nginx 設定檔路徑	自動偵測
--alb-path	-	ALB 安裝目錄	從工具位置推斷
--from-running	-	從運行中的 nginx 程序獲取設定路徑	false
--dry-run	-	僅掃描，不執行遷移	false
-y / --yes	-	自動確認所有提示	false
-v / --verbose	-	顯示詳細日誌	false
--backup	-	遷移前備份原設定	true

6.4 使用範例

```
# 從運行中的 nginx 程序獲取設定並遷移
./nginx-migrate --from-running --alb-path /path/to/ALB

# 完整模式 (自動確認 + 詳細日誌)
./nginx-migrate --nginx-path /etc/nginx/nginx.conf --alb-path
/opt/ALB -y -v

# 預覽模式 (僅掃描)
./nginx-migrate --nginx-path /etc/nginx/nginx.conf --alb-path
/opt/ALB --dry-run
```

```
# 停用備份
./nginx-migrate --nginx-path /etc/nginx/nginx.conf --alb-path
/opt/ALB --backup=false
```

6.5 遷移後的目錄結構

```
ALB/
├── conf/
│   ├── alb.conf                # 主設定檔
│   ├── certs/                 # SSL 憑證
│   │   ├── server.crt
│   │   ├── server.key
│   │   └── ca.crt
│   ├── conf.d/                # 子設定檔
│   │   ├── default.conf
│   │   ├── ssl.conf
│   │   └── upstream.conf
│   └── html/                  # 靜態資源
│       ├── index.html
│       ├── css/
│       └── js/
└── logs/                      # 日誌目錄
    ├── access.log
    ├── error.log
    └── nginx.pid
```

6.6 注意事項

6.6.1 路徑問題

- 確保 nginx 設定檔中的路徑為相對路徑或絕對路徑
- 相對路徑會基於設定檔所在目錄解析
- 建議在測試環境先使用 `--dry-run` 預覽

6.6.2 備份

- 遷移前會自動備份原設定到 `conf.backup_YYYYMMDD_HHMMSS`
- 如需恢復，刪除新設定並將備份目錄重新命名為 `conf`

6.6.3 權限

- 確保有讀取 nginx 設定和憑證檔案的權限
- 確保有寫入 ALB 目錄的權限

6.6.4 驗證

- 遷移完成後會自動驗證設定語法
- 建議重啟 ALB 前手動檢查設定

7 Nginx遷移至ALB（手動遷移）

7.1 簡單直接使用nginx.conf檔案

1. 備份舊設定

- 備份舊ALB設定，並將nginx.conf替換為alb.conf：

```
cp /opt/ALB-V2.0.6-SE-amd64/conf/alb.conf /opt/ALB-V2.0.6-SE-amd64/conf/alb.conf.bak
cp /path/to/nginx.conf /opt/ALB-V2.0.6-SE-amd64/conf/alb.conf
# 根據實際情況修改來源路徑，將nginx.conf檔案複製到目標路徑
chmod 644 /opt/ALB-V2.0.6-SE-amd64/conf/alb.conf
```

2. 調整include路徑

- 如果原nginx.conf中使用了 `include` 指令引用子設定檔（如 `conf.d/*.conf`），需確保這些路徑在ALB安裝目錄中有效。建議將子設定檔複製到 `/opt/ALB-V2.0.6-SE-amd64/conf/conf.d/` 目錄，並修改include路徑為相對路徑或絕對路徑：

```
mkdir -p /opt/ALB-V2.0.6-SE-amd64/conf/conf.d
cp /path/to/nginx/conf.d/*.conf /opt/ALB-V2.0.6-SE-amd64/conf/conf.d/
```

3. 遷移SSL憑證檔案

- 將原nginx使用的SSL憑證檔案複製到ALB目錄，並更新alb.conf中的憑證路徑：

```
mkdir -p /opt/ALB-V2.0.6-SE-amd64/conf/certs
cp /path/to/ssl/server.crt /opt/ALB-V2.0.6-SE-amd64/conf/certs/
cp /path/to/ssl/server.key /opt/ALB-V2.0.6-SE-amd64/conf/certs/
```

- 編輯alb.conf，將 `ssl_certificate` 和 `ssl_certificate_key` 路徑更新為新的絕對路徑。

4. 調整日誌路徑

- 將alb.conf中的 `error_log`、`access_log` 和 `pid` 路徑修改為ALB日誌目錄：

```
error_log /opt/ALB-V2.0.6-SE-amd64/logs/error.log;  
access_log /opt/ALB-V2.0.6-SE-amd64/logs/access.log;  
pid /opt/ALB-V2.0.6-SE-amd64/logs/alb.pid;
```

5. 驗證設定並啟動ALB

- 使用ALB自帶的設定檢查命令驗證語法：

```
/opt/ALB-V2.0.6-SE-amd64/sbin/alb -t -c /opt/ALB-V2.0.6-SE-  
amd64/conf/alb.conf
```

- 停止Nginx，並啟動ALB：

```
/path/to/sbin/nginx -s stop # 根據實際情況修改Nginx安裝路徑  
/opt/ALB-V2.0.6-SE-amd64/bin/start-alb.sh
```

8 常見問題

8.1 遷移後管控台無法登入

- 原因：V2.0.6 管控台使用新的使用者資料庫，舊版本使用者資料不相容。
- 解決：首次啟動V2.0.6管控台時，系統會自動產生初始隨機密碼並輸出到控制台日誌中。請查看日誌獲取預設使用者帳號和密碼進行登入。若需重置密碼，請透過管控台介面或聯絡系統管理員操作，不要手動修改系統檔案。

8.2 遷移後授權驗證失敗

- 原因：授權檔案未正確複製到新版本安裝目錄，或授權檔案已過期。
- 解決：檢查 `license.lic`、`license.xml` 或 `acls.properties` 是否存在且有效。

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

