



APUSIC
固若长城
睿比世界

迁移手册

金蝶Apusic负载均衡器

版权声明

本文档所涉及的软件著作权、版权等知识产权已依法进行了注册，由金蝶天燕云计算股份有限公司合法拥有。受《中华人民共和国著作权法》《计算机软件保护条例》《知识产权保护条例》和相关国际版权条约、法律、法规以及其它知识产权法律和条约的保护。未经授权许可，不得非法使用。

免责声明

本文档包含的版权信息由金蝶天燕云计算股份有限公司合法拥有，受法律的保护，金蝶天燕云计算股份有限公司对本文档可能涉及到的非金蝶天燕云计算股份有限公司的信息不承担任何责任。在法律允许的范围内，您可以查阅并仅能够在《中华人民共和国著作权法》规定的合法范围内复制和打印本文档。任何单位和个人未经金蝶天燕云计算股份有限公司书面授权许可，不得使用、修改、再发布本文档的任何部分和内容，否则将被视为侵权，金蝶天燕云计算股份有限公司有依法追究其责任的权利。

本文档如有更新，不另行通知。对本文档中的问题您可向金蝶天燕云计算股份有限公司告知或查询。未经本公司明确授予的任何权利均予保留。

商标声明

 是深圳市金蝶天燕云计算股份有限公司向中华人民共和国国家商标局申请注册的注册商标，注册商标专用权由金蝶天燕合法拥有，受法律保护。未经金蝶天燕的书面许可，任何单位及个人不得以任何方式或理由对该商标的任何部分进行使用、复制、修改、传播、抄录或与其它产品捆绑使用销售。凡侵犯金蝶天燕商标权的，金蝶天燕将依法追究其法律责任。本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

目录

- 1 前言
 - 1.1 适用对象
 - 1.2 相关文档
 - 1.3 技术支持
- 2 迁移概述
- 3 支持迁移的版本
- 4 V2.0.5 迁移至V2.0.6
- 5 V2.0.2 敏捷版迁移至V2.0.6
- 6 Nginx迁移至ALB （使用ALB迁移工具）
 - 6.1 功能特性
 - 6.1.1 配置迁移
 - 6.1.2 SSL 证书迁移
 - 6.1.3 静态资源迁移
 - 6.1.4 日志路径迁移
 - 6.1.5 其他特性
 - 6.2 使用方法
 - 6.3 命令行选项
 - 6.4 使用示例
 - 6.5 迁移后的目录结构
 - 6.6 注意事项
 - 6.6.1 路径问题
 - 6.6.2 备份
 - 6.6.3 权限
 - 6.6.4 验证
- 7 Nginx迁移至ALB （手工迁移）
 - 7.1 简单直接使用nginx.conf文件
- 8 常见问题
 - 8.1 迁移后管控台无法登录
 - 8.2 迁移后授权验证失败

1 前言

本文档为金蝶Apusic负载均衡器软件V2.0.6标准版的迁移手册，为用户介绍如何从ALB历史版本迁移升级到V2.0.6标准版，帮助用户顺利完成版本升级。

1.1 适用对象

本文档适用于IT信息化业务负责人、研发经理、软件项目经理、软件架构师、运维工程师。

1.2 相关文档

了解更多ALB V2.0.6产品相关的信息，请参阅以下ALB V2.0.6产品手册文档集：

序号	手册文档	说明
1	金蝶Apusic负载均衡器软件 V2.0.6 发版说明	介绍了ALB 产品版本更新内容。
2	金蝶Apusic负载均衡器软件 V2.0.6 快速使用手册	简单介绍了如何快速上手使用ALB。
3	金蝶Apusic负载均衡器软件 V2.0.6 安装手册	详细介绍如何在各操作系统上安装ALB，以及ALB服务启停操作，产品的注册过程。
4	金蝶Apusic负载均衡器软件 V2.0.6 代理功能手册	详细介绍 ALB 相关功能的使用、配置、管理及配套工具的使用方法。
5	金蝶Apusic负载均衡器软件 V2.0.6 管控台用户手册	详细介绍ALB管控台相关功能的使用和操作说明。
6	金蝶Apusic负载均衡器软件 V2.0.6 开发手册	详细介绍ALB外部API、alb-cli命令行工具及AI Agent集成方法。
7	金蝶Apusic负载均衡器软件 V2.0.6 迁移手册	详细介绍ALB历史版本迁移升级到V2.0.6版本的说明。
8	金蝶Apusic负载均衡器软件 V2.0.6 运维手册	详细介绍ALB的监控、运维、安全加固等运维说明。
9	金蝶Apusic负载均衡器软件 V2.0.6 性能优化手册	详细介绍ALB性能调优的说明。

1.3 技术支持

ALB产品提供全面的技术支持服务，您可以通过以下方式获得技术支持：

- 网址：www.apusic.com
- 电话：400-855-5800
- 邮箱：support@apusic.com
- 金蝶云社区：<https://vip.kingdee.com/?productId=73&productLineId=14&lang=zh-CN>

您在取得技术支持时，请提供如下信息：

1. 您的姓名
2. 公司与联系方式
3. 操作系统及其版本
4. 产品版本号
5. 出现异常及错误的日志、截图等详细信息

2 迁移概述

ALB V2.0.6 标准版在 V2.0.5 标准版的基础上进行了全面升级，主要包括：

- 管控台全新升级，支持更细粒度的三员权限管理和审计功能。
- 安装包目录结构统一为 `ALB-V2.0.6-SE-amd64` / `ALB-V2.0.6-SE-arm64`。
- 控制台二进制统一为 `console/alb-console`，并采用新的用户数据库结构。
- 配置文件保持 Nginx 兼容，历史版本的 `alb.conf` 可直接复用。
- 授权文件兼容：`license.lic`、`license.xml` 和统一授权 `acls.properties` 均可继续使用。

3 支持迁移的版本

源版本	目标版本	说明
ALB V2.0.5 标准版	ALB V2.0.6 标准版	推荐迁移路径，配置文件高度兼容
ALB V2.0.2 标准版	兼容	如果采用的是标准Nginx配置文件，可整体升级至V2.0.6
ALB V2.0.2 敏捷版	ALB V2.0.6 标准版	功能差异较大，建议按新安装部署后迁移业务配置

4 V2.0.5 迁移至V2.0.6

V2.0.5可以不停机升级至V2.0.6，操作方式如下：

1. 备份现有环境

- 备份旧版本安装目录，例如：

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

2. 上传V2.0.6安装包至/opt目录并解压

- 解压安装包

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

3. 替换lua lib目录

- 本迁移采用原地替换策略：在保留原有安装目录结构的前提下，将V2.0.6的新文件逐步替换到V2.0.5目录中。替换 `lua lib` 目录：

```
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. 替换alb可执行文件

- 替换 `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. 替换启动脚本

- 替换 bin 目录启动脚本:

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

6. 发送重启信号

- 发送 `USR2` 信号给 `alb` 主进程, 以重新加载配置文件:

```
kill -USR2 $(cat /opt/ALB-V2.0.5-SE-amd64/logs/alb.pid)
# 查看是否存在新旧进程, 即存在两个alb master进程
ps -ef | grep alb | grep -v grep
# 查看logs目录是否存在alb.pid.oldbin文件, 如果存在则新旧进程均存
在
```

7. 优雅停掉旧worker和master进程

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

# 优雅停止旧master进程
kill -QUIT `cat /opt/ALB-V2.0.5-SE-amd64/logs/alb.pid.oldbin`
```

8. 检查是否只有新进程

- 仅存在一个 `alb master` 进程, 则说明旧版本已停掉

9. 版本查看

```
./sbin/alb -v  
# 输出结果为:  
alb version: alb/2.0.6-260615
```

10. 控制台停机升级

```
# 找到控制台进程，并杀掉  
ps aux | grep alb-console | grep -v grep | awk '{print $2}' |  
xargs kill -9  
# 备份旧控制台目录  
mv /opt/ALB-V2.0.5-SE-amd64/console /opt/ALB-V2.0.5-SE-  
amd64/console.bak  
# 替换控制台目录  
cp -r /opt/ALB-V2.0.6-SE-amd64/console /opt/ALB-V2.0.5-SE-  
amd64/console  
# 启动控制台（使用绝对路径，避免目录错误）  
cd /opt/ALB-V2.0.5-SE-amd64/console && nohup ./alb-console >  
nohup.log 2>&1 &
```

5 V2.0.2 敏捷版迁移至V2.0.6

V2.0.2 敏捷版可以迁移至V2.0.6，操作方式如下：

1. 备份现有环境

- 备份旧版本安装目录，例如：

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

2. 上传V2.0.6标准版安装包至/opt目录并解压

- 解压安装包（标准版安装包可用于迁移替换二进制和脚本）

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

3. 拷贝lualib目录

- 先备份旧版lualib，再复制新版目录：

```
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. 替换alb可执行文件

- 替换 `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. 替换启动脚本

- 替换 `bin` 目录启动脚本:

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

6. 发送重启信号

- 发送 `USR2` 信号给 `alb` 主进程，以重新加载配置文件:

```
kill -USR2 $(cat /opt/ALB-V2.0.2-AE-amd64/logs/alb.pid)
# 查看是否存在新旧进程，即存在两个alb master进程
ps -ef | grep alb | grep -v grep
# 查看logs目录是否存在alb.pid.oldbin文件，如果存在则新旧进程均存在
```

7. 优雅停掉旧worker和master进程

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

# 优雅停止旧master进程
kill -QUIT `cat /opt/ALB-V2.0.2-AE-amd64/logs/alb.pid.oldbin`
```

8. 检查是否只有新进程

- 仅存在一个 `alb master` 进程，则说明旧版本已停掉

9. 版本查看

```
./sbin/alb -v
# 输出结果为:
alb version: alb/2.0.6-260615
```

10. 控制台停机升级

找到控制台进程，并杀掉

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

备份旧控制台目录

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

替换控制台目录

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

启动控制台（使用绝对路径，避免目录错误）

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

6 Nginx迁移至ALB（使用ALB迁移工具）

ALB 自带nginx配置文件迁移工具，用于将 nginx 配置迁移为 ALB 配置。

ALB-Nginx迁移工具： `[ALB安装路径]/utils/nginx-migrate`

6.1 功能特性

6.1.1 配置迁移

- 扫描 nginx 主配置文件（nginx.conf）
- 解析并迁移所有 include 子配置文件
- 子配置迁移到 `conf/conf.d/` 目录
- 自动更新主配置的 include 路径为 `conf.d/*.conf`

6.1.2 SSL 证书迁移

- 自动扫描 `ssl_certificate`、`ssl_certificate_key`、`ssl_trusted_certificate`
- 复制证书文件到 `conf/certs/` 目录
- 更新配置中的证书路径为新路径

6.1.3 静态资源迁移

- 自动扫描 `root` 和 `alias` 指令
- 复制静态资源到 `conf/html/` 目录
- 更新配置中的静态资源路径为绝对路径

6.1.4 日志路径迁移

- 自动迁移 `error_log`、`access_log`、`pid` 路径
- 统一迁移到 `{ALBPath}/logs/` 目录

6.1.5 其他特性

- 迁移前自动备份原配置
- 交互式确认（可跳过）
- 详细日志模式
- 配置验证

6.2 使用方法

使用前提：

1. 同一台机器已经部署安装好ALB。

2. nginx 配置文件所在目录和 ALB 安装目录已存在。

```
# 自动检测 nginx 配置和 ALB 路径
./nginx-migrate

# 指定 nginx 配置和 ALB 目录
./nginx-migrate --nginx-path /path/to/nginx.conf --alb-path
/path/to/ALB
```

6.3 命令行选项

选项	简写	说明	默认值
--nginx-path	-	nginx 配置文件路径	自动检测
--alb-path	-	ALB 安装目录	从工具位置推断
--from-running	-	从运行中的 nginx 进程获取配置路径	false
--dry-run	-	仅扫描，不执行迁移	false
-y / --yes	-	自动确认所有提示	false
-v / --verbose	-	显示详细日志	false
--backup	-	迁移前备份原配置	true

6.4 使用示例

```
# 从运行中的 nginx 进程获取配置并迁移
./nginx-migrate --from-running --alb-path /path/to/ALB

# 完整模式（自动确认 + 详细日志）
./nginx-migrate --nginx-path /etc/nginx/nginx.conf --alb-path
/opt/ALB -y -v

# 预览模式（仅扫描）
./nginx-migrate --nginx-path /etc/nginx/nginx.conf --alb-path
/opt/ALB --dry-run
```

禁用备份

```
./nginx-migrate --nginx-path /etc/nginx/nginx.conf --alb-path
/opt/ALB --backup=false
```

6.5 迁移后的目录结构

```
ALB/
├── conf/
│   ├── alb.conf           # 主配置文件
│   ├── certs/            # SSL 证书
│   │   ├── server.crt
│   │   ├── server.key
│   │   └── ca.crt
│   ├── conf.d/           # 子配置文件
│   │   ├── default.conf
│   │   ├── ssl.conf
│   │   └── upstream.conf
│   └── html/             # 静态资源
│       ├── index.html
│       ├── css/
│       └── js/
└── logs/                 # 日志目录
    ├── access.log
    ├── error.log
    └── nginx.pid
```

6.6 注意事项

6.6.1 路径问题

- 确保 nginx 配置文件中的路径为**相对路径**或**绝对路径**
- 相对路径会基于配置文件所在目录解析
- 建议在测试环境先使用 `--dry-run` 预览

6.6.2 备份

- 迁移前会自动备份原配置到 `conf.backup_YYYYMMDD_HHMMSS`
- 如需恢复，删除新配置并将备份目录重命名为 `conf`

6.6.3 权限

- 确保有读取 nginx 配置和证书文件的权限
- 确保有写入 ALB 目录的权限

6.6.4 验证

- 迁移完成后会自动验证配置语法
- 建议重启 ALB 前手动检查配置

7 Nginx迁移至ALB（手工迁移）

7.1 简单直接使用nginx.conf文件

1. 备份旧配置

- 备份旧ALB配置，并将nginx.conf替换为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
# 根据实际情况修改源路径，将nginx.conf文件复制到目标路径
chmod 644 /opt/ALB-V2.0.6-SE-amd64/conf/alb.conf
```

2. 调整include路径

- 如果原nginx.conf中使用了 `include` 指令引用子配置文件（如 `conf.d/*.conf`），需确保这些路径在ALB安装目录中有效。建议将子配置文件复制到 `/opt/ALB-V2.0.6-SE-amd64/conf/conf.d/` 目录，并修改include路径为相对路径或绝对路径：

```
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. 迁移SSL证书文件

- 将原nginx使用的SSL证书文件复制到ALB目录，并更新alb.conf中的证书路径：

```
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/
```

- 编辑alb.conf，将 `ssl_certificate` 和 `ssl_certificate_key` 路径更新为新的绝对路径。

4. 调整日志路径

- 将alb.conf中的 `error_log`、`access_log` 和 `pid` 路径修改为ALB日志目录：

```
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. 验证配置并启动ALB

- 使用ALB自带的配置检查命令验证语法：

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

- 停止Nginx，并启动ALB：

```
/path/to/sbin/nginx -s stop    # 根据实际情况修改Nginx安装路径  
/opt/ALB-V2.0.6-SE-amd64/bin/start-alb.sh
```

8 常见问题

8.1 迁移后管控台无法登录

- 原因：V2.0.6 管控台使用新的用户数据库，旧版本用户数据不兼容。
- 解决：首次启动V2.0.6管控台时，系统会自动生成初始随机密码并输出到控制台日志中。请查看日志获取默认用户账号和密码进行登录。若需重置密码，请通过管控台界面或联系系统管理员操作，不要手动修改系统文件。

8.2 迁移后授权验证失败

- 原因：授权文件未正确复制到新版本安装目录，或授权文件已过期。
- 解决：检查 `license.lic`、`license.xml` 或 `acls.properties` 是否存在且有效。

全国统一服务热线
4008-555-800



金蝶天燕云计算股份有限公司(简称“金蝶天燕云”)成立于2000年,前身为“金蝶中间件公司”,是金蝶集团旗下新一代软件基础云平台服务商,云计算国家标准制定企业,国家信创产业核心软件企业。金蝶天燕是国家863重点研发计划与核高基重大专项承接企业,也是“两网一站四库十二金”国家重点工程的基础平台提供商,产品广泛应用于政府、军工、金融、能源等关键行业,累计服务客户总数超过10万家。



云计算国家标准制定企业
金蝶集团旗下基础软件企业
信息技术应用创新核心企业
官网: www.apusic.com

