



APUSIC
固若长城
睿比世界

命令列操作指南

金蝶Apusic應用伺服器V10標準版

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 基本介紹
 - 2.1 簡介
 - 2.2 術語
- 3 系統環境要求
 - 3.1 平台環境
 - 3.2 系統環境
- 4 產品清單
- 5 安裝與解除安裝
 - 5.1 Java 環境
 - 5.2 安裝
 - 5.3 解除安裝
- 6 授權許可
 - 6.1 普通授權
 - 6.2 集中授權
 - 6.3 取得特徵碼
 - 6.4 嵌入式部署方式取得特徵碼
- 7 啟動
 - 7.1 在 Linux 環境下
 - 7.2 在 Windows 環境下
- 8 停止
 - 8.1 在 Linux 環境下
 - 8.2 在 Windows 環境下
- 9 命令列工具 aascli
 - 9.1 簡介
 - 9.2 登入
 - 9.2.1 概要
 - 9.2.2 選項
 - 9.2.3 示例

- 9.3 退出登入
 - 9.3.1 概要
 - 9.3.2 示例
- 9.4 查看可用命令
- 9.5 命令通用約定
- 10 系統管理員功能模組
 - 10.1 應用管理
 - 10.1.1 應用管理 (app)
 - 10.1.1.1 概要
 - 10.1.1.2 選項
 - 10.1.2 列出應用 (app list)
 - 10.1.2.1 概要
 - 10.1.2.2 選項
 - 10.1.2.3 示例
 - 10.1.3 查看應用詳情 (app get)
 - 10.1.3.1 概要
 - 10.1.3.2 選項
 - 10.1.3.3 示例
 - 10.1.4 更新應用配置 (app update)
 - 10.1.4.1 概要
 - 10.1.4.2 選項
 - 10.1.4.3 示例
 - 10.1.5 卸載應用 (app delete)
 - 10.1.5.1 概要
 - 10.1.5.2 選項
 - 10.1.5.3 示例
 - 10.1.6 重新載入應用 (app reload)
 - 10.1.6.1 概要
 - 10.1.6.2 示例
 - 10.1.7 部署應用 (app deploy)
 - 10.1.7.1 概要
 - 10.1.7.2 選項
 - 10.1.7.3 示例
 - 10.1.8 重新部署應用 (app redeploy)

- 10.1.8.1 概要
- 10.1.8.2 示例
- 10.1.9 應用類庫管理 (app class-lib)
 - 10.1.9.1 概要
 - 10.1.9.2 選項
 - 10.1.9.3 示例
- 10.1.10 應用增量更新 (app increment)
 - 10.1.10.1 概要
 - 10.1.10.2 選項
 - 10.1.10.3 示例
- 10.1.11 備份應用 (app backup)
 - 10.1.11.1 概要
 - 10.1.11.2 示例
- 10.1.12 應用備份管理 (app-backup)
 - 10.1.12.1 概要
 - 10.1.12.2 示例
- 10.1.13 應用回收管理 (app-recycle)
 - 10.1.13.1 概要
 - 10.1.13.2 示例
- 10.2 配置管理
 - 10.2.1 虛擬主機管理 (host)
 - 10.2.1.1 概要
 - 10.2.1.2 示例
 - 10.2.2 執行緒池管理 (threadpool)
 - 10.2.2.1 概要
 - 10.2.2.2 示例
 - 10.2.3 會話管理器管理 (session-manager)
 - 10.2.3.1 概要
 - 10.2.3.2 示例
 - 10.2.4 HTTP 通道管理 (httpchannel)
 - 10.2.4.1 概要
 - 10.2.4.2 示例
 - 10.2.5 JVM 配置管理 (jvm)
 - 10.2.5.1 概要

- 10.2.5.2 示例
- 10.2.6 JMX 配置管理 (jmx)
 - 10.2.6.1 概要
 - 10.2.6.2 示例
- 10.2.7 證書管理 (cert)
 - 10.2.7.1 概要
 - 10.2.7.2 示例
- 10.3 資源管理
 - 10.3.1 數據源管理 (datasource)
 - 10.3.1.1 概要
 - 10.3.1.2 示例
- 10.4 監控管理
 - 10.4.1 監控配置管理 (monitor-config)
 - 10.4.1.1 概要
 - 10.4.1.2 示例
 - 10.4.2 告警配置管理 (alert-config)
 - 10.4.2.1 概要
 - 10.4.2.2 示例
 - 10.4.3 SNMP 代理配置管理 (snmp-config)
 - 10.4.3.1 概要
 - 10.4.3.2 示例
 - 10.4.4 一鍵巡檢管理 (inspection)
 - 10.4.4.1 概要
 - 10.4.4.2 示例
 - 10.4.5 監控快照管理 (snapshot)
 - 10.4.5.1 概要
 - 10.4.5.2 示例
- 10.5 日誌服務
 - 10.5.1 系統日誌配置 (log-config system)
 - 10.5.1.1 概要
 - 10.5.1.2 示例
 - 10.5.2 存取日誌配置 (log-config access)
 - 10.5.2.1 概要
 - 10.5.2.2 示例

- 10.6 用戶管理
 - 10.6.1 用戶管理 (user)
 - 10.6.1.1 概要
 - 10.6.1.2 示例
- 10.7 安全服務
 - 10.7.1 安全域管理 (security-domain)
 - 10.7.1.1 概要
 - 10.7.1.2 示例
- 10.8 系統與賬戶
 - 10.8.1 用戶設定 (usersettings)
 - 10.8.1.1 概要
 - 10.8.1.2 選項
 - 10.8.1.3 示例
 - 10.8.2 修改當前登入用戶的密碼 (changepassword)
 - 10.8.2.1 概要
 - 10.8.2.2 選項
 - 10.8.2.3 示例
- 11 安全管理員功能模組
 - 11.1 用戶管理
 - 11.1.1 用戶管理 (user)
 - 11.1.1.1 概要
 - 11.1.2 列出用戶 (user list)
 - 11.1.2.1 示例
 - 11.1.3 查看用戶詳情 (user get)
 - 11.1.3.1 示例
 - 11.1.4 重置用戶密碼 (user reset-password)
 - 11.1.4.1 概要
 - 11.1.4.2 選項
 - 11.1.4.3 示例
 - 11.1.5 刪除用戶 (user delete)
 - 11.1.5.1 示例
 - 11.1.6 用戶權限配置 (user permission)
 - 11.1.6.1 概要
 - 11.1.6.2 選項

- 11.1.6.3 示例
- 11.2 角色管理
 - 11.2.1 角色管理 (role)
 - 11.2.1.1 概要
 - 11.2.2 列出角色 (role list)
 - 11.2.2.1 概要
 - 11.2.2.2 示例
 - 11.2.3 查看角色詳情 (role get)
 - 11.2.3.1 概要
 - 11.2.3.2 示例
 - 11.2.4 列出可用權限資源 (role permissions)
 - 11.2.4.1 概要
 - 11.2.4.2 示例
 - 11.2.5 創建角色 (role add)
 - 11.2.5.1 概要
 - 11.2.5.2 選項
 - 11.2.5.3 示例
 - 11.2.6 更新角色 (role update)
 - 11.2.6.1 概要
 - 11.2.6.2 示例
 - 11.2.7 刪除角色 (role delete)
 - 11.2.7.1 概要
 - 11.2.7.2 示例
- 11.3 安全配置管理
 - 11.3.1 概要
 - 11.3.2 密碼策略配置 (security-config password-policy)
 - 11.3.2.1 概要
 - 11.3.2.2 示例
 - 11.3.3 審計日誌配置 (security-config audit-log)
 - 11.3.3.1 概要
 - 11.3.3.2 示例
 - 11.3.4 郵件驗證配置 (security-config mail)
 - 11.3.4.1 概要
 - 11.3.4.2 示例

- 11.3.5 動態碼驗證配置 (security-config otp)
 - 11.3.5.1 概要
 - 11.3.5.2 示例
- 11.3.6 業務配置備份管理 (security-config backup)
 - 11.3.6.1 概要
 - 11.3.6.2 示例

1 前言

本文檔介紹金蝶Apusic應用伺服器（AAS）V10標準版的命令列操作指引，涵蓋系統環境要求、安裝方法，以及透過 aascli 命令列工具進行的日常管理與維運操作，適用於使用 AAS 進行開發的開發人員、生產環境的系統管理員、維運人員等。

1.1 適用對象

本手冊主要面向使用金蝶Apusic應用伺服器進行應用開發的開發人員，生產環境的系統管理員，應用發佈人員，技術維運工程師等。具備以下技能會更有助於理解和使用本手冊內容：

- 熟悉 Linux 常用命令
- 基本的系統管理任務
- 安裝和管理軟體

1.2 相關文檔

了解更多 AAS-V10 標準版產品相關的資訊，請參閱以下 AAS-V10 產品手冊文檔集：

序號	手冊文檔	說明
1	金蝶Apusic應用伺服器V10標準版安裝手冊	詳細介紹如何在各作業系統上安裝AAS，以及AAS服務啟停操作，產品的註冊過程
2	金蝶Apusic應用伺服器V10標準版快速入門	簡單介紹瞭如何快速上手使用AAS
3	金蝶Apusic應用伺服器V10標準版用戶手冊	詳細介紹AAS相關功能的使用和操作說明，指導用戶對系統進行安裝部署、功能操作、參數配置、日常運維及問題自查
4	金蝶Apusic應用伺服器V10標準版管控台用戶手冊	詳細介紹AAS管控台相關功能的使用和操作說明，指導用戶通過可視化管控台進行操作
5	金蝶Apusic應用伺服器V10標準版命令列操作指南	詳細介紹AAS通過命令列進行操作使用
6	金蝶Apusic應用伺服器V10標準版監控指標文件	詳細介紹AAS各類型監控指標
7	金蝶Apusic應用伺服器V10標準版API文件	詳細介紹AAS的API資訊

1.3 技術支援

金蝶Apusic應用伺服器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
- 金蝶雲社區：<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. 公司資訊與聯繫方式
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 基本介紹

2.1 簡介

金蝶Apusic應用伺服器（AAS）V10標準版是一款符合 JAVAEE 規範的 Web 應用伺服器，屬於輕量級應用伺服器，通常用於應用系統高效開發場合。

2.2 術語

AAS：金蝶Apusic應用伺服器

AE：金蝶Apusic應用伺服器標準版產品包標識

APUSIC_HOME：金蝶Apusic應用伺服器安裝目錄

3 系統環境要求

3.1 平台環境

金蝶Apusic應用伺服器相容各類型作業系統，如下表所示。

表1：金蝶Apusic應用伺服器V10標準版適應的作業系統

作業系統	說明
Windows	Windows 7、Windows 10、Windows Server
Linux	CentOS、Ubuntu、SUSE、Redhat
國產平台	銀河麒麟系列、中標麒麟系列、普華、中科紅旗、深度等

3.2 系統環境

金蝶Apusic應用伺服器系統環境建議如下。

表2：金蝶Apusic應用伺服器V10標準版系統環境

系統組件	系統要求
Java 環境	JDK8 及以上，針對 HTTP2 建議使用 8U252 以上
記憶體	至少需要 512MB 的記憶體，建議 1024MB 以上
磁碟空間	至少需要 1024MB 的磁碟空間，建議 10G 以上

4 產品清單

產品安裝包通常打包為 `AAMS-[version].tar.gz` 或 `AAMS-[version].zip`。

文中 `${APUSIC_HOME}` 表示應用伺服器的安裝目錄。

表3：金蝶Apusic應用伺服器V10標準版產品包

產品包	說明
AAMS-[version].zip / AAMS-[version].tar.gz	金蝶Apusic應用伺服器V10標準版安裝包

金蝶Apusic應用伺服器V10標準版產品包說明。

表4：金蝶Apusic應用伺服器V10標準版產品包說明

檔案	說明
bin	應用伺服器啟動、停止等腳本所在的目錄
conf	應用伺服器配置檔所在的目錄
console	應用伺服器管控制台所在的目錄
lib	應用伺服器 JAR 檔案所在的目錄
logs	應用伺服器日誌檔案預設目錄
plugins	應用伺服器外掛程式所在的目錄
store	資料倉儲目錄，各類資料的集中存放地
webapps	自動部署應用程式包的目錄
webapps-console	透過管控制台部署應用程式包的目錄
webapps-jakarta	部署 JakartaEE 規範應用程式包的目錄
temp	應用伺服器臨時檔案存放的目錄
work	存放 JSP 編譯後產生的 class 檔案
license.xml	產品包的授權檔案

5 安裝與解除安裝

5.1 Java 環境

安裝前，需要先確認 Java 環境。JDK 版本建議在 1.8.0_201 及以上，查看 JDK 版本命令如下：

```
[user@test bin]# java -version
java version "1.8.0_212"
Java(TM) SE Runtime Environment (build 1.8.0_212-b10)
Java HotSpot(TM) 64-Bit Server VM (build 25.212-b10, mixed mode)
```

支援指定 Java 執行環境，透過在 `apusic.sh` 或 `apusic.bat` 設定，在開頭新增 `JAVA_HOME`，如：

```
set JAVA_HOME="/opt/java/jdk1.8.0_271"
```

5.2 安裝

準備安裝包 `AAMS-V10.0.8-SP12-AE-20260701.tar.gz` 或 `AAMS-V10.0.8-SP12-AE-20260701.zip`，拷貝至有權限的目錄，進行解壓縮。

解壓縮即安裝成功。

```
執行 tar -zxvf AAMS-V10.0.8-SP12-AE-20260701.tar.gz
或 unzip AAMS-V10.0.8-SP12-AE-20260701.zip
```

5.3 解除安裝

刪除安裝目錄 `${APUSIC_HOME}/` 下所有檔案即可完成解除安裝操作。解除安裝將會遺失當前 AAS 所有已配置的檔案，請謹慎操作。

6 授權許可

AAS 需要有對應的許可證才能正常使用，通常情況下，金蝶天燕會根據用戶購買的產品版本配套對應的許可證，如果在使用過程中出現許可證過期或無效等問題，建議優先聯繫對接的天燕服務人員，重新申請對應許可證。

產品授權方式分為普通授權和集中授權。

6.1 普通授權

普通授權指根據 IP、域名等方式生成 `license.xml` 檔案，將授權檔案放置安裝目錄下，

`${APUSIC_HOME}/license.xml`。

如若安裝有管控平台，系統管理員 admin 登入管控平台，點擊右上角的授權圖標，點擊「更新授權」，上傳可用的授權檔案，儲存即可完成更新。重啟 AAS 後新授權檔案生效。

6.2 集中授權

集中授權指連接授權中心，進行統一授權。需要先搭建金蝶Apusic授權中心，操作方式可參考《金蝶Apusic許可授權中心用戶手冊》，或聯繫金蝶天燕技術支援人員。

在系統環境中配置環境變數，或在 AAS 安裝目錄根目錄建立 `acls.properties` 檔案，新增以下參數：

```
apusic_acls_enable=true
apusic_acls_authUrls=127.0.0.1:6886
apusic_acls_ns=apusic
apusic_acls_tenant=ApusicTest
```

連接參數說明：

表5：金蝶Apusic許可授權中心連接參數說明

參數名	參數值說明
apusic_acls_enable	是否開啟授權中心認證，取值為 true 或 false，為 true 則表示開啟授權中心認證。沒有該參數或該參數值為 false，都表示沒有開啟授權中心認證；
apusic_acls_authUrls	授權中心的位址，可設定多個授權位址，格式為 ip1:port1,ip2:port2，如果一個授權位址連接失敗，會輪詢其他的位址；如果開啟授權中心認證，則為必填參數，其中埠為授權中心的 https 埠；
apusic_acls_ns	設定該實例所屬的命名空間名稱，可選參數；預設值為 public，具體的命名空間可以在

	授權中心管理控制台-系統管理-授權管理查看。
apusic_acls_tenant	設定該實例所屬的租戶名稱，可選參數。

AAS 啟動時將會自動連接到 Apusic 授權中心。

6.3 取得特徵碼

如果在使用過程中出現許可證過期或無效等問題，建議優先聯繫對接的天燕服務人員，重新申請對應許可證。重新申請對應許可證時，需要將產品的特徵碼（auth code）提供到天燕對接人員。

在 `${APUSIC_HOME}/bin`，執行 `apusic.sh -ac [ethname or ip]`，`[ethname or ip]` 取值為 IP 位址或者網卡名稱，類似如下：

```
apusic.sh -ac 127.0.0.1
```

列印特徵碼資訊，類似如下，Auth Code=特徵碼內容：

```
Using APUSIC_BASE:      /opt/AAS/AAMS-V10.0.0
Using APUSIC_HOME:      /opt/AAS/AAMS-V10.0.0
Using APUSIC_TMPDIR:    /opt/AAS/AAMS-V10.0.0/temp
Using JRE_HOME:         /usr/local/java/jdk1.8.0_231/jre
Using CLASSPATH:        /opt/AAS/AAMS-
V10.0.0/bin/bootstrap.jar:/opt/AAS/AAMS-V10.0.0/bin/aas-juli.jar
Using APUSIC_OPTS:
Auth Code=SZTY829390371
```

取得特徵碼後再提供特徵碼申請授權檔案。

6.4 嵌入式部署方式取得特徵碼

在與 springboot 整合的方式下，沒有取得特徵碼的 `apusic` 命令，需要依賴用戶的應用進行取得。

如果已經有臨時的授權檔案可以啟動應用，則先把授權檔案移除，然後在啟動命令中增加 JVM 參數 `apusic-ac`，取值為 IP 位址或者網卡名稱，類似如下，eth3 為測試機器的網卡名稱，需要根據實際的網路資訊修改：

```
java -jar -Dapusic-ac=eth3 boot-quickstart-1.0.jar
```

由於沒有授權檔案，應用啟動會停止，在停止前會列印特徵碼資訊，類似如下，Auth Code=特徵碼內容：

```
Auth Code=SZTY263537626  
license is invalid.
```

7 啟動

7.1 在 Linux 環境下

安裝完成並更新授權檔案後，進入安裝目錄 `${APUSIC_HOME}/bin`，執行 `apusic.sh run`。

也可執行 `./startapusic.sh` 啟動服務。

7.2 在 Windows 環境下

安裝完成並更新授權檔案後，進入安裝目錄 `${APUSIC_HOME}\bin`，執行 `apusic.bat run`。

也可執行 `startapusic.bat` 啟動服務。

8 停止

8.1 在 Linux 環境下

1. Ctrl + C

前端啟動狀態下，可以直接 Ctrl + C 退出，停止 AAS。

2. stopapusic.sh

進入安裝目錄 `${APUSIC_HOME}/bin`，執行 `stopapusic.sh`。

3. kill 程序

可以透過 kill [程序號] 方式停止。

8.2 在 Windows 環境下

1. Ctrl + C

前端啟動狀態下，可以直接 Ctrl + C 退出，停止 AAS。

2. stopapusic.bat

進入安裝目錄 `${APUSIC_HOME}\bin`，執行 `stopapusic.bat`。

3. 停止程序

可以開啟工作管理員，找到對應程序，停止執行。

9 命令列工具 aascli

9.1 簡介

AAS 提供命令列工具 `aascli`，用於對應用伺服器進行日常管理與維運操作，功能與管控台保持一致。命令列工具位於 `${APUSIC_HOME}/bin` 目錄下：

- Linux 環境：`aascli.sh`
- Windows 環境：`aascli.bat`

使用 `aascli` 前，需確認：

- 已配置 `JAVA_HOME` 或 `JRE_HOME` 環境變數（JDK8 及以上）；
- AAS 服務已啟動（`aascli` 透過管控台埠與服務端通訊，預設埠 6888）；
- 擁有可登入管控台的帳號。

`aascli` 支援 `en`、`zh`、`zh_TW` 三種介面語言，透過 `usersettings --lang=<語言>` 切換，重新登入 `aascli` 後生效。本文檔示例輸出均以 `zh_TW` 語言擷取。

9.2 登入

`login` 命令用於登入到 AAS 控制台，登入成功後進入互動式命令空間。系統內建三個管理員帳號，對應「三員」分立的管理模型：

用戶名	角色	說明
admin	sysadmin	系統管理員，負責應用、配置、資源、監控、日誌等系統維運管理
secure	security	安全管理員，負責角色、安全配置等安全管理（本文檔有描述）
audit	auditor	審計管理員，負責審計日誌管理（本文檔暫不描述）

9.2.1 概要

```
Usage: aascli login [-h] [--no-interactive] [--file=<loginFile>]
                  [--generate-file=<generateFile>] [-p=<password>]
```

```
[-u=<username>]
```

登入到 AAS 控制台

```
--file=<loginFile>
```

登入檔案路徑 (檔案內含加密的登入資訊，

由 `--generate-file` 生成)

<code>--generate-file=<generateFile></code>	校驗帳號密碼後生成包含加密登入資訊的登入檔案並退出, 不進入互動式命令空間
<code>-h, --help</code>	顯示說明資訊
<code>--no-interactive</code>	登入成功後不進入互動式命令空間, 直接退出 (用於腳本自動化)
<code>-p, --password=<password></code>	密碼
<code>-u, --username=<username></code>	用戶名

9.2.2 選項

```
-u, --username=<username>
```

登入用戶名。系統管理員預設為 admin。未提供時互動式輸入。

```
-p, --password=<password>
```

登入密碼。未提供時互動式輸入（輸入不回顯）。腳本自動化場景可直接指定。

```
--no-interactive
```

登入成功後不進入互動式命令空間，直接退出，用於腳本自動化中校驗帳號可用性。

```
--generate-file=<generateFile>
```

校驗帳號密碼後生成包含加密登入資訊的登入檔案並退出。生成的登入檔案可配合 `--file` 使用，避免在腳本中明文儲存密碼。

```
--file=<loginFile>
```

使用登入檔案登入，檔案內含加密的登入資訊，由 `--generate-file` 生成。

9.2.3 示例

示例1 以系統管理員身份登入

進入 `${APUSIC_HOME}/bin` 目錄執行 `aascli`，在互動式提示中輸入用戶名、密碼（密碼不回顯）：

```
[user@test bin]$ ./aascli.sh
aas> login
```

```
用戶名: admin
密碼:
admin登錄成功
admin@localhost>
```

也可以在登入命令中直接指定用戶名密碼：

```
[user@test bin]$ ./aascli.sh login -u admin -p [YOURPASSWORD]
admin登錄成功
admin@localhost>
```

登入成功後進入互動式命令空間，提示符為 `用戶名@主機名>`，即可輸入各功能子命令。

示例2 密碼錯誤時的提示

密碼連續錯誤會鎖定帳號，請謹慎操作：

```
登錄失敗：用戶名或密碼錯誤，剩余重试次数 4 次
```

示例3 生成登入檔案並用於腳本自動化

先生成加密登入檔案，再使用登入檔案登入，避免腳本中出現明文密碼：

```
[user@test bin]$ ./aascli.sh login -u admin -p [YOURPASSWORD] --
generate-file=/home/user/aas.login
登錄文件已生成: /home/user/aas.login

[user@test bin]$ ./aascli.sh login --file=/home/user/aas.login
admin登錄成功
admin@localhost>
```

示例4 校驗帳號可用性後直接退出

```
[user@test bin]$ ./aascli.sh login -u admin -p [YOURPASSWORD] --no-
interactive
admin登錄成功
```

9.3 退出登入

`logout` 命令用於退出當前登入，退出互動式命令空間可直接使用 `exit` 或 `Ctrl + D`。

9.3.1 概要

```
Usage: logout [-h]
```

退出登錄

`-h, --help` 顯示說明資訊

9.3.2 示例

示例1 退出登入

```
admin@localhost> logout
```

9.4 查看可用命令

`help` 命令用於顯示當前登入用戶可用的命令列表。命令列表與登入用戶的角色權限相關，系統管理員登入後可見的命令如下：

```
admin@localhost> help
```

應用管理：

<code>app</code>	应用列表
<code>app-backup</code>	应用备份
<code>app-recycle</code>	应用回收

配置管理：

<code>host</code>	虛擬主機管理
<code>threadpool</code>	线程池管理
<code>session-manager</code>	会话管理器管理
<code>httpchannel</code>	HTTP 通道管理
<code>jvm</code>	JVM 配置管理
<code>jmx</code>	JMX 配置管理
<code>cert</code>	证书管理

資源管理：

<code>datasource</code>	数据源管理
-------------------------	-------

監控管理：

```
monitor-config  監控配置管理
alert-config    告警配置管理
snmp-config     SNMP 代理配置管理
inspection      一键巡检管理
snapshot        监控快照管理（手动创建、下载、删除、自动配置）
```

日誌服務：

```
log-config  日志配置管理
```

用戶管理：

```
user        用户管理
```

安全服務：

```
security-domain 安全域管理
```

系統與賬戶：

```
usersettings  用户设置
changepassword 修改当前登录用户的密码
logout        退出登录
help          显示可用命令
```

不同角色登入後可見的命令集合不同：系統管理員的功能模組參見「系統管理員功能模組」章節，安全管理員的功能模組參見「安全管理員功能模組」章節；審計管理員的功能模組本文檔暫不描述。

9.5 命令通用約定

- 命令結構：**命令採用 **主命令** [子命令] [參數] [選項] 的結構，如 `app list --page=2`。輸入 **主命令** `--help` 或 **主命令** **子命令** `--help` 可查看對應階層的使用說明。
- 說明選項：**所有命令均支援 `-h` 或 `--help` 顯示說明資訊。
- 確認提示：**刪除類操作預設會提示確認，輸入 `y` 確認執行，其他輸入取消；指定 `-f` 或 `--force` 則不再提示確認，適用於腳本自動化場景。
- 輸出格式：**列表查詢類命令支援 `--output=table|json` 設定輸出格式，預設 `table`；`--page=<頁碼>` 指定顯示第幾頁；`--no-paging` 關閉分頁、一次輸出全部記錄。詳情查詢類命令支援 `--output=json` 以 `json` 格式輸出。
- 生效說明：**部分配置類操作提示「重啟服務器後生效」，需重啟 AAS 後新配置才生效。

10 系統管理員功能模組

10.1 應用管理

10.1.1 應用管理 (app)

`app` 子命令用於應用管理，包括應用的部署、卸載、更新、備份、類庫與增量包管理等，通常需要帶具體的請求操作執行。

10.1.1.1 概要

```
Usage:  app [-h] [COMMAND]
```

應用管理

`-h, --help` 顯示說明資訊

Commands:

<code>list</code>	列出應用
<code>get</code>	查看應用詳情
<code>update</code>	更新應用配置
<code>delete</code>	卸載應用
<code>reload</code>	重新載入應用
<code>deploy</code>	部署應用
<code>redeploy</code>	重新部署應用
<code>class-lib</code>	應用類庫管理
<code>increment</code>	應用增量更新
<code>backup</code>	備份應用

10.1.1.2 選項

`list` 列出應用； `get` 查看應用詳情； `update` 更新應用配置； `delete` 卸載應用； `reload` 重新載入應用； `deploy` 部署應用； `redeploy` 重新部署應用； `class-lib` 應用類庫管理； `increment` 應用增量更新； `backup` 備份應用。

10.1.2 列出應用 (app list)

`app list` 子命令用於列出當前已部署的應用，展示應用名稱、上下文路徑、虛擬主機等資訊。

10.1.2.1 概要

```
Usage: app list [--page=<page>] [--output=<format>] [--no-paging] [-h]
```

列出應用

-h, --help	顯示說明資訊
--page=<page>	頁碼
--no-paging	本次命令關閉分頁輸出
--output=<output>	輸出格式: table 或 json

10.1.2.2 選項

```
--page=<page>
```

設定頁碼，展示第幾頁的資訊。

```
--no-paging
```

在當前子命令中停用分頁。

```
--output=<output>
```

設定輸出的格式，可設定為 table 或 json。table 表示輸出的格式為表格，為預設值；json 表示輸出的格式為 json。

10.1.2.3 示例

示例1 列出當前已部署的應用

```
admin@localhost> app list
| 應用名稱 | 上下文路徑 | 虛擬主機 | 部署順序 | 部署時間 | 密級 |
+-----+-----+-----+-----+-----+-----+
---+
| testapp | /testapp | localhost | 100 | 2026-09-27 17:36 | 秘密 |
| ROOT | / | localhost | 100 | 2026-09-27 14:13 | 秘密 |
第 1 / 1 頁, 共 2 條
```

示例2 以 json 格式列出當前已部署的應用

```
admin@localhost> app list --output=json
{
```

```

"pagination" : {
  "total" : 1,
  "pages" : 1,
  "page" : 1
},
"data" : [ {
  "id" : 1,
  "appName" : "ROOT",
  "contextPath" : "/",
  "status" : 1,
  "classLoaderOrder" : null,
  "jspPrecompile" : false,
  "virtualHost" : "localhost",
  "sessionManager" : null,
  "deployOrder" : 100,
  "sharedLib" : null,
  "sharedLibId" : null,
  "deployTimeout" : 300,
  "classLevel" : "秘密",
  "description" : null,
  "deployTime" : "2026-09-27 14:13",
  "recycledTime" : null,
  "securityDomain" : null,
  "deployProps" : null,
  "managedByConsole" : true,
  "docBase" : "D:\\AAMS-V10.0.8\\webapps-console\\ROOT",
  "deploySource" : "CONSOLE"
} ]
}

```

10.1.3 查看應用詳情 (app get)

`app get` 子命令用於查看指定應用的配置詳情。

10.1.3.1 概要

```
Usage: app get <NAME> [--virtual-host=<host>] [--output=<format>] [-h]
```

查看應用詳情

<code><name></code>	應用名稱
<code>--virtual-host=<virtualHost></code>	虛擬主機名稱，可透過 <code>aascli host</code>
<code>list</code>	查詢；存在多個同名應用時必填
<code>--output=<output></code>	輸出格式：json
<code>--help</code>	顯示說明資訊

10.1.3.2 選項

```
<name>
```

應用名稱，可透過 `app list` 查詢。

```
--virtual-host=<virtualHost>
```

虛擬主機名稱。當不同虛擬主機下存在多個同名應用時必填，用於定位具體應用。

```
--output=<output>
```

輸出格式：json。不指定時以格式化文字展示。

10.1.3.3 示例

示例1 查看應用詳情

```
admin@localhost> app get ROOT
應用名稱：ROOT
上下文路徑：/
類載入順序：
JSP 預編譯：否
虛擬主機：localhost
部署順序：100
安全域：-
密級：秘密
應用描述：-
```

10.1.4 更新應用配置 (app update)

`app update` 子命令用於更新已部署應用的配置項，可選擇更新後立即重新載入應用使配置生效。

10.1.4.1 概要

Usage: `app update <NAME> [--virtual-host=<host>] [選項] [--reload]`

更新應用配置

<code><name></code>	應用名稱
<code>--virtual-host=<virtualHost></code>	虛擬主機名稱，可透過
<code>aascli host list</code> 查詢；存在多個同名應用時必填	
<code>--reload</code>	更新成功後立即重新載入應用，使配置生效
<code>--context-path=<contextPath></code>	存取應用的上下文路徑，預設
為應用名稱，同一虛擬主機內不可重複	
<code>--class-loader-order=<classLoaderOrder></code> (父類優先) 或 <code>child-first</code> (應用優先)	類載入順序: <code>parent-first</code>
<code>--jsp-precompile=<jspPrecompile></code> 檔案: <code>true</code> 或 <code>false</code>	部署時是否預編譯所有 JSP
<code>--deploy-order=<deployOrder></code> 優先，範圍 0-2147483647	應用啟動優先級，數值越小越
<code>--security-domain=<securityDomain></code> <code>aascli security-domain list</code> 查看可選安全域	應用使用的安全域，可透過
<code>--session-manager=<sessionManager></code> 過 <code>aascli session-manager list</code> 查看可選項；傳	應用綁定的會話管理器，可透
<code>--class-level=<classLevel></code> 密、4=絕密	空字串可解綁 應用安全密級：2=秘密、3=機
<code>--description=<description></code>	應用備註說明，最多1000字元
<code>--deploy-props=<deployProps></code> <code>key1=value1,key2=value2</code> ；置空可傳空字串	自訂部署參數鍵值對，格式：

10.1.4.2 選項

`--reload`

更新成功後立即重新載入應用，使配置即時生效。不指定時配置在應用下次載入時生效。

`--context-path=<contextPath>`

存取應用的上下文路徑，預設為應用名稱，同一虛擬主機內不可重複。

```
--class-loader-order=<classLoaderOrder>
```

類載入順序：parent-first（父類優先）或 child-first（應用優先）。

```
--jsp-precompile=<jspPrecompile>
```

部署時是否預編譯所有 JSP 檔案：true 或 false。

```
--deploy-order=<deployOrder>
```

應用啟動優先級，數值越小越優先，範圍 0-2147483647。

```
--security-domain=<securityDomain>
```

應用使用的安全域。

```
--session-manager=<sessionManager>
```

應用綁定的會話管理器；傳空字串可解綁。

```
--class-level=<classLevel>
```

應用安全密級：2=秘密、3=機密、4=絕密。

```
--description=<description>
```

應用備註說明，最多1000字元。

```
--deploy-props=<deployProps>
```

自訂部署參數鍵值對，格式：key1=value1,key2=value2；置空可傳空字串。

10.1.4.3 示例

示例1 更新應用備註並立即重新載入應用

```
admin@localhost> app update testapp --description=cli-guide-demo --
reload
```

應用配置更新成功。
應用重新載入成功。

示例2 修改應用啟動優先級

```
admin@localhost> app update testapp --deploy-order=50
應用配置更新成功。
```

10.1.5 卸載應用 (app delete)

`app delete` 子命令用於卸載應用。若回收站功能已啟用，卸載的應用將進入回收站；否則直接刪除。

10.1.5.1 概要

```
Usage: app delete <NAME> [--virtual-host=<host>] [-f] [-h]
```

卸載應用

<code><name></code>	應用名稱
<code>--virtual-host=<virtualHost></code>	虛擬主機名稱，可透過 <code>aascli host list</code> 查詢；存在多個同名應用時必填
<code>-f, --force</code>	不提示確認
<code>--help</code>	顯示說明資訊

10.1.5.2 選項

```
-f, --force
```

不提示確認，直接卸載。適用於腳本自動化場景。

10.1.5.3 示例

示例1 卸載應用（需確認）

確認提示中顯示的是應用的數字 ID，可透過 `app get` 查詢：

```
admin@localhost> app delete testapp
確定要卸載應用 11 嗎？(Y/N) y
應用卸載成功。
```

示例2 卸載應用（免確認）

```
admin@localhost> app delete testapp -f
```

應用卸載成功。

10.1.6 重新載入應用 (app reload)

`app reload` 子命令用於重新載入應用，通常用於應用內容更新後使其重新生效。

10.1.6.1 概要

```
Usage: app reload <NAME> [--virtual-host=<host>] [-h]
```

重新載入應用

<code><name></code>	應用名稱
<code>--virtual-host=<virtualHost></code>	虛擬主機名稱，可透過 <code>aascli host list</code> 查詢；存在多個同名應用時必填
<code>--help</code>	顯示說明資訊

10.1.6.2 示例

示例1 重新載入應用

```
admin@localhost> app reload testapp
```

應用重新載入成功。

10.1.7 部署應用 (app deploy)

`app deploy` 子命令用於部署一個新的應用，應用包為 WAR 檔案或應用資料夾。

10.1.7.1 概要

```
Usage: app deploy --app-name=<name> --app-path=<path> [選項] [-h]
```

部署應用

<code>--app-name=<appName></code>	應用名稱，僅支援字母、數字、底線、短橫線或點號，最多255字元
<code>--app-path=<appPath></code>	應用 WAR 檔案或應用資料夾在伺服器上的路徑
<code>--context-path=<contextPath></code>	存取應用的上下文路徑，預設為應用名稱，同一虛擬主機內不可重複
<code>--class-loader-order=<classLoaderOrder></code>	類載入順序：parent-first (父類優先) 或 child-first (應用優先)，預設

```

--jsp-precompile=<jspPrecompile>
檔案, 預設 false
--virtual-host=<virtualHost>
localhost, 可透過 aascli host list 查詢
--deploy-order=<deployOrder>
優先, 範圍 0-2147483647, 預設 100
--security-domain=<securityDomain>
aascli security-domain list 查看可選安全域
--session-manager=<sessionManager>
過 aascli session-manager list 查看可選項
--class-level=<classLevel>
密、4=絕密, 預設 2
--description=<description>
--deploy-props=<deployProps>
key1=value1,key2=value2
--help

```

child-first

部署時是否預編譯所有 JSP

應用綁定的虛擬主機, 預設

應用啟動優先級, 數值越小越

應用使用的安全域, 可透過

應用綁定的會話管理器, 可透

應用安全密級: 2=秘密、3=機

應用備註說明, 最多1000字元
自訂部署參數鍵值對, 格式:

顯示說明資訊

10.1.7.2 選項

```
--app-name=<appName>
```

應用名稱, 僅支援字母、數字、底線、短橫線或點號, 最多255字元。必填。

```
--app-path=<appPath>
```

應用 WAR 檔案或應用資料夾在伺服器上的路徑。必填。路徑需為伺服器可存取的本機路徑, 建議避免包含中文。

其餘選項同 `app update`。

10.1.7.3 示例

示例1 部署應用資料夾

```
admin@localhost> app deploy --app-name=testapp --app-path=D:/aas-
testapp --description=cli-demo
```

應用創建成功, ID: 12

應用部署成功。

示例2 部署 WAR 包並指定上下文路徑與密級

```
admin@localhost> app deploy --app-name=demo --app-
path=/opt/apps/demo.war --context-path=/demo --class-level=3
應用創建成功，ID：13
應用部署成功。
```

注意： `--app-path` 必須為伺服器本機可存取的路徑，建議使用不含中文與空格的路徑。

10.1.8 重新部署應用 (app redeploy)

`app redeploy` 子命令用於以新的應用包重新部署指定應用，應用配置未指定的項保持不變。

10.1.8.1 概要

```
Usage: app redeploy <NAME> [--virtual-host=<host>] --app-path=<path>
[選項] [-h]
```

重新部署應用	
<code><name></code>	應用名稱
<code>--virtual-host=<virtualHost></code>	虛擬主機名稱，可透過
<code>aascli host list</code> 查詢；存在多個同名應用時必填	
<code>--app-path=<appPath></code>	應用 WAR 檔案或應用資料夾
在伺服器上的路徑	
<code>--context-path=<contextPath></code>	存取應用的上下文路徑，預設
為應用名稱，同一虛擬主機內不可重複	
<code>--class-loader-order=<classLoaderOrder></code>	類載入順序：parent-first
(父類優先) 或 <code>child-first</code> (應用優先)	
<code>--jsp-precompile=<jspPrecompile></code>	部署時是否預編譯所有 JSP
檔案：true 或 false	
<code>--deploy-order=<deployOrder></code>	應用啟動優先級，數值越小越
優先，範圍 0-2147483647	
<code>--security-domain=<securityDomain></code>	應用使用的安全域，可透過
<code>aascli security-domain list</code> 查看可選安全域	
<code>--class-level=<classLevel></code>	應用安全密級：2=秘密、3=機
密、4=絕密	
<code>--description=<description></code>	應用備註說明，最多1000字元
<code>--deploy-props=<deployProps></code>	自訂部署參數鍵值對，格式：

```
key1=value1,key2=value2
```

```
--help
```

顯示說明資訊

10.1.8.2 示例

示例1 重新部署應用

```
admin@localhost> app redeploy testapp --app-path=D:/aas-testapp
應用重新部署成功。
```

10.1.9 應用類庫管理 (app class-lib)

`app class-lib` 子命令用於管理指定應用的共享類庫 (JAR 包) , 包括列出、新增、刪除類庫。

10.1.9.1 概要

```
Usage:  app class-lib [-h] [COMMAND]
```

應用類庫管理

```
-h, --help    顯示說明資訊
```

Commands:

```
list      列出應用類庫
add       新增應用類庫
delete    刪除應用類庫
```

```
Usage: app class-lib list <NAME> [--virtual-host=<host>] [--page=
<page>] [--output=<format>] [--no-paging] [-h]
```

```
Usage: app class-lib add <NAME> [--virtual-host=<host>] --file <jar>
[-h]
```

```
Usage: app class-lib delete <NAME> [--virtual-host=<host>] --jar-name
<name> [-f] [-h]
```

10.1.9.2 選項

```
--file=<file>
```

`add` 時指定 JAR 檔案路徑。

```
--jar-name=<jarName>
```

`delete` 時指定類庫 JAR 檔案名，可透過 `app class-lib list` 查詢。

10.1.9.3 示例

示例1 新增並查看應用類庫

```
admin@localhost> app class-lib add testapp --file=D:/aas-h2.jar
類庫上傳成功。

admin@localhost> app class-lib list testapp
| 類庫名稱      | 大小      | 版本 |
+-----+
| aas-h2.jar    | 2.49 MB   | -    |
第 1 / 1 頁, 共 1 條
```

示例2 刪除應用類庫

```
admin@localhost> app class-lib delete testapp --jar-name=aas-h2.jar -f
類庫刪除成功。
```

10.1.10 應用增量更新 (app increment)

`app increment` 子命令用於管理指定應用的增量更新包，適用於僅更新應用部分檔案的場景。

10.1.10.1 概要

```
Usage:  app increment [-h] [COMMAND]
```

應用增量更新

`-h, --help` 顯示說明資訊

Commands:

```
list      列出應用增量包
add       新增應用增量包
delete    刪除應用增量包
```

```
Usage: app increment list <NAME> [--virtual-host=<host>] [--page=
<page>] [--output=<format>] [--no-paging] [-h]
```

```
Usage: app increment add <NAME> [--virtual-host=<host>] --file <pkg>
[--description <desc>] [-h]
```

```
Usage: app increment delete <NAME> [--virtual-host=<host>] --
increment-id <ID> [-f] [-h]
```

10.1.10.2 選項

```
--file=<file>
```

`add` 時指定增量包檔案路徑（zip 格式壓縮包）。

```
--description=<description>
```

增量包描述。

```
--increment-id=<incrementId>
```

`delete` 時指定增量包 ID，可透過 `app increment list` 查詢。

10.1.10.3 示例

示例1 新增量包

增量包上傳後將自動應用到目標應用：

```
admin@localhost> app increment add testapp --file=D:/aas-increment.zip
--description=update-web-xml
增量包上傳並應用成功，ID：3
```

示例2 列出應用增量包

```
admin@localhost> app increment list testapp
| id | 名稱 | 大小 | 描述 | 上傳時間 |
+----+-----+-----+-----+-----+
| 3 | aas-increment.zip | 236 B | update-web-xml | 2026-09-27 17:52 |
第 1 / 1 頁, 共 1 條
```

示例3 刪除應用增量包

```
admin@localhost> app increment delete testapp --increment-id=3 -f
增量包刪除成功。
```

10.1.11 備份應用 (app backup)

`app backup` 子命令用於對指定應用進行手動備份，備份檔案可用於還原應用。

10.1.11.1 概要

Usage: `app backup <NAME> [--virtual-host=<host>] [-h]`

備份應用

<code><name></code>	應用名稱
<code>--virtual-host=<virtualHost></code>	虛擬主機名稱，可透過 <code>aascli host list</code> 查詢；存在多個同名應用時必填
<code>--help</code>	顯示說明資訊

10.1.11.2 示例

示例1 備份應用

```
admin@localhost> app backup testapp
應用備份創建成功。
```

10.1.12 應用備份管理 (app-backup)

`app-backup` 子命令用於管理應用備份，包括查看、下載、還原、刪除備份及備份設定。

10.1.12.1 概要

Usage: `app-backup [-h] [COMMAND]`

應用備份

`-h, --help` 顯示說明資訊

Commands:

<code>list</code>	列出應用備份
<code>delete</code>	刪除應用備份
<code>download</code>	下載應用備份
<code>restore</code>	從備份還原應用
<code>settings</code>	備份設置

10.1.12.2 示例

示例1 列出應用備份

```
admin@localhost> app-backup list
```

備份名稱	應用名稱	上下文路徑	備份時間
testapp_1790502776901.tar.gz	testapp	/testapp	2026-09-27 17:52

大小 | 768 B |

第 1 / 1 頁, 共 1 條

示例2 下載應用備份

```
admin@localhost> app-backup download testapp_1790502776901.tar.gz --
file=D:/aas-tw-backup.tar.gz
```

備份已下載到: D:\aas-tw-backup.tar.gz

示例3 從備份還原應用

```
admin@localhost> app-backup restore testapp_1790502776901.tar.gz
```

應用從備份恢復成功。

`restore` 支援在還原時覆蓋原備份中的應用配置（上下文路徑、虛擬主機、密級等），不指定時使用備份中的原值。

示例4 刪除應用備份

```
admin@localhost> app-backup delete testapp_1790502776901.tar.gz -f
```

備份刪除成功。

示例5 查看備份設定

```
admin@localhost> app-backup settings get
```

備份設置

備份目錄: \${apusic.home}/store/app_backup

備份保留天數: 7

示例6 修改備份設定

```
admin@localhost> app-backup settings set --
path=${apusic.home}/store/app_backup --retention-days=7
備份設置已保存。
```

`settings set` 選項說明：

選項	說明
<code>--path=</code>	備份目錄
<code>--retention-days=</code>	備份保留天數，範圍 1-365

10.1.13 應用回收管理 (app-recycle)

`app-recycle` 子命令用於管理應用回收站。啟用回收站後，卸載的應用將進入回收站，可還原或徹底刪除。

10.1.13.1 概要

```
Usage: app-recycle [-h] [COMMAND]
```

應用回收

`-h, --help` 顯示說明資訊

Commands:

<code>list</code>	列出回收站應用
<code>restore</code>	還原回收站應用
<code>delete</code>	徹底刪除回收站應用
<code>settings</code>	回收設置

10.1.13.2 示例

示例1 查看回收設定並啟用回收站

```
admin@localhost> app-recycle settings get
```

回收設置

啟用回收：關閉

回收路徑：\${apusic.home}/store/app_recycle

保存天數：30

```
admin@localhost> app-recycle settings set --enabled=true
```

回收設置已保存。

`settings set` 選項說明：

選項	說明
<code>--enabled=true false</code>	是否啟用回收
<code>--path=</code>	回收路徑
<code>--retention-days=</code>	回收保留天數，範圍 1-365

示例2 列出回收站應用

```
admin@localhost> app-recycle list
| 回收文件名 | 應用名稱 | 上下文路徑 | 卸載時間 | 大小 |
+-----+-----+-----+-----+-----+
-----+
| testapp_20260927175226 | testapp | /testapp | 2026-09-27 17:52 | 808 B |
第 1 / 1 頁, 共 1 條
```

示例3 還原回收站應用

`restore` 需指定完整的回收檔案名稱（而非應用名稱）：

```
admin@localhost> app-recycle restore testapp_20260927175226
應用還原成功。
```

還原時同樣支援覆蓋上下文路徑、虛擬主機、密級等配置項，不指定時使用回收站中的原值。

示例4 徹底刪除回收站應用

```
admin@localhost> app-recycle delete testapp_20260927175226 -f
回收站應用已徹底刪除。
```

10.2 配置管理

10.2.1 虛擬主機管理 (host)

`host` 子命令用於管理虛擬主機，包括查看、創建、更新、刪除虛擬主機及設定預設虛擬主機。

10.2.1.1 概要

Usage: host [-h] [COMMAND]

虛擬主機管理

-h, --help 顯示說明資訊

Commands:

list 列出虛擬主機
get 查看虛擬主機詳情
add 創建虛擬主機
update 更新虛擬主機
delete 刪除虛擬主機
set-default 將虛擬主機設為默認

host add / host update 選項:

選項	說明
--name=	虛擬主機唯一業務標識 (add 必填)
--docBase=	文檔根目錄路徑，為空時預設 webapps；同一文檔根目錄不允許被多個虛擬主機使用
--alias=	主機名稱或 IP 位址，多個用英文逗號分隔
--sso=true false	單點登錄模式配置，啟用後當前虛擬主機啟用 SSO 登錄邏輯
--sso-http-only=true false	設定 SSO Cookie 的 HttpOnly 屬性
--access-ip-whitelist=	允許存取的 IP，可用 * 萬用字元，逗號分隔，留空代表不限制，示例： 192.168.*,10.0.0.8
--access-ip-blacklist=	禁止存取的 IP，格式同上
--access-domain-whitelist=	允許存取的網域名稱，可用 * 萬用字元，逗號分隔，留空代表不限制，示例： *.example.com
--access-domain-blacklist=	禁止存取的網域名稱，格式同上
--access-time=	僅配置時段內允許用戶端存取；留空代表無限制；格式為時:分:秒-時:分:秒，逗號分隔多個時段
--extra-props= <header=value,...>	HTTP 回應頭欄位，如 X-Frame-Options=SAMEORIGIN

10.2.1.2 示例

示例1 列出虛擬主機

```
admin@localhost> host list
```

名稱	別名	文檔根目錄	默認
admin		admin	是
localhost		webapps	是

第 1 / 1 頁, 共 2 條

示例2 查看虛擬主機詳情

```
admin@localhost> host get localhost
```

名稱: localhost
 文檔根目錄: webapps
 主機名稱或IP地址, 多個用英文逗號分隔:
 域名:
 單點登錄模式配置, 啟用後當前虛擬主機啟用 SSO 登錄邏輯, 關閉則本虛擬主機不啟用單點登錄: 禁用
 設定SSO Cookie的HttpOnly屬性: 否
 允許訪問的IP, 可用 * 通配, 逗號分隔, 留空代表不限制, 示例: 192.168.*,10.0.0.8:
 禁止訪問的IP, 可用 * 通配, 逗號分隔, 留空代表不限制:
 允許訪問的域名, 可用 * 通配, 逗號分隔, 留空代表不限制, 示例: *.example.com:
 禁止訪問的域名, 可用 * 通配, 逗號分隔, 留空代表不限制:
 僅配置時段內允許客戶端訪問; 留空代表無限制; 格式為時:分:秒-時:分:秒, 逗號分隔多個時段:
 HTTP 響應頭字段, 如 X-Frame-Options=SAMEORIGIN, 響應頭名只能包含字母、數字及 !#\$%&*+-.^_`|~:

示例3 創建虛擬主機

```
admin@localhost> host add --name=test-host --docBase=webapps-test --alias=www.test.com --sso=false
```

虛擬主機創建成功, 重啟服務器後生效。

示例4 更新虛擬主機存取 IP 白名單

```
admin@localhost> host update test-host --access-ip-whitelist=192.168.*,10.0.0.8
```

虛擬主機配置保存成功，重啟服務器後生效。

示例5 將虛擬主機設為默認

```
admin@localhost> host set-default test-host
確定要將虛擬主機 test-host 設為默認嗎？ (y/N) y
虛擬主機 test-host 已設為默認。
```

示例6 刪除虛擬主機

```
admin@localhost> host delete test-host -f
虛擬主機刪除成功，重啟服務器後生效。
```

10.2.2 執行緒池管理 (threadpool)

`threadpool` 子命令用於管理執行緒池，包括查看、創建、更新、刪除執行緒池。

10.2.2.1 概要

Usage: `threadpool [-h] [COMMAND]`

執行緒池管理

`-h, --help` 顯示說明資訊

Commands:

<code>list</code>	列出執行緒池
<code>get</code>	查看執行緒池詳情
<code>add</code>	創建執行緒池
<code>update</code>	更新執行緒池
<code>delete</code>	刪除執行緒池

`threadpool add` 選項 (`update` 時名稱由參數指定，其餘選項相同)：

選項	說明
<code>--name=</code>	執行緒池名稱 (必填)
<code>--min-threads=</code>	最小執行緒數 (必填, 1-65535)
<code>--max-threads=</code>	最大執行緒數 (必填, 1-65535)
<code>--queue-size=</code>	佇列大小 (必填, 1-65535)

--max-idle-time=	空閒超時時間 ms (必填, 1-86400000)
--priority=	執行緒優先級 (必填, 1-10)
--connection-auto-reclaim=	數據源連接自動回收 (必填: false/reclaim/detect)
--thread-step-reclaim=	階梯式回收 (true 或 false, 預設 false)
--thread-step-size=	單次回收數 (thread-step-reclaim=true 時必填, 1-65535)
--thread-step-time=	回收時間間隔 ms (thread-step-reclaim=true 時必填, 1-86400000)

10.2.2.2 示例

示例1 列出執行緒池

```
admin@localhost> threadpool list
| 執行緒池名稱      | 最大執行緒數 | 核心執行緒數 | 佇列大小 | 空閒時間      | 優先
級 |
+-----+-----+-----+-----+-----+-----+
-----+-----+
| admin-thread-pool | 100          | 25          | 5000     | 60000        |
ms   | 5           |
| http-thread-pool  | 200          | 25          | 5000     | 60000        |
ms   | 5           |
第 1 / 1 頁, 共 2 條
```

示例2 查看執行緒池詳情

```
admin@localhost> threadpool get http-thread-pool
執行緒池名稱: http-thread-pool
最小執行緒數: 25
最大執行緒數: 200
佇列大小: 5000
空閒超時時間: 60000 ms
階梯式回收: 關閉
數據源連接自動回收: 不啟用
執行緒優先級: 5
```

示例3 創建執行緒池

```
admin@localhost> threadpool add --name=test-pool --min-threads=10 --max-threads=100 --queue-size=1000 --max-idle-time=60000 --priority=5 --connection-auto-reclaim=false
```

執行緒池創建成功，重啟服務器後生效。

示例4 更新執行緒池

```
admin@localhost> threadpool update test-pool --max-threads=200
```

執行緒池配置保存成功，重啟服務器後生效。

示例5 刪除執行緒池

```
admin@localhost> threadpool delete test-pool -f
```

執行線程池刪除成功，重啟服務器後生效。

10.2.3 會話管理器管理 (session-manager)

`session-manager` 子命令用於管理會話管理器，支援記憶體、檔案、Redis、資料庫四種儲存類型。

10.2.3.1 概要

Usage: `session-manager [-h] [COMMAND]`

會話管理器管理

`-h, --help` 顯示說明資訊

Commands:

`list` 列出會話管理器
`get` 查看會話管理器詳情
`add` 創建會話管理器
`update` 更新會話管理器 (名稱與類型不可修改)
`delete` 刪除會話管理器 (被應用綁定的會話管理器不允許刪除)

`session-manager add` 主要選項 (`update` 時名稱與類型不可修改，其餘選項相同)：

選項	說明
<code>--name=</code>	會話管理器名稱 (必填，字母、數字、底線、短橫線，最長64字元)
<code>--type=</code>	儲存類型 (必填)：memory、file、redis、database
<code>--max-sessions=</code>	最大會話數，-1 表示不限制，預設 -1

--persist-path=	memory 類型：持久化檔案路徑，預設 SESSIONS.ser
--save-on-restart=	file/database 類型：重啟時保存會話（預設 true）
--max-idle-backup=	file/database 類型：空閒備份閾值秒（-1 不啟用，預設 -1）
--min-idle-swap=	file/database 類型：最小空閒換出閾值秒（-1 不啟用，預設 -1）
--max-idle-swap=	file/database 類型：最大空閒換出閾值秒（-1 不啟用，預設 -1）
--persistence-dir=	file 類型：持久化目錄
--data-source-name=	database 類型：數據源 JNDI 名稱（database 類型必填）
--local-data-source=	database 類型：使用應用本機數據源（預設 false）
--redis-mode=	redis 類型：standalone 或 sentinel，預設 standalone
--redis-nodes= host:port [,...]	redis 類型：節點位址，standalone 單個 host:port，sentinel 逗號分隔多個
--redis-master=	redis 類型：哨兵主節點名稱（sentinel 模式必填）
--redis-database= <0-15>	redis 類型：資料庫索引，預設 0
--redis-password=	redis 類型：密碼，最長256字元
--session-persist-policies=	redis 類型：持久化策略 SAVE_ON_CHANGE、ALWAYS_SAVE_AFTER_REQUEST，可逗號組合，預設空
--redis-connect-timeout=	redis 類型：連接超時 ms，預設 2000
--redis-max-conn=	redis 類型：最大連接數，預設 100
--redis-min-idle=	redis 類型：最小空閒連接數，預設 10
--redis-max-idle=	redis 類型：最大空閒連接數，留空使用預設值 8
--redis-max-wait=	redis 類型：取得連接最大等待 ms，預設 3000

另有會話屬性過濾相關進階選項（`--session-attribute-name-filter`、`--session-attribute-value-class-name-filter`、`--warn-on-filter-failure`、`--notify-attribute-on-unchanged`、`--notify-binding-on-unchanged`），可透過 `session-manager add --help` 查看詳細說明。

10.2.3.2 示例

示例1 創建記憶體類型會話管理器

```
admin@localhost> session-manager add --name=test-sm --type=memory
```

會話管理器創建成功。

示例2 列出會話管理器

```
admin@localhost> session-manager list
```

名稱	類型	創建時間	更新時間
test-sm	內存	2026-09-27 17:52:41	2026-09-27 17:52:41

第 1 / 0 頁, 0 條記錄

示例3 查看會話管理器詳情

```
admin@localhost> session-manager get test-sm
```

名稱: test-sm
 類型: 內存
 最大會話數: 不限制
 持久化文件路徑: SESSIONS.ser
 會話屬性名過濾器:
 會話屬性值類名過濾器:
 過濾失敗時告警: 關閉
 屬性未變化仍派發事件: 開啟
 綁定未變化仍派發事件: 關閉

示例4 刪除會話管理器

被應用綁定的會話管理器不允許刪除，需先解綁應用：

```
admin@localhost> session-manager delete test-sm -f
```

會話管理器刪除成功。

10.2.4 HTTP 通道管理 (httpchannel)

`httpchannel` 子命令用於管理 HTTP/HTTPS 通道（網路監聽通道），包括查看、創建、更新、刪除通道。

10.2.4.1 概要

```
Usage: httpchannel [-h] [COMMAND]
```

HTTP 通道管理

`-h, --help` 顯示說明資訊

Commands:

`list` 列出 HTTP 通道
`get` 查看 HTTP 通道詳情
`add` 創建 HTTP 通道
`update` 更新 HTTP 通道
`delete` 刪除 HTTP 通道

`httpchannel add` 基本選項 (`update` 時不指定名稱選項，其餘相同，預設保持當前配置)：

選項	說明
<code>--name=</code>	通道名稱（必填）。長度 1-64 字元，僅支援字母、數字、底線、短橫線
<code>--protocol=</code>	IO 模式（必填）：NIO、NIO2、APR
<code>--host=</code>	監聽位址，0.0.0.0 代表監聽所有網卡，預設 0.0.0.0
<code>--port=</code>	監聽埠（必填），1-65535，1024 以下需 root 權限
<code>--executor=</code>	業務執行緒池（必填），可用 <code>aascli threadpool list</code> 查詢候選項
<code>--redirect-port=</code>	HTTP 請求需加密時自動跳轉的 HTTPS 埠，1-65535
<code>--receive-buffer-size=</code>	接收緩衝區大小（TCP SO_RCVBUF），預設 8192 位元組
<code>--send-buffer-size=</code>	傳送緩衝區大小（TCP SO_SNDBUF），預設 8192 位元組
<code>--tcp-no-delay=</code>	設定 TCP_NODELAY，預設 true
<code>--connection-timeout=</code>	連接超時（TCP SO_TIMEOUT），預設 20000 毫秒
<code>--accept-count=</code>	TCP backlog，預設 100
<code>--max-connections=</code>	最大並發 TCP 連接數，-1 表示無限制，預設 8192
<code>--max-keep-alive-requests=</code>	單個 TCP 連線處理的最大請求數，預設 100
<code>--keep-alive-timeout=</code>	TCP 連線空閒超時時間，預設 20 秒
<code>--uri-encoding=</code>	解析 URL 路徑使用的字元編碼，預設 UTF-8
<code>--compression=</code>	壓縮模式 on/off/force，預設 on
<code>--compressible-mime-type=</code>	僅列表內資源類型才執行 GZIP 壓縮，逗號分隔
<code>--compression-min-size=</code>	回應體小於該值時不執行壓縮，預設 2048 位元組

--max-post-size=	POST 請求體的資料上限，-1 表示無限制，預設 2097152 位元組
--max-parameter-count=	單次請求的最大參數總數，預設 10000
--max-header-count=	請求頭總欄位上限，預設 100
--max-http-header-size=	請求頭的總位元組上限，預設 8192 位元組
--response-headers=	自訂 HTTP 回應頭，格式 Name=Value,Name2=Value2

SSL/HTTPS 相關選項：

選項	說明
--ssl-enabled=	啟用 SSL 加密，僅 HTTPS 協定通道需配置，預設 false
--gm-enabled=	啟用國密認證，開啟後需要校驗是否配置國密證書，預設 false
--tls-protocols=	TLS 版本，多個逗號分隔，預設 TLSv1.2,TLSv1.3
--client-auth=	用戶端驗證：false=禁用、want=可選、true=需要（雙向認證），預設 false
--certificate-custom=	證書自訂，關閉時選擇證書庫，開啟時輸入自訂證書庫資訊
--keystore-name=	密鑰庫名稱（證書自訂關閉時使用），可用 aascli cert list 查詢
--keystore-entry=	密鑰庫證書條目，可用 aascli cert get 查詢候選項
--truststore-name=	信任庫名稱（證書自訂關閉且用戶端驗證開啟時使用）
--truststore-entry=	信任庫證書條目
--keystore-path=	密鑰庫檔案路徑（證書自訂開啟時使用）
--keystore-password=	密鑰庫存取密碼（證書自訂開啟時使用）
--certificate-format=	密鑰庫類型：JKS、PKCS12，預設 JKS
--certificate-alias=	證書別名
--key-password=	證書條目的保護密碼
--truststore-path=	信任庫檔案路徑（證書自訂開啟且用戶端驗證開啟時使用）
--truststore-password=	信任庫存取密碼
--truststore-type=	信任庫類型：JKS、PKCS12，預設 JKS
--ssl-cipher-suites=	SSL 非國密時的密碼套件，可用 aascli cert cipher-suites 查詢候選項，逗號分隔

--http2-enabled=	開啟 HTTP/2 需先啟用 SSL, 預設 false
------------------	------------------------------

另提供慢請求監控 (--long-thread-*)、SDOS 防護 (--sdos-*) 及 HTTP/2 詳細參數 (--http2-*) 等進階選項, 可透過 `httpchannel add --help` 查看完整說明。

10.2.4.2 示例

示例1 列出 HTTP 通道

```
admin@localhost> httpchannel list
| 名稱          | 位址 | 埠   | IO模式 |
+-----+-----+-----+-----+
| admin-http    |      | 6848 | NIO     |
| ams-http      |      | 6888 | NIO     |
第 1 / 1 頁, 共 2 條
```

示例2 查看 HTTP 通道詳情

```
admin@localhost> httpchannel get ams-http
名稱: ams-http
IO模式: NIO
監聽位址:
監聽埠: 6888
重定向埠: 6443
執行緒池: http-thread-pool
TCP NoDelay: 開啟
連接超時時間: 30000 毫秒
TCP等待佇列大小: 100
長連接最大請求數: 100
長連接超時時間: 15 秒
IP位址匹配虛擬主機: 關閉
慢請求監控: 關閉
黑名單檢測: 關閉
URI編碼: utf-8
請求體編碼解析URI: 關閉
DNS反向查找: 關閉
允許TRACE方法: 關閉
Powered By: 關閉
```

```

異步請求的超時時間: 30000 毫秒
禁用上傳超時: 開啟
最大POST大小: 104857600
最大請求頭個數: 100
最大請求頭大小: 8192 位元組
啟用SSL: 關閉
啟用HTTP/2: 關閉

```

示例3 創建 HTTP 通道

```

admin@localhost> httpchannel add --name=test-channel --protocol=NIO --
port=6999 --executor=http-thread-pool
HTTP 通道創建成功，重啟服務器後生效。

```

示例4 更新 HTTP 通道最大連接數

```

admin@localhost> httpchannel update test-channel --max-
connections=4096
HTTP 通道配置保存成功，重啟服務器後生效。

```

示例5 刪除 HTTP 通道

```

admin@localhost> httpchannel delete test-channel -f
HTTP 通道刪除成功，重啟服務器後生效。

```

10.2.5 JVM 配置管理 (jvm)

`jvm` 子命令用於查看、更新、重置 JVM 啟動配置（堆大小、GC 演算法、除錯選項等）。

10.2.5.1 概要

```

Usage:  jvm [-h] [COMMAND]
JVM 配置管理
    -h, --help    顯示說明資訊
Commands:
    get          查看 JVM 配置
    update       更新 JVM 配置
    reset        恢復 JVM 預設配置

```

jvm update 選項：

選項	說明
--xms=	堆最小值 (-Xms), 單位 MB
--xmx=	堆最大值 (-Xmx), 單位 MB
--metaspace-size=	元空間初始值, 單位 MB
--max-metaspace-size=	元空間最大值, 單位 MB
--gc-type=	垃圾回收演算法, 可選: ParallelGC、G1GC、ConcMarkSweepGC、ParallelOldGC、ZGC、SerialGC
--debug=	開啟除錯 (true 或 false)
--debug-options=	除錯選項, 例如: -agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=8000
--native-library-path=	本機庫路徑, 多個路徑用分號分隔
--jvm-options=	JVM 選項, 多條以分號分隔

10.2.5.2 示例

示例1 查看 JVM 配置

```
admin@localhost> jvm get
JDK路徑: D:\jdk\jdk-17.0.7
Java版本: 17.0.7
堆最小值 (-Xms): 1024
堆最大值 (-Xmx): 4096
元空間初始值:
元空間最大值:
垃圾回收演算法:
開啟調試: 關閉
調試選項:
本地庫路徑:
JVM選項列表:
```

示例2 更新 JVM 堆記憶體配置

```
admin@localhost> jvm update --xms=1024 --xmx=4096
```

JVM 配置保存成功，重啟服務器後生效。

示例3 恢復 JVM 預設配置

```
admin@localhost> jvm reset
```

10.2.6 JMX 配置管理 (jmx)

`jmx` 子命令用於查看、更新、重置 JMX 遠端監控配置。

10.2.6.1 概要

```
Usage: jmx [-h] [COMMAND]
```

JMX 配置管理

`-h, --help` 顯示說明資訊

Commands:

`get` 查看 JMX 配置
`update` 更新 JMX 配置
`reset` 恢復 JMX 預設配置

`jmx update` 主要選項：

選項	說明
<code>--enabled=</code>	啟用 JMX (true 或 false)
<code>--host=</code>	監聽位址，留空預設監聽本機 127.0.0.1
<code>--port=</code>	監聽埠，1-65535，留空預設 6868
<code>--authenticate=</code>	啟用認證 (true 或 false)
<code>--password-file=</code>	密碼檔案路徑，如 <code>conf/jmxremote.password</code>
<code>--access-file=</code>	存取控制檔案路徑，如 <code>conf/jmxremote.access</code>
<code>--ssl-enabled=</code>	啟用 SSL (true 或 false)
<code>--client-auth=</code>	用戶端校驗 (true 或 false)
<code>--certificate-custom=</code>	證書自訂 (true 或 false)
<code>--key-store-file=</code>	密鑰庫路徑，如 <code>conf/jmx.keystore</code>

--key-password=	密鑰密碼
--key-store-password=	密鑰庫密碼
--key-store-type=	密鑰庫類型, JKS 或 PKCS12
--trust-store-file=	信任庫路徑, 如 conf/jmx.truststore
--trust-store-password=	信任庫密碼
--trust-store-type=	信任庫類型, JKS 或 PKCS12
--enabled-cipher-suites=	密碼套件, 多個以英文逗號分隔, 可用 <code>aascli cert cipher-suites</code> 查詢候選項
--key-store-name=	密鑰庫名稱 (證書自訂關閉時使用), 可用 <code>aascli cert list</code> 查詢候選項
--trust-store-name=	信任庫名稱 (證書自訂關閉時使用)
--key-store-entry=	密鑰庫條目別名, 可用 <code>aascli cert get</code> 查詢候選項

10.2.6.2 示例

示例1 查看 JMX 配置

```
admin@localhost> jmx get
啟用 JMX: 關閉
```

示例2 啟用 JMX

```
admin@localhost> jmx update --enabled=true
JMX 配置保存成功, 重啟服務器後生效。
```

示例3 恢復 JMX 預設配置

```
admin@localhost> jmx reset
```

10.2.7 證書管理 (cert)

`cert` 子命令用於管理證書庫 (密鑰庫/信任庫), 包括證書庫的創建、匯入、下載、刪除及證書條目管理。

10.2.7.1 概要

```
Usage: cert [-h] [COMMAND]
證書管理
    -h, --help    顯示說明資訊
```

Commands:

list	列出證書庫
get	查看證書庫詳情及條目列表
add	新建證書庫
update	更新證書庫
delete	刪除證書庫
delete-entry	刪除證書條目
import	從證書庫檔案匯入條目
import-pem	匯入 PEM 格式證書或私鑰
download	下載證書庫檔案
cipher-suites	查詢支援的密碼套件

cert add 選項:

選項	說明
--name=	證書庫名稱（必填），最長 64 字元，僅支援字母、數字、底線、短橫線、點號，全域唯一
--type=	證書庫類型，JKS 或 PKCS12，預設 JKS
--store-type=	庫類型，keystore（密鑰庫）或 truststore（信任庫），預設 keystore
--password=	證書庫密碼，8-32 字元，省略時互動式輸入

cert import 選項:

選項	說明
	目標證書庫名稱
--file=	證書庫檔案路徑，支援 .jks/.p12/.pfx/.pkcs12，大小不超過 10MB
--source-password=	來源證書庫密碼，省略時互動式輸入
--aliases=	要匯入的別名，多個以英文逗號分隔，省略則匯入全部
--target-alias=	匯入後在目標證書庫中的別名，留空則使用來源別名
--entry-password=	條目保護密碼，僅 PKCS12 類型的密鑰庫可用，留空則使用目標證書庫密碼

cert import-pem 選項:

選項	說明
----	----

	目標證書庫名稱
--file=	PEM 檔案路徑，內容為證書或私鑰（含 BEGIN/END 標記）
--alias=	匯入後在目標證書庫中的別名
--entry-password=	條目保護密碼，僅 PKCS12 類型的密鑰庫可用，留空則使用目標證書庫密碼

10.2.7.2 示例

示例1 新建證書庫

```
admin@localhost> cert add --name=test-store --password=Test12345
證書庫創建成功，ID：4
```

示例2 列出證書庫

```
admin@localhost> cert list
| 名稱          | 用途          | 類型 | 路徑          |
+-----+-----+-----+-----+
| test-store | keystore | JKS   | D:\AAMS-V10.0.8\store\ssl\test-
store.jks |
第 1 / 0 頁, 共 0 條
```

示例3 從證書庫檔案匯入條目

```
admin@localhost> cert import test-store --file=D:/aas-source.jks --
source-password=Source123
證書條目導入成功。
```

示例4 查看證書庫詳情及條目列表

```
admin@localhost> cert get test-store
名稱: test-store
路徑: D:\AAMS-V10.0.8\store\ssl\test-store.jks
用途: 密鑰庫
類型: JKS

證書條目:
| 別名          | 條目類型 | 算法          | 有效期起      | 有效期止      |
```

```
+-----+-----+-----+-----+-----+
| demo-cert | 私钥      | SHA256WITHRSA | 2026-09-27 | 2027-09-27 |
```

條目詳情：

demo-cert:

Subject: CN=demo,OU=test,O=test,L=bj,ST=bj,C=CN

Issuer: CN=demo,OU=test,O=test,L=bj,ST=bj,C=CN

示例5 下載證書庫檔案

```
admin@localhost> cert download test-store --file=D:/aas-tw-store.jks
證書庫文件已下載到: D:\aas-tw-store.jks
```

--file 省略時儲存為當前目錄下的 <證書庫名稱>.<jks|p12>。

示例6 刪除證書條目

```
admin@localhost> cert delete-entry test-store demo-cert -f
證書條目刪除成功。
```

示例7 刪除證書庫

```
admin@localhost> cert delete test-store -f
證書庫刪除成功。
```

示例8 查詢支援的密碼套件

配置 HTTP 通道或 JMX 的 SSL 密碼套件時，可先查詢當前 JDK 支援的密碼套件：

```
admin@localhost> cert cipher-suites
TLS_AES_256_GCM_SHA384
TLS_AES_128_GCM_SHA256
TLS_CHACHA20_POLY1305_SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
... (省略) ...
TLS_EMPTY_RENEGOTIATION_INFO_SCSV
共 23 個密碼套件。
```

10.3 資源管理

10.3.1 數據源管理 (datasource)

`datasource` 子命令用於管理 JDBC 數據源，包括數據源的創建、查看、更新、啟停、連通性測試與刪除。

10.3.1.1 概要

Usage: `datasource [-h] [COMMAND]`

數據源管理

`-h, --help` 顯示說明資訊

Commands:

`list` 列出數據源
`get` 查看數據源詳情
`add` 創建數據源
`update` 更新數據源
`delete` 刪除數據源
`toggle` 啟用或禁用數據源
`ping` 測試數據源連通性

`datasource add` 主要選項（`update` 時名稱由參數指定，其餘選項相同）：

選項	說明
<code>--name=</code>	數據源名稱（必填），支援字母、數字、點號、斜線、中橫線、下橫線，最長128字元
<code>--db-type=</code>	資料庫驅動供應商（必填），可選：Oracle、MySQL8、MySQL5、PostgreSQL、DM（達夢）、Kingbase（人大金倉）、GaussDB（高斯）、HighGo（瀚高）、Vastbase（海量）、Other（其他）
<code>--resource-type=</code>	資源類型（必填），可選：javax.sql.DataSource、javax.sql.XADataSource、javax.sql.ConnectionPoolDataSource（預設 javax.sql.DataSource）
<code>--driver-class=</code>	資料庫驅動類名（必填），最長128字元
<code>--driver-path=</code>	資料庫驅動 jar 在伺服器上的絕對路徑，如 /opt/drivers/mysql-connector.jar
<code>--status=</code>	創建後是否立即啟用（預設 true）
<code>--class-level=</code>	密級標識（2=秘密、3=機密、4=絕密，預設 2）
<code>--host=</code>	資料庫主機名稱或 IP，最長255字元

--port=	資料庫埠（必填，1-65535）
--database-name=	資料庫名稱或 Schema，最長64字元
--username=	連接用戶名（必填），最長255字元
--password=	連接密碼（必填），最長255字元
--url=	JDBC URL（必填），必須以 jdbc: 開頭，最長500字元
--custom-props=	自訂連接屬性，格式：key1=value1,key2=value2
--initial-connections=	初始連接數（預設 10，範圍 0-10000）
--max-connections=	最大連接數（預設 20，範圍 1-10000）
--min-idle=	最小空閒連接數（預設 4，範圍 0-10000）
--max-idle=	最大空閒連接數（預設 10，範圍 1-10000）
--max-wait-ms=	最大等待時間 ms，-1 表示無限等待（預設 15000）
--idle-timeout-ms=	空閒超時時間 ms（預設 60000，範圍 1-86400000）
--idle-check-period-ms=	空閒檢測週期 ms，0 表示停用檢測（預設 60000）
--max-lifetime-ms=	最大存活時間 ms，0 表示不限制（預設 900000）
--validation-query=	驗證 SQL，建議使用唯讀 SELECT 語句（最長255字元，禁止危險操作關鍵字）
--test-on-borrow=	取得連接時校驗（預設 true）
--test-on-return=	歸還連接時校驗（預設 false）
--test-while-idle=	空閒狀態時校驗（預設 false）
--test-on-connect=	建立連接時校驗（預設 false）
--validation-interval-ms=	連接校驗間隔 ms（預設 30000）
--connection-auto-reclaim=	連接自動回收（預設 false）
--leak-recover=	洩漏連接回收（預設 false）
--leak-detection-threshold-seconds=	洩漏判定超時秒（預設 30，範圍 1-1800）

--auto-commit=	自動提交事務 (預設 true)
--log-abandoned=	記錄洩漏日誌 (預設 false)
--sql-trace-enabled=	慢語句跟蹤 (預設 false)
--slow-sql-ms=	慢 SQL 閾值 ms (預設 2000, 範圍 1-300000)
--failed-sql-alert-enabled=	記錄失敗 SQL (預設 false)

另有 `--max-check-connections`、`--remove-abandoned-on-maintenance`、`--abandon-when-percentage-full`、`--ha-probe-enabled` 等選項，可透過 `datasource add --help` 查看完整說明。

10.3.1.2 示例

示例1 創建數據源

```
admin@localhost> datasource add --name=test-ds --db-type=Other --
resource-type=javax.sql.DataSource --driver-class=org.h2.Driver --
driver-path=D:/aas-h2.jar --port=9092 --username=[YOURUSER] --
password=[YOURPASSWORD] --url=jdbc:h2:mem:testdb
```

數據源創建成功，ID: 3

示例2 列出數據源

```
admin@localhost> datasource list
| 數據源名 | 資源類型 | 密級 | 驅動類名 | 狀態 | 連通性 |
+-----+-----+-----+-----+-----+-----+
| test-ds | javax.sql.DataSource | 秘密 | org.h2.Driver | 已啟用 | 已連通 |
第 1 / 1 頁, 共 1 條
```

示例3 查看數據源詳情

```
admin@localhost> datasource get test-ds
```

【基本配置】

ID: 6

名稱: test-ds

數據庫類型：其他

資源類型： javax.sql.DataSource

驅動類名： org.h2.Driver

驅動路徑： D:/aas-h2.jar

狀態： 開啟

密級： 秘密

主機名：

埠： 9092

數據庫名：

用戶名： sa

密碼： *****

URL： jdbc:h2:mem:testdb

自定義屬性：

【連接池配置】

初始連接數： 10

最大連接數： 20

最小空閒連接數： 4

最大空閒連接數： 10

最大等待時間： 15000 毫秒

空閒超時時間： 60000 毫秒

空閒檢測週期： 60000 毫秒

最大存活時間： 900000 毫秒

單次檢測連接數： 5

【連接驗證配置】

驗證 SQL：

校驗超時時間： 3 秒

獲取連接時校驗： 開啟

歸還連接時校驗： 關閉

空閒狀態時校驗： 關閉

建立連接時校驗： 關閉

連接校驗間隔： 30000 毫秒

【高級配置】

連接自動回收： 關閉

洩漏連接回收：關閉
 維護時回收洩漏連接：關閉
 洩漏判定超時：30 秒
 連接滿閾值比例：10 %
 自動提交事務：開啟
 記錄洩漏日誌：關閉

【監控與診斷】

慢語句跟蹤：關閉
 慢 SQL 閾值：2000 毫秒
 記錄失敗 SQL：關閉
 連接喚醒池：關閉

示例4 測試數據源連通性

```
admin@localhost> datasource ping test-ds
連接成功：连接成功，耗時 1ms
```

示例5 禁用/啟用數據源

```
admin@localhost> datasource toggle test-ds --status=false
數據源已關閉。

admin@localhost> datasource toggle test-ds --status=true
數據源已開啟。
```

示例6 刪除數據源

處於啟用狀態的數據源不允許直接刪除，需先禁用：

```
admin@localhost> datasource delete test-ds -f
數據源操作失敗：数据源已启用，请先禁用后再删除

admin@localhost> datasource toggle test-ds --status=false
數據源已關閉。
```

```
admin@localhost> datasource delete test-ds -f
```

數據源刪除成功。

10.4 監控管理

10.4.1 監控配置管理 (monitor-config)

`monitor-config` 子命令用於查看、更新、重置伺服器監控採集配置。

10.4.1.1 概要

```
Usage: monitor-config [-h] [COMMAND]
```

監控配置管理

`-h, --help` 顯示說明資訊

Commands:

`get` 查看監控配置

`update` 更新監控配置

`reset` 恢復監控配置預設值

`monitor-config update` 主要選項：

選項	說明
<code>--monitor-enabled=</code>	監視服務總開關 (true/false)
<code>--mbean-monitor-enabled=</code>	MBean 監控開關 (true/false)
<code>--collect-interval-seconds=</code>	監控採集週期 (秒) , 範圍 10~3600, 預設 60
<code>--jvm-monitor-enabled=</code>	JVM 監控開關 (true/false)
<code>--thread-pool-monitor-enabled=</code>	執行緒池監控開關 (true/false)
<code>--datasource-monitor-enabled=</code>	數據源監控開關 (true/false)
<code>--http-monitor-enabled=</code>	HTTP 通道監控開關 (true/false)
<code>--session-monitor-enabled=</code>	應用會話監控開關 (true/false)
<code>--system-resource-monitor-enabled=</code>	系統資源監控開關 (true/false)
<code>--fullgc-monitor-enabled=</code>	FullGC 監控開關 (true/false)
<code>--cpu-monitor-enabled=</code>	CPU 監控開關 (true/false)

--memory-monitor-enabled=	記憶體監控開關 (true/false)
--max-data-points=	監控數據總量, 範圍 1~2147483647
--replay-sample-interval=	回放取樣間隔 (秒) , 範圍 1~86400

10.4.1.2 示例

示例1 查看監控配置

```
admin@localhost> monitor-config get
```

監控配置：

監視服務：關閉

監視 MBean：關閉

監控採集週期 (秒) : 60

JVM 監控：關閉

執行緒池監控：關閉

數據源監控：關閉

HTTP 通道監控：關閉

應用會話監控：關閉

系統資源監控：關閉

FullGC 監控：關閉

CPU 監控：關閉

記憶體監控：關閉

監控數據總量：11000

回放取樣間隔 (秒) : 30

示例2 開啟監控並調整採集週期

```
admin@localhost> monitor-config update --monitor-enabled=true --
collect-interval-seconds=120
```

監控配置已更新。

示例3 恢復監控配置預設值

```
admin@localhost> monitor-config reset
```

監控配置已恢復默認值。

10.4.2 告警配置管理 (alert-config)

`alert-config` 子命令用於查看、更新、重置告警配置，包括告警開關、通知方式（郵件/SNMP/告警平台）及各項閾值。

10.4.2.1 概要

Usage: `alert-config [-h] [COMMAND]`

告警配置管理

`-h, --help` 顯示說明資訊

Commands:

`get` 查看告警配置
`update` 更新告警配置
`reset` 恢復告警配置預設值

`alert-config update` 主要選項:

分類	選項	說明
告警開關	<code>--enabled=</code>	啟用告警 (true/false)
	<code>--interval-minutes=</code>	告警間隔 (分鐘)，範圍 1~527040
郵件通知	<code>--email-enabled=</code>	啟用郵件通知 (true/false)
	<code>--email-recipients=</code>	接收郵件，多個信箱用英文分號隔開
	<code>--email-smtp-host=</code>	發送郵件伺服器位址，如 smtp.example.com
	<code>--email-smtp-ssl=</code>	啟用郵件 SSL (true/false)
	<code>--email-smtp-port=</code>	發送郵件伺服器埠，範圍 1~65535
	<code>--email-smtp-user=</code>	發送信箱位址
	<code>--email-smtp-password=</code>	發送信箱密碼，留空表示保留原值
SNMP 通知	<code>--snmp-enabled=</code>	啟用 SNMP 通知 (true/false)
	<code>--snmp-version=</code>	SNMP 版本: v2c 或 v3
	<code>--snmp-community=</code>	SNMP 團體名 (v2c)
	<code>--snmp-address=</code>	SNMP 通知目標位址，如 172.24.4.108:162
	<code>--snmp-security=</code>	SNMP 安全級別 (v3)：無、認證不加密、認證加密

	--snmp-username=	SNMP 用戶名 (v3)
	--snmp-auth-method=	SNMP 認證方式 (v3) : MD5 或 SHA
	--snmp-auth-password=	SNMP 認證密碼 (v3) , 留空表示保留原值
	--snmp-cipher=	SNMP 加密演算法 (v3) : DES、3DES、AES128、AES192、AES256
	--snmp-priv-password=	SNMP 加密密碼 (v3) , 留空表示保留原值
告警平台	--platform-enabled=	啟用告警平台通知 (true/false)
	--platform-level=	告警級別, 如 警告、嚴重
	--platform-url=	告警平台位址, 須為 http:// 或 https:// 開頭
閾值	--cpu-threshold= / --cpu-consecutive=	CPU 使用率告警閾值 / 連續發生次數 (不小於 1)
	--memory-threshold= / --memory-consecutive=	記憶體使用率告警閾值 / 連續發生次數
	--fullgc-count-threshold= / --fullgc-count-consecutive=	FullGC 次數告警閾值 / 連續發生次數 (固定為 1)
	--fullgc-duration-threshold= / --fullgc-duration-consecutive=	FullGC 時間告警閾值 (毫秒) / 連續發生次數
	--http-timeout-threshold= / --http-timeout-consecutive=	HTTP 超時請求告警閾值 / 連續發生次數
	--http-waiting-threshold= / --http-waiting-consecutive=	HTTP 等待任務數告警閾值 / 連續發生次數
	--http-error-threshold= / --http-error-consecutive=	HTTP 錯誤數告警閾值 / 連續發生次數
	--jdbc-waiting-threshold= / --jdbc-waiting-consecutive=	JDBC 等待請求數告警閾值 / 連續發生次數
	--jdbc-leak-threshold= / --jdbc-leak-consecutive=	JDBC 洩漏連接數告警閾值 / 連續發生次數

10.4.2.2 示例

示例1 查看告警配置

```
admin@localhost> alert-config get
```

告警配置：

告警開關：關閉

告警間隔（分鐘）：30

郵件通知

郵件通知開關：關閉

接收郵件：

發送郵件伺服器：

郵件 SSL：開啟

發送郵件伺服器埠：

發送郵箱：

發送郵箱密碼：

SNMP 通知

SNMP 通知開關：關閉

SNMP 版本：v2c

SNMP 團體名：public

SNMP 位址：

SNMP 安全級別：无

SNMP 用戶名：

SNMP 認證方式：SHA

SNMP 認證密碼：

SNMP 加密演算法：AES128

SNMP 加密密碼：

SNMP engineID: 0x8000137001AC1806A0FB9B64B9

告警平台通知

告警平台通知開關：關閉

告警級別：警告

告警平台位址：

告警指標

指標	閾值	單位	連續次數
+	+	+	+

CPU使用率	80.0	%	3	
内存使用率	80.0	%	3	
FullGC次数	5.0	次/分钟	1	
FullGC时间	1000.0	毫秒	2	
HTTP超时请求	10.0	-	2	
HTTP等待任务数	10.0	-	3	
HTTP错误数	5.0	-	2	
JDBC等待请求数	10.0	-	3	
JDBC泄露连接数	1.0	-	2	

說明：告警配置中部分由服務端返回的值（安全級別、告警級別、指標名稱等）目前無論語言設定為何均以簡體中文顯示，以上為實際擷取輸出。

示例2 開啟告警並配置郵件通知

```
admin@localhost> alert-config update --enabled=true --email-enabled=true --email-recipients=ops@example.com --email-smtp-host=smtp.example.com --email-smtp-port=465 --email-smtp-user=alerts@example.com --email-smtp-password=*****
```

告警配置已更新。

示例3 調整 CPU 告警閾值

```
admin@localhost> alert-config update --cpu-threshold=90 --cpu-consecutive=5
```

告警配置已更新。

示例4 恢復告警配置預設值

```
admin@localhost> alert-config reset
```

告警配置已恢復預設值。

10.4.3 SNMP 代理配置管理 (snmp-config)

`snmp-config` 子命令用於查看、更新、重置 AAS 內建 SNMP 代理配置，供上層網管系統透過 SNMP 方式採集監控數據。

10.4.3.1 概要

Usage: snmp-config [-h] [COMMAND]

SNMP 代理配置管理

-h, --help 顯示說明資訊

Commands:

get 查看 SNMP 代理配置
 update 更新 SNMP 代理配置
 reset 恢復 SNMP 配置預設值

snmp-config update 選項:

選項	說明
--enabled=	啟用 SNMP 服務 (true/false)
--host=	綁定位址, 如 0.0.0.0、192.168.1.1, 最長 128 個字元
--port=	綁定埠, 範圍 1~65535, 修改時服務端會做埠佔用探測
--protocol=	傳輸協定: udp 或 tcp
--version=	SNMP 版本: v2c 或 v3
--community=	SNMP 團體名 (v2c)
--security=	SNMP 安全級別 (v3) : 無、認證不加密、認證加密
--username=	SNMP 用戶名 (v3)
--auth-method=	SNMP 認證方式 (v3) : MD5 或 SHA
--auth-password=	SNMP 認證密碼 (v3) , 留空表示保留原值
--cipher=	SNMP 加密演算法 (v3) : DES、3DES、AES128、AES192、AES256
--priv-password=	SNMP 加密密碼 (v3) , 留空表示保留原值

10.4.3.2 示例

示例1 查看 SNMP 代理配置

```
admin@localhost> snmp-config get
```

SNMP 代理配置:

SNMP 服務開關: 關閉

綁定位址: 0.0.0.0

綁定埠: 161

```

傳輸協定: udp
SNMP 版本: v2c
SNMP 團體名: public

```

示例2 啟用 SNMP 代理 (v2c)

```

admin@localhost> snmp-config update --enabled=true --host=0.0.0.0 --
port=161 --protocol=udp --version=v2c --community=public
SNMP 配置已更新。

```

示例3 恢復 SNMP 配置預設值

```

admin@localhost> snmp-config reset
SNMP 配置已恢復預設值。

```

10.4.4 一鍵巡檢管理 (inspection)

`inspection` 子命令用於一鍵巡檢伺服器的執行狀態，生成巡檢報告，並可查看、下載、刪除報告及查看巡檢系統快照。

10.4.4.1 概要

```
Usage: inspection [-h] [COMMAND]
```

一鍵巡檢管理

`-h, --help` 顯示說明資訊

Commands:

<code>list</code>	列出巡檢報告
<code>get</code>	查看巡檢報告詳情
<code>run</code>	發起一鍵巡檢
<code>status</code>	查詢巡檢報告執行狀態
<code>summary</code>	查看巡檢報告摘要
<code>download</code>	下載巡檢報告壓縮包 (僅巡檢成功的報告)
<code>delete</code>	刪除巡檢報告
<code>rerun</code>	重新巡檢 (僅已完成或異常的報告)
<code>snapshot</code>	查看或下載巡檢系統快照 (<code>jstack</code> 、 <code>lsof</code> 、 <code>netstat</code> 、 <code>top</code> 、 <code>lsof</code> 僅 Linux)
<code>has-running</code>	查詢是否存在正在執行的巡檢任務

`inspection list` 過濾選項:

選項	說明
<code>--name=</code>	按報告名稱過濾
<code>--status=</code>	按狀態過濾: RUNNING、COMPLETED、TIMEOUT、ABNORMAL (異常)
<code>--start-time=</code>	開始時間, 格式: yyyy-MM-dd HH:mm:ss
<code>--end-time=</code>	結束時間, 格式: yyyy-MM-dd HH:mm:ss

`inspection run` 選項:

選項	說明
<code>--timeout-seconds=</code>	巡檢超時時間 (秒), 範圍 1~172800, 預設 600

`inspection snapshot` 參數:

參數	說明
	報告 ID, 可用 <code>aascli inspection list</code> 查詢
	快照類型: jstack、lsof、netstat、top (lsof 僅 Linux)
<code>--file=</code>	儲存路徑, 缺省時直接輸出快照內容

10.4.4.2 示例

示例1 查詢是否存在正在執行的巡檢任務

```
admin@localhost> inspection has-running
不存在正在執行的巡檢任務
```

示例2 發起一鍵巡檢

巡檢在背景非同步執行, 提交後可透過報告 ID 查詢進度:

```
admin@localhost> inspection run
巡檢任務已提交 (ID: 3, 名稱: InspectionReport_20260927175359), 正在後台執行
可使用 aascli inspection status 3 查詢執行狀態
```

示例3 查詢巡檢執行狀態

```
admin@localhost> inspection status 3
正在巡檢

admin@localhost> inspection status 3
巡檢成功
```

示例4 列出巡檢報告

```
admin@localhost> inspection list
| ID | 巡檢結果 | 狀態 | 超時時間 (秒) | 日期 |
|-----+-----+-----+-----+-----|
| 3 | InspectionReport_20260927175359 | 巡檢成功 | 600 | 2026-09-27 17:53:59 |
第 1 / 1 頁, 共 1 條
```

示例5 查看巡檢報告摘要

```
admin@localhost> inspection summary 3
CPU 負載 80% 以上的實例數: 0
記憶體使用率 80% 以上的實例數: 0
堆使用率 80% 以上的實例數: 0
硬碟使用率 80% 以上的實例數: 1
```

示例6 下載巡檢報告壓縮包

```
admin@localhost> inspection download 3 --file=D:/aas-tw-inspection.zip
下載成功: D:\aas-tw-inspection.zip
```

示例7 重新巡檢

僅已完成或異常的報告可重新巡檢:

```
admin@localhost> inspection rerun 3
```

示例8 查看巡檢系統快照

```
admin@localhost> inspection snapshot 3 jstack
```

示例9 刪除巡檢報告

```
admin@localhost> inspection delete 3 -f
巡檢報告已刪除。
```

10.4.5 監控快照管理 (snapshot)

`snapshot` 子命令用於管理監控快照，支援手動創建、下載、刪除快照，以及配置快照自動生成策略。

10.4.5.1 概要

Usage: `snapshot [-h] [COMMAND]`

監控快照管理（手動創建、下載、刪除、自動配置）

`-h, --help` 顯示說明資訊

Commands:

<code>list</code>	列出監控快照
<code>get</code>	按名稱精確查詢快照詳情
<code>create</code>	手動創建監控快照（非同步生成）
<code>delete</code>	刪除監控快照
<code>download</code>	下載監控快照壓縮包（僅生成完成的快照）
<code>config</code>	快照自動生成配置

`snapshot list` 過濾選項：

選項	說明
<code>--name=</code>	按快照名稱過濾
<code>--status=</code>	按狀態過濾：pending、running、done、failed
<code>--start-time=</code>	開始時間，格式：yyyy-MM-dd HH:mm:ss
<code>--end-time=</code>	結束時間，格式：yyyy-MM-dd HH:mm:ss

`snapshot create` 選項：

選項	說明
<code>--contents=</code>	快照內容（必填），逗號分隔：accessLog、gcLog、systemConfig、systemLog、

<a,b,...>	hostMonitor、instanceMetrics、coreDump、threadDump、heapDump
--thread-stack-count=	執行緒堆疊擷取次數，範圍 1~10（含 threadDump 時建議指定）
--heapdump-mode=	Heapdump 模式：all 或 live（含 heapDump 時建議指定）

snapshot config 子命令：

Usage: snapshot config [-h] [COMMAND]

快照自動生成配置

-h, --help 顯示說明資訊

Commands:

get 查看快照自動生成配置
 update 更新快照自動生成配置
 reset 恢復快照自動生成配置預設值

snapshot config update 主要選項：

分類	選項	說明
基本設定	--auto-enabled=	啟用自動快照 (true/false)
	--max-snapshots=	最多保存快照數，範圍 1~1000
	--interval-minutes=	生成間隔（分鐘），範圍 1~1440
快照內容	--content-access-log=	採集存取日誌 (true/false)
	--content-gc-log=	採集 GC 日誌 (true/false)
	--content-system-config=	採集系統配置 (true/false)
	--content-system-log=	採集系統日誌 (true/false)
	--content-host-monitor=	採集主機監控數據 (true/false)
	--content-instance-metrics=	採集實例監視量 (true/false)
	--content-core-dump=	採集系統 Coredump (true/false)
	--content-thread-dump=	採集執行緒堆疊 (true/false)

	--thread-stack-count=	執行緒堆疊擷取次數，範圍 1~10
	--content-heap-dump=	採集 Heapdump (true/false)
	--heapdump-mode=	Heapdump 模式: all 或 live
觸發規則	--rule-cpu-threshold= / --rule-cpu-occurrences=	CPU 使用率閾值 (%) / 連續發生次數
	--rule-memory-threshold= / --rule-memory-occurrences=	記憶體使用率閾值 (%) / 連續發生次數
	--rule-fullgc-count-threshold= / --rule-fullgc-count-occurrences=	FullGC 次數閾值 / 連續發生次數
	--rule-fullgc-time-threshold= / --rule-fullgc-time-occurrences=	FullGC 時間閾值 (毫秒) / 連續發生次數
	--rule-http-timeout-threshold= / --rule-http-timeout-occurrences=	HTTP 慢請求數閾值 / 連續發生次數
	--rule-http-wait-threshold= / --rule-http-wait-occurrences=	HTTP 等待任務數閾值 / 連續發生次數
	--rule-http-error-threshold= / --rule-http-error-occurrences=	HTTP 錯誤數閾值 / 連續發生次數
	--rule-jdbc-wait-threshold= / --rule-jdbc-wait-occurrences=	JDBC 等待請求數閾值 / 連續發生次數
	--rule-jdbc-leak-threshold= / --rule-jdbc-leak-occurrences=	JDBC 疑似洩漏連接數閾值 / 連續發生次數

10.4.5.2 示例

示例1 手動創建監控快照

快照為非同步生成，提交後可透過列表查看進度：

```
admin@localhost> snapshot create --contents=systemLog,threadDump --
thread-stack-count=1
快照任務已提交，正在背景生成。
可使用 aascli snapshot list 查看生成進度。
```

示例2 列出監控快照

```
admin@localhost> snapshot list
```

ID	名稱	類型	狀態	內容
3	snapshot-20260927-175544-manual	手動	已完成	systemLog,threadDump

2026-09-27 17:55:44

第 1 / 1 頁, 共 1 條

示例3 按名稱精確查詢快照詳情

```
admin@localhost> snapshot get snapshot-20260927-175544-manual
```

示例4 下載監控快照壓縮包

```
admin@localhost> snapshot download 3 --file=D:/aas-tw-snapshot.zip
快照下載成功: D:\aas-tw-snapshot.zip
```

示例5 刪除監控快照

```
admin@localhost> snapshot delete 3 -f
快照已刪除。
```

示例6 查看快照自動生成配置

```
admin@localhost> snapshot config get
```

自動快照生成配置：

自動快照

自動快照開關：關閉

最多保存快照數：10

生成間隔（分鐘）：5

快照內容

存取日誌：開啟

GC 日誌：關閉

系統配置：關閉

```

系統日誌：開啟
主機監控數據：關閉
實例監視量：關閉
系統 Coredump：關閉
執行緒堆疊：開啟
執行緒堆疊擷取次數：3
Heapdump：開啟
Heapdump 模式：live

```

觸發規則

```

CPU 使用率閾值 (%)：80.0
CPU 連續發生次數：3
記憶體使用率閾值 (%)：80.0
記憶體連續發生次數：3
FullGC 次數閾值：5
FullGC 次數連續發生次數：1
FullGC 時間閾值 (毫秒)：1000
FullGC 時間連續發生次數：2
HTTP 慢請求數閾值：1
HTTP 慢請求數連續發生次數：2
HTTP 等待任務數閾值：1
HTTP 等待任務數連續發生次數：3
HTTP 錯誤數閾值：3
HTTP 錯誤數連續發生次數：2
JDBC 等待請求數閾值：1
JDBC 等待請求數連續發生次數：3
JDBC 疑似洩漏連接數閾值：1
JDBC 疑似洩漏連接數連續發生次數：2

```

示例7 更新快照自動生成配置

```

admin@localhost> snapshot config update --auto-enabled=true --
interval-minutes=30 --max-snapshots=20

```

示例8 恢復快照自動生成配置預設值

```
admin@localhost> snapshot config reset
```

10.5 日誌服務

10.5.1 系統日誌配置 (log-config system)

`log-config system` 子命令用於查看、更新伺服器系統日誌配置。

10.5.1.1 概要

Usage: `log-config system [-h] [COMMAND]`

系統日誌配置

`-h, --help` 顯示說明資訊

Commands:

`get` 查看系統日誌配置

`update` 更新系統日誌配置

`log-config system update` 主要選項：

分類	選項	說明
基本配置	<code>--global-level=</code>	全域日誌級別，預設 INFO，可選：SEVERE/WARNING/INFO/CONFIG/FINE/FINER/FINEST/ALL
	<code>--class-loggers=</code>	自訂類級別配置，為指定包或類設定獨立日誌級別，多個以英文逗號分隔，格式：package.Class=LEVEL
	<code>--console-enabled=</code>	是否在控制台輸出日誌資訊，預設 true
	<code>--file-enabled=</code>	是否將日誌資訊寫入檔案，預設 true
檔案配置	<code>--file-prefix=</code>	日誌檔案名字首標識，預設 apusic.，最長 100 字元
	<code>--file-suffix=</code>	日誌檔案名字尾，預設 .log，最長 100 字元
	<code>--encoding=</code>	日誌檔案字元編碼，預設 UTF-8，可選：UTF-8/GBK
	<code>--file-timestamp-format=</code>	日誌檔案名中的日期格式，預設 yyyy-MM-dd
格式配置	<code>--console-pattern=</code>	控制台日誌輸出格式模板，如 "%d{yyyy-MM-dd HH:mm:ss.SSS} %level [%t] %logger %msg%n"

	--file-pattern=	檔案日誌輸出格式模板，格式同上
輪轉配置	--retention-count=	按天保留日誌檔案，超過保留天數的日誌將自動刪除，預設 30，0 或負數表示不限制
	--max-size=	單個日誌檔案達到該大小時自動輪轉，單位 MB，預設 100，0 或負數表示不限制
	--rotate-by-date=	是否每天生成一個新的日誌檔案，預設 true
壓縮配置	--compress-enabled=	是否開啟日誌自動壓縮歸檔功能，預設 false
	--compress-time=	每天執行日誌壓縮的時間點，預設 02:00，格式 HH:mm
	--compress-dir=	壓縮檔案的儲存路徑，預設 \${apusic.base}/logs/archive，支援變數如 \${apusic.base}

10.5.1.2 示例

示例1 查看系統日誌配置

```
admin@localhost> log-config system get
```

【基本配置】

全局日誌級別：FINE

自定義類級別配置：

控制台日誌輸出：開啟

日誌文件輸出：開啟

【文件配置】

日誌文件前綴：apusic.

日誌文件後綴：.log

日誌文件編碼：UTF-8

日誌文件名日期格式：yyyy-MM-dd

【格式配置】

控制台日誌格式：%d{yyyy-MM-dd HH:mm:ss.SSS} %level [%t] %logger %msg%n

文件日誌格式：%d{yyyy-MM-dd HH:mm:ss.SSS} %level [%t] %logger %msg%n

【輪轉配置】

保留天數：30
按大小輪轉：0 MB
按天輪轉：開啟

【壓縮配置】

日誌壓縮：關閉

示例2 調整全域日誌級別

```
admin@localhost> log-config system update --global-level=INFO
```

日誌配置更新成功。

示例3 為指定包設定獨立日誌級別

```
admin@localhost> log-config system update --class-loggers=com.example.biz=DEBUG,com.example.dao=INFO
```

日誌配置更新成功。

10.5.2 存取日誌配置 (log-config access)

`log-config access` 子命令用於查看、更新 HTTP 存取日誌 (Access Log) 配置。

10.5.2.1 概要

Usage: `log-config access [-h] [COMMAND]`

存取日誌配置

`-h, --help` 顯示說明資訊

Commands:

`get` 查看存取日誌配置

`update` 更新存取日誌配置

`log-config access update` 主要選項：

分類	選項	說明
基礎配置	<code>--enabled=</code>	啟用存取日誌 (true/false)
	<code>--virtual-hosts=</code>	應用虛擬主機，多個以英文逗號分隔，可用 <code>aascli host list</code> 查詢候選項

	--directory=	日誌檔案路徑，支援變數如 \${apusic.base}
	--encoding=	日誌檔案編碼，可選：UTF-8/GBK，預設 UTF-8
	--buffer-enabled=	寫入緩衝區 (true/false)
	--buffer-size=	緩衝區大小，單位位元組，預設 32768
	--file-date-format=	日誌檔案名日期格式，如 .yyyy-MM-dd
	--max-days=	保留天數，預設 30，0 或負數表示不限制
格式配置	--custom-format-enabled=	啟用自訂格式 (true/false)
	--log-content=	日誌內容，多個以英文逗號分隔，可選：time、remoteAddr、method、protocol、status、uri、bytesSent、sessionId、threadName、serverName、localPort
	--pattern=	自訂存取日誌的內容記錄格式，使用佔位符如 %a(遠端IP)、%t(時間) 等，如 "%h %l %u %t \"%r\" %s %b"
	--date-format=	日期格式，控制時間欄位輸出格式 (%{xxx}t 中的 xxx) ，預設 dd/MM/yyyy:HH:mm:ss
	--request-headers=	記錄請求頭，多個以英文逗號分隔
	--response-headers=	記錄回應頭，多個以英文逗號分隔
	--cookies=	記錄 Cookie，多個以英文逗號分隔
	--session-attrs=	記錄 Session，多個以英文逗號分隔
輪轉與壓縮	--rotate-by-date=	按天輪轉 (true/false)
	--max-size=	按大小輪轉，單位 MB，0 表示不限制
	--compress-enabled=	日誌壓縮 (true/false)
	--compress-	日誌壓縮時間，格式 HH:mm，預設 00:00

	time=	
	--compress-dir=	日誌壓縮路徑，支援變數如 \${apusic.base}/logs/archive

10.5.2.2 示例

示例1 啟用存取日誌

```
admin@localhost> log-config access update --enabled=true
```

日誌配置更新成功。

示例2 查看存取日誌配置

```
admin@localhost> log-config access get
```

【基礎配置】

啟用訪問日誌：開啟

應用虛擬主機：localhost

日誌文件路徑：\${apusic.base}/logs/localhost_access_log

日誌文件編碼：UTF-8

寫入緩衝區：開啟

緩衝區大小：32768

日誌文件名日期格式：.yyyy-MM-dd

保留天數：30

【日誌格式配置】

啟用自定義格式：

日誌內容：time, remoteAddr, method, uri, status

日期格式：dd/MM/yyyy:HH:mm:ss

記錄請求頭：

記錄響應頭：

記錄 Cookie：

記錄 Session：

【日誌輪轉配置】

按天輪轉：開啟

按大小輪轉：30 MB

【日誌壓縮配置】**日誌壓縮：關閉****示例3 關閉存取日誌**

```
admin@localhost> log-config access update --enabled=false
```

日誌配置更新成功。

10.6 用戶管理

10.6.1 用戶管理 (user)

`user` 子命令用於管控制台用戶管理，包括用戶的創建、查看、刪除及權限查看。用戶的角色分配、權限配置由安全管理員負責，系統管理員可查看用戶權限配置。

10.6.1.1 概要

```
Usage: user [-h] [COMMAND]
```

用戶管理

`-h, --help` 顯示說明資訊

Commands:

<code>list</code>	列出用戶
<code>get</code>	查看用戶詳情
<code>add</code>	創建用戶
<code>delete</code>	刪除用戶
<code>permission</code>	用戶權限配置

`user add` 選項：

選項	說明
<code>--username=</code>	用戶名（必填）。長度 1-64 字元，僅支援字母、數字、底線、短橫線
<code>--password=</code>	登入密碼。腳本場景可指定，未指定時互動式輸入並確認
<code>--status=</code>	用戶狀態：0=禁用，1=正常，2=鎖定。預設：1
<code>--email=</code>	信箱位址。有效信箱格式，最長 255 字元
<code>--phone=</code>	手機號碼。中國手機號 1[3-9]XXXXXXXXXX
<code>--description=</code>	說明。最長 255 字元

10.6.1.2 示例

示例1 創建用戶

```
admin@localhost> user add --username=testuser --password=Test@12345 --description=CLI-test-user
```

用戶創建成功。

注意：新創建的用戶尚未分配角色，此時登入會提示 登錄失敗：用戶未授權，請聯系管理員。需由安全管理員為用戶分配角色後，用戶方可登入使用。

示例2 列出用戶

```
admin@localhost> user list
```

用戶名	狀態	角色	密級	過期時間	特權標識
testuser	正常		秘密	永久	否
admin	正常	sysadmin	絕密	永久	是
secure	正常	security	絕密	永久	是
audit	正常	auditor	絕密	永久	是

第 1 / 1 頁, 共 4 條

示例3 查看用戶詳情

```
admin@localhost> user get testuser
```

ID: 8

用戶名: testuser

狀態: 正常

密級: 秘密

特權標識: 否

角色:

郵箱:

手機號:

OTP: 否

過期時間: 永久

創建時間: 2026-09-27 17:46:14

描述: CLI-test-user

示例4 查看用戶權限配置

```
admin@localhost> user permission get testuser
```

用戶權限配置：

用戶名：testuser

狀態：正常

權限分配角色：

過期時間：永久

允許訪問 IP：

開始訪問時間：

結束訪問時間：

數據密級：秘密

手機號：

郵箱：

描述：CLI-test-user

示例5 刪除用戶

```
admin@localhost> user delete testuser -f
```

用戶刪除成功。

10.7 安全服務

10.7.1 安全域管理 (security-domain)

`security-domain` 子命令用於管理安全域 (Realm)，為應用提供認證與授權資料來源，支援數據源、用戶資料庫、JAAS、JNDI (LDAP 等) 四種類型。

10.7.1.1 概要

```
Usage: security-domain [-h] [COMMAND]
```

安全域管理

`-h, --help` 顯示說明資訊

Commands:

<code>list</code>	列出安全域
<code>get</code>	查看安全域詳情
<code>add</code>	創建安全域
<code>update</code>	更新安全域
<code>delete</code>	刪除安全域

`security-domain add` 基本選項 (`update` 時名稱由參數指定, 類型不提供則保持不變) :

選項	說明
<code>--name=</code>	安全域名稱 (必填), 支援字母、數字、底線、中橫線, 最長128字元
<code>--type=</code>	安全域類型 (必填), 可選: DataSourceRealm (數據源)、UserDatabaseRealm (用戶資料庫)、JAASRealm (JAAS)、JNDIRealm (JNDI)
<code>--description=</code>	安全域描述, 最長500字元, 不能包含 < 和 >
<code>--password-hash-algorithm=</code>	密碼比較演算法, 可選空值 (明文) 或 MD5、SHA-1、SHA-256、SHA-512
<code>--lock-enabled=</code>	是否啟用登入失敗鎖定 (true/false, 預設 false)
<code>--max-failed-attempts=</code>	允許錯誤次數, 啟用鎖定時連續失敗多少次後鎖定帳號 (預設 5)
<code>--lock-timeout=</code>	鎖定超時時間 (秒), 帳號被鎖定後多久自動解鎖 (預設 300)

各類型專屬選項:

類型	主要選項
DataSourceRealm	<code>--data-source-jndi</code> (數據源 JNDI 名稱, 必填)、 <code>--user-table</code> 、 <code>--username-field</code> 、 <code>--password-field</code> 、 <code>--role-table</code> 、 <code>--role-field</code>
UserDatabaseRealm	<code>--resource-name</code> (apusic.conf 中已配置的 UserDatabase 資源名, 必填)
JAASRealm	<code>--user-principal-class</code> (必填)、 <code>--role-principal-class</code> (必填)、 <code>--jaas-module</code> (LoginModule 配置, 格式: loginModuleClass@controlFlag@options, 多個模組用分號分隔)
JNDIRealm	<code>--context-factory</code> 、 <code>--jndi-connection-url</code> (必填, 如 ldap://localhost:389)、 <code>--connection-username</code> 、 <code>--connection-password</code> 、 <code>--user-search-base</code> 、 <code>--user-search-filter</code> (如 (uid={0}))、 <code>--role-search-base</code> 、 <code>--role-search-filter</code> (如 (member={0}))、 <code>--user-subtree</code> 、 <code>--role-subtree</code> 等

各類型選項的完整說明可透過 `security-domain add --help` 查看。

10.7.1.2 示例

示例1 創建用戶資料庫類型安全域

```
admin@localhost> security-domain add --name=test-realm --
type=UserDatabaseRealm --resource-name=UserDatabase
```

安全域創建成功。

示例2 列出安全域

```
admin@localhost> security-domain list
```

安全域名稱	類型	綁定應用	失敗鎖定
test-realm	UserDatabaseRealm	--	關閉

第 1 / 1 頁, 共 1 條

示例3 查看安全域詳情

```
admin@localhost> security-domain get test-realm
```

【基本配置】

ID: 3

安全域名稱: test-realm

安全域類型: UserDatabaseRealm

描述:

密碼比較演算法:

【鎖定機制】

啟用鎖定: 關閉

【UserDatabase 配置】

資源名字: UserDatabase

示例4 更新安全域

```
admin@localhost> security-domain update test-realm --lock-enabled=true
--max-failed-attempts=5 --lock-timeout=300
```

示例5 刪除安全域

```
admin@localhost> security-domain delete test-realm -f
```

安全域刪除成功。

示例6 應用綁定安全域

創建安全域後，可透過 `app update` 為應用綁定：

```
admin@localhost> app update testapp --security-domain=test-realm --
reload
應用配置更新成功。
應用重新載入成功。
```

10.8 系統與賬戶

10.8.1 用戶設定 (usersettings)

`usersettings` 子命令用於查看或修改當前登入用戶的個人設定，包括命令列分頁、語言、歷史記錄條數及個人資訊。

10.8.1.1 概要

```
Usage: usersettings [-hl] [--no-paging] [--history-size=<historySize>]
[--lang=<language>] [--page-size=<pageSize>]
[COMMAND]
```

用戶設置

<code>-h, --help</code>	顯示說明資訊
<code>--history-size=<historySize></code>	歷史命令記錄條數
<code>-l, --list</code>	查看當前設置
<code>--lang=<language></code>	語言: en / zh / zh_TW
<code>--no-paging</code>	關閉分頁輸出
<code>--page-size=<pageSize></code>	每頁記錄數

Commands:

`profile` 查看或修改個人信息

```
Usage: usersettings profile [-h] [--description=<description>] [--
email=<email>] [--phone=<phone>]
```

查看或修改個人信息

<code>--description=<description></code>	說明
<code>--email=<email></code>	郵箱地址
<code>-h, --help</code>	顯示說明資訊
<code>--phone=<phone></code>	手機號碼

10.8.1.2 選項

```
-l, --list
```

查看當前設定。

```
--lang=<language>
```

設定命令列介面語言：en / zh / zh_TW。重新登入 aascli 後生效。

```
--page-size=<pageSize>
```

設定列表查詢每頁顯示的記錄數。

```
--no-paging
```

關閉命令列分頁輸出，列表一次輸出全部記錄。

```
--history-size=<historySize>
```

設定歷史命令記錄條數。

10.8.1.3 示例

示例1 查看當前設定

```
admin@localhost> usersettings -l
```

當前配置：

分頁輸出：開啟

每頁條數：20

語言：zh_TW

歷史命令記錄條數：1,000

示例2 修改每頁記錄數與歷史記錄條數

```
admin@localhost> usersettings --page-size=50 --history-size=2000
```

示例3 查看個人資訊

```
admin@localhost> usersettings profile
```

用戶名：admin

手機：-

郵箱： -
說明： -

示例4 修改個人資訊

```
admin@localhost> usersettings profile --email=admin@example.com --
phone=13800000000
```

10.8.2 修改當前登入用戶的密碼 (changepassword)

`changepassword` 子命令用於修改當前登入用戶自己的密碼。

10.8.2.1 概要

```
Usage: changepassword [-h] [--new-password=<newPassword>] [--old-
password=<oldPassword>]
```

修改當前登入用戶的密碼

`-h, --help`

顯示說明資訊

`--new-password=<newPassword>`

新密碼; 未提供時互動式輸入

`--old-password=<oldPassword>`

舊密碼; 未提供時互動式輸入

10.8.2.2 選項

```
--old-password=<oldPassword>
```

當前登入用戶的舊密碼。未提供時互動式輸入（輸入不回顯）。

```
--new-password=<newPassword>
```

新密碼。未提供時互動式輸入並二次確認（輸入不回顯）。新密碼需滿足密碼複雜度要求。

10.8.2.3 示例

示例1 互動式修改密碼

```
admin@localhost> changepassword
舊密碼:
新密碼:
確認新密碼:
密碼修改成功, 請重新 login 登錄。
```

密碼修改成功後當前會話失效，需重新執行 `login` 登入。

示例2 指定舊密碼與新密碼

```
admin@localhost> changepassword --old-password=Old@12345 --new-  
password=New@67890
```

密碼修改成功，請重新 `login` 登錄。

11 安全管理員功能模組

安全管理員使用內建帳號 `secure`（角色 `security`）登入 `aascli`，負責角色管理、用戶授權與安全策略配置。安全管理員登入後可見的命令如下：

```
secure@localhost> help
用戶管理：
    user          用戶管理

角色管理：
    role          角色管理

安全服務：
    security-config 安全配置管理

系統與帳戶：
    usersettings  用戶設置
    changepassword 修改當前登錄用戶的密碼
    logout        退出登錄
    help          顯示可用命令
```

`usersettings`、`changepassword`、`logout`、`help` 為通用帳戶命令，用法與系統管理員一致，參見「系統與帳戶」章節，此處不再重複。

安全管理員與系統管理員的用戶管理職責分工如下：

操作	系統管理員 (admin)	安全管理員 (secure)
創建用戶 (user add)	支援	不支援
列出/查看用戶	支援	支援
刪除用戶 (user delete)	支援	支援
重置用戶密碼 (user reset-password)	不支援	支援
分配/解除用戶角色 (user permission update)	不支援 (僅可查看)	支援

11.1 用戶管理

11.1.1 用戶管理 (user)

`user` 子命令用於用戶管理。安全管理員可查看、刪除用戶，重置用戶密碼，並配置用戶權限（角色綁定、存取控制等）。

11.1.1.1 概要

```
Usage: user [-h] [COMMAND]
```

用戶管理

`-h, --help` 顯示說明資訊

Commands:

<code>list</code>	列出用戶
<code>get</code>	查看用戶詳情
<code>delete</code>	刪除用戶
<code>reset-password</code>	重置用戶密碼
<code>permission</code>	用戶權限配置

11.1.2 列出用戶 (user list)

用法與系統管理員一致，展示用戶名、狀態、角色、密級等資訊。

11.1.2.1 示例

示例1 列出用戶

```
secure@localhost> user list
```

用戶名	狀態	角色	密級	過期時間	特權標識
admin	正常	sysadmin	絕密	永久	是
secure	正常	security	絕密	永久	是
audit	正常	auditor	絕密	永久	是

第 1 / 1 頁, 共 3 條

11.1.3 查看用戶詳情 (user get)

用法與系統管理員一致。

11.1.3.1 示例

示例1 查看用戶詳情

```
secure@localhost> user get admin
ID: 1
用戶名: admin
狀態: 正常
密級: 絕密
特權標識: 是
角色: sysadmin
郵箱:
手機號:
OTP: 否
過期時間: 永久
創建時間: 2026-09-27 11:23:52
描述:
```

11.1.4 重置用戶密碼 (user reset-password)

`user reset-password` 子命令用於重置指定用戶的密碼，用戶忘記密碼或帳號密碼洩漏時使用。

11.1.4.1 概要

```
Usage: user reset-password <ID> [--password=<password>] [-h]
```

重置用戶密碼

<code><id></code>	用戶 ID
<code>--password=<password></code>	新密碼。腳本場景可指定，未指定時互動式輸入並確認
<code>--help</code>	顯示說明資訊

11.1.4.2 選項

```
<ID>
```

用戶 ID，可透過 `user get <用戶名>` 查詢。注意此處參數為用戶 ID 而非用戶名。

```
--password=<password>
```

新密碼，需符合當前密碼策略。未指定時互動式輸入並二次確認（輸入不回顯）。

11.1.4.3 示例

示例1 重置用戶密碼

```
secure@localhost> user reset-password 14 --password=New@67890
```

密碼重置成功。

11.1.5 刪除用戶 (user delete)

用法與系統管理員一致，刪除前建議先解除用戶角色綁定。

11.1.5.1 示例

示例1 刪除用戶

```
secure@localhost> user delete testuser6 -f
```

用戶刪除成功。

11.1.6 用戶權限配置 (user permission)

`user permission` 子命令用於配置用戶權限，包括角色綁定、用戶狀態、帳號有效期、存取 IP 白名單、存取時段、數據密級等，是安全管理員的核心操作。

11.1.6.1 概要

```
Usage: user permission [-h] [COMMAND]
```

用戶權限配置

```
-h, --help 顯示說明資訊
```

Commands:

```
get 查看用戶權限配置
```

```
update 更新用戶權限配置
```

```
Usage: user permission update <USERNAME> [--status=<status>] [--role=
<code>|--unbind-role] [--expire-date=<date>]
[--ip-whitelist=<ips>] [--start-time=<time>] [--end-time=<time>] [--
class-level=<level>] [--phone=<phone>]
[--email=<email>] [--description=<desc>] [-h]
```

更新用戶權限配置

```
<username>
```

用戶名，可用 `aascli user list` 查詢

```
--status=<status>
```

用戶狀態：0=禁用，1=正常，2=鎖定

```
--role=<role>
```

權限分配角色編碼，可透過 ``aascli role`

`list`` 查看可用角色編碼，或透過 ``aascli user`

`permission get <USERNAME>`` 查看

當前用戶的角色編碼

<code>--unbind-role</code>	解除用戶當前綁定的角色，與 <code>--role</code> 互斥
<code>--expire-date=<expireDate></code>	過期時間，格式 <code>yyyy-MM-dd HH:mm:ss</code> ， 為空表示永久有效
<code>--ip-whitelist=<ipWhitelist></code>	允許存取 IP，多個用英文逗號分隔，為空表示不限制
<code>--start-time=<startTime></code>	開始存取時間，格式 <code>yyyy-MM-dd HH:mm:ss</code>
<code>--end-time=<endTime></code>	結束存取時間，格式 <code>yyyy-MM-dd HH:mm:ss</code>
<code>--class-level=<classLevel></code>	數據密級：2=秘密，3=機密，4=絕密
<code>--phone=<phone></code>	手機號碼。取值範圍：中國手機號 1[3-9]XXXXXXXXXX
<code>--email=<email></code>	郵箱地址。取值範圍：有效郵箱格式，最長 255 字元
<code>--description=<description></code>	說明。取值範圍：最長 255 字元
<code>--help</code>	顯示說明資訊

11.1.6.2 選項

```
--status=<status>
```

用戶狀態：0=禁用，1=正常，2=鎖定。禁用後用戶無法登入。

```
--role=<role>
```

權限分配的角色編碼，可透過 `role list` 查看可用角色編碼。用戶只有綁定了角色才能登入管控台或 `aascli`。

```
--unbind-role
```

解除用戶當前綁定的角色，與 `--role` 互斥。解綁後用戶無法登入。

```
--expire-date=<expireDate>
```

帳號過期時間，格式 `yyyy-MM-dd HH:mm:ss`，為空表示永久有效。過期後帳號無法登入。含空格的日期時間需使用引號包裹。

```
--ip-whitelist=<ipWhitelist>
```

允許存取的 IP 白名單，多個用英文逗號分隔，為空表示不限制。僅支援 IPv4、IPv6 位址或 * 萬用字元（如 192.168.1.*），不支援 CIDR 格式（如 192.168.1.0/24）。

```
--start-time=<startTime> / --end-time=<endTime>
```

允許存取的開始/結束時間，格式 yyyy-MM-dd HH:mm:ss，用於限制用戶僅在指定時段內存取。

```
--class-level=<classLevel>
```

用戶數據密級：2=秘密，3=機密，4=絕密。

```
--phone=<phone> / --email=<email> / --description=<description>
```

用戶聯繫方式與說明資訊，與 `usersettings profile` 的區別在於可維護其他用戶的資訊。

11.1.6.3 示例

示例1 為用戶分配角色並設定帳號有效期與數據密級

系統管理員創建用戶後，需由安全管理員分配角色，用戶方可登入：

```
secure@localhost> user permission update testuser6 --role=test-role --
expire-date="2027-12-31 23:59:59" --class-level=3
```

用戶權限配置已保存。

```
secure@localhost> user permission get testuser6
```

用戶權限配置：

用戶名：testuser6

狀態：正常

權限分配角色：test-role

過期時間：2027-12-31 23:59:59

允許訪問 IP：

開始訪問時間：

結束訪問時間：

數據密級：機密

手機號：

郵箱：

描述：

示例2 設定用戶存取 IP 白名單

僅支援 IPv4、IPv6 位址或 * 萬用字元：

```
secure@localhost> user permission update testuser6 --ip-
whitelist=192.168.1.*,10.0.0.8
用戶權限配置已保存。
```

傳入 CIDR 格式會被拒絕並提示：

```
secure@localhost> user permission update testuser6 --ip-
whitelist=192.168.1.0/24
用戶操作失敗：允許訪問IP格式非法，只允許IPv4、IPv6地址或*通配符
```

示例3 禁用/啟用用戶

```
secure@localhost> user permission update testuser6 --status=0
用戶權限配置已保存。

secure@localhost> user permission update testuser6 --status=1
用戶權限配置已保存。
```

示例4 解除用戶角色綁定

```
secure@localhost> user permission update testuser6 --unbind-role
用戶權限配置已保存。
```

11.2 角色管理

11.2.1 角色管理 (role)

`role` 子命令用於角色管理，包括角色的創建、查看、更新、刪除及可用權限資源查詢。角色是權限的集合，透過 `user permission update --role` 綁定到用戶。

11.2.1.1 概要

```
Usage:  role [-h] [COMMAND]
角色管理
    -h, --help    顯示說明資訊
```

Commands:

list	列出角色
get	查看角色詳情
add	創建角色
update	更新角色
delete	刪除角色
permissions	列出可用權限資源

11.2.2 列出角色 (role list)

11.2.2.1 概要

```
Usage: role list [--page=<page>] [--output=<format>] [--no-paging] [-h]
```

列出角色

--page=<page>	頁碼
--help	顯示說明資訊
--no-paging	本次命令關閉分頁輸出
--output=<output>	輸出格式: table 或 json

11.2.2.2 示例

示例1 列出角色

```
secure@localhost> role list
| 角色名 | 關聯用戶 | 特權標識 | 說明 |
+-----+-----+-----+-----+
--+
| sysadmin | admin | 是 | System maintenance and user creation |
| security | secure | 是 | Security policy and authorization |
| auditor | audit | 是 | Operation supervision and audit |
第 1 / 1 頁, 共 3 條
```

sysadmin、security、auditor 為系統內建特權角色，不可修改、不可刪除。

11.2.3 查看角色詳情 (role get)

11.2.3.1 概要

```
Usage: role get <ROLE_CODE> [--output=<format>] [-h]
```

查看角色詳情

<code><roleCode></code>	角色編碼, 可用 <code>aascli role list</code> 查詢
<code>--output=<output></code>	輸出格式: json
<code>--help</code>	顯示說明資訊

11.2.3.2 示例

示例1 查看角色詳情

```
secure@localhost> role get test-role
ID: 7
角色編碼: test-role
角色名稱: test-role
說明: CLI-test-role
內置標識: 否
數據密級: 公開
權限列表: dashboard:view, app_list:view
```

11.2.4 列出可用權限資源 (role permissions)

`role permissions` 子命令用於列出系統中全部可用權限資源及其編碼, 創建或更新角色時 `--permissions` 參數取值即來自該列表。

11.2.4.1 概要

```
Usage: role permissions [--output=<format>] [-h]
```

列出可用權限資源

<code>--help</code>	顯示說明資訊
<code>--output=<output></code>	輸出格式: json

11.2.4.2 示例

示例1 列出可用權限資源

輸出為 json 格式的權限資源樹, 每個資源包含 `code` (權限編碼)、`name` (名稱) 及 `children` (子操作權限, 如 view 查看、create 新增、update 更新、delete 刪除) :

```
secure@localhost> role permissions --output=json
[ {
  "code" : "dashboard",
  "name" : "首页概览",
  "resourceId" : 1,
  "children" : [ {
    "code" : "dashboard:view",
    "name" : "查看",
    "resourceId" : null,
    "children" : [ ]
  } ]
}, {
  "code" : "user_mgmt",
  "name" : "用户管理",
  ...
}, ... ]
```

頂層權限資源編碼包括：

權限編碼	名稱
dashboard	首頁概覽
user_mgmt	用戶管理
app_mgmt	應用管理
role_mgmt	角色管理
config_mgmt	配置管理
monitor_mgmt	監控管理
resource_mgmt	資源管理
log_service	日誌服務
security_service	安全服務
system_mgmt	系統管理
audit_log	審計日誌

security_config	安全配置
ops_mgmt	維運巡檢

11.2.5 創建角色 (role add)

11.2.5.1 概要

```
Usage: role add --name=<name> --permissions=<codes> [--description=
<desc>] [-h]
```

創建角色

```
--name=<name>                角色名，必填。取值範圍：字母、數字、底
線、短橫線，最長 64 字元
--permissions=<permissions>  權限資源編碼，多個用英文逗號分隔，必填。
可透過 `aascli role permissions` 查看可用編
碼
--description=<description>  說明。取值範圍：最長 255 字元
--help                        顯示說明資訊
```

11.2.5.2 選項

```
--name=<name>
```

角色名（即角色編碼），字母、數字、底線、短橫線，最長 64 字元。

```
--permissions=<permissions>
```

權限資源編碼，多個用英文逗號分隔，必填。頂層資源編碼（如 app_mgmt）與具體操作編碼（如 app_list:view）均可使用。

```
--description=<description>
```

角色說明，最長 255 字元。

11.2.5.3 示例

示例1 創建角色

```
secure@localhost> role add --name=test-role --
permissions=dashboard:view,app_list:view --description=CLI-test-role
```

角色創建成功。

11.2.6 更新角色 (role update)

11.2.6.1 概要

```
Usage: role update <ROLE_CODE> [--permissions=<codes>] [--description=
<desc>] [-h]
```

更新角色

<code><roleCode></code>	角色編碼
<code>--permissions=<permissions></code>	權限資源編碼，多個用英文逗號分隔。可透過 `aascli role permissions` 查看可用編碼
<code>--description=<description></code>	說明。取值範圍：最長 255 字元
<code>--help</code>	顯示說明資訊

11.2.6.2 示例

示例1 更新角色權限與說明

更新角色時必須攜帶 `--permissions`，且權限列表為全量覆蓋式更新（未保留的權限將被移除）：

```
secure@localhost> role update test-role --permissions=dashboard:view -
-description=updated-desc
```

角色更新成功。

僅指定 `--description` 不指定 `--permissions` 時會提示：

```
secure@localhost> role update test-role --description=updated-desc
角色操作失敗：请至少为角色分配一个权限资源
```

11.2.7 刪除角色 (role delete)

11.2.7.1 概要

```
Usage: role delete <ROLE_CODE> [-f] [-h]
```

刪除角色

<code><roleCode></code>	角色編碼
<code>-f, --force</code>	不提示確認
<code>--help</code>	顯示說明資訊

11.2.7.2 示例

示例1 刪除角色

```
secure@localhost> role delete test-role -f
```

角色刪除成功。

示例2 刪除內建角色被拒絕

```
secure@localhost> role delete sysadmin -f
```

內置角色不能刪除。

11.3 安全配置管理

`security-config` 子命令用於安全策略配置，包括密碼策略、審計日誌配置、業務配置備份、郵件驗證與動態碼驗證。

11.3.1 概要

```
Usage: security-config [-h] [COMMAND]
```

安全配置管理

```
-h, --help    顯示說明資訊
```

Commands:

<code>password-policy</code>	密碼策略配置
<code>audit-log</code>	審計日誌配置
<code>backup</code>	業務配置備份管理
<code>mail</code>	郵件驗證配置
<code>otp</code>	動態碼驗證配置

11.3.2 密碼策略配置 (security-config password-policy)

`password-policy` 子命令用於查看、更新管控台帳號的密碼策略。

11.3.2.1 概要

```
Usage: security-config password-policy [-h] [COMMAND]
```

密碼策略配置

```
-h, --help    顯示說明資訊
```

Commands:

get 查看密碼策略配置
update 更新密碼策略配置

security-config password-policy update 選項:

選項	說明
--password-min-length=	密碼最小長度，取值範圍 8~32 位，預設 8
--password-valid-days=	密碼強制修改週期（天），取值範圍 1~90，預設 30
--password-max-failed-attempts=	連續登入失敗次數上限，取值範圍 1~5 次，預設 5
--password-complexity=	密碼複雜度，normal=普通（字母、數字、特殊符號至少 2 種），complex=複雜（需同時包含 3 種），預設 normal
--password-history-limit=	禁止重複使用的歷史密碼數量，取值範圍 1~10 次，預設 5
--password-lock-duration=	帳號鎖定時長（分鐘），取值範圍 1~1440，預設 15
--weak-passwords=	弱密碼列表，多個用英文逗號分隔，最多 255 個字元，預設空

11.3.2.2 示例

示例1 查看密碼策略配置

```
secure@localhost> security-config password-policy get
```

密碼策略配置：

密碼長度：8 位

密碼有效天數：30 天

密碼重試次數：5 次

密碼複雜度：normal

最近密碼重複限制：5 次

用戶鎖定時間：15 分鐘

弱密碼設置：

示例2 提高密碼強度要求

```
secure@localhost> security-config password-policy update --password-min-length=10 --password-complexity=complex
```

安全配置已保存。

11.3.3 審計日誌配置 (security-config audit-log)

`audit-log` 子命令用於查看、更新審計日誌（登入日誌、操作日誌）的保留策略與定時備份策略。審計日誌的查詢與匯出由審計管理員負責。

11.3.3.1 概要

Usage: `security-config audit-log [-h] [COMMAND]`

審計日誌配置

`-h, --help` 顯示說明資訊

Commands:

`get` 查看審計日誌配置

`update` 更新審計日誌配置

`security-config audit-log update` 選項:

選項	說明
<code>--login-log-max-count=</code>	登入日誌數量上限（條），範圍 1000~100000，預設 10000
<code>--login-log-retention=</code>	登入日誌保留時間（天），範圍 30~365，預設 180
<code>--operation-log-max-count=</code>	操作日誌數量上限（條），範圍 1000~100000，預設 10000
<code>--operation-log-retention=</code>	操作日誌保留時間（天），範圍 30~365，預設 180
<code>--audit-backup-period-days=</code>	審計日誌定時備份週期（天/次），範圍 1~365，預設 30
<code>--audit-backup-retention-days=</code>	審計日誌定時備份保留時間（天），-1 表示永久保留，預設 -1

11.3.3.2 示例

示例1 查看審計日誌配置

```
secure@localhost> security-config audit-log get
```

審計日誌配置：

登錄日誌數量上限：10000 條

登錄日誌保留時間：180 天

操作日誌數量上限：10000 條

操作日誌保留時間：180 天

定時備份週期：30 天/次

定時備份保留時間：永久

示例2 調整日誌保留時間

```
secure@localhost> security-config audit-log update --login-log-retention=90 --operation-log-retention=90
安全配置已保存。
```

11.3.4 郵件驗證配置 (security-config mail)

`mail` 子命令用於查看、更新登入郵件驗證配置。開啟後，用戶登入需輸入信箱驗證碼。

11.3.4.1 概要

Usage: security-config mail [-h] [COMMAND]

郵件驗證配置

-h, --help 顯示說明資訊

Commands:

get 查看郵件驗證配置

update 更新郵件驗證配置

security-config mail update 選項：

選項	說明
--smtp-verify-enabled=	郵件驗證開關：true 或 false，預設 false。開啟後登入需輸入信箱驗證碼，並自動關閉動態碼驗證
--smtp-host=	SMTP 伺服器位址，如 smtp.example.com，預設空
--smtp-ssl=	SMTP SSL 開關：true 或 false，預設 false
--smtp-port=	SMTP 埠，範圍 1~65535，預設 25 (SSL 時通常為 465)
--smtp-user=	發送信箱帳號，預設空
--smtp-password=	發送信箱密碼或用戶端授權碼，預設空
--smtp-license-email=	接收授權提醒信箱，多個用英文分號分隔，預設空

11.3.4.2 示例

示例1 查看郵件驗證配置

```
secure@localhost> security-config mail get
```

郵件驗證配置：

郵件驗證開關：false

發送郵件伺服器：

開啟SSL：false

發送埠：25

發送郵箱：

發送郵箱密碼：

接收授權提醒郵箱：

示例2 配置並開啟郵件驗證

開啟郵件驗證前，需先配置可用的 SMTP 服務，否則所有用戶將無法正常登入：

```
secure@localhost> security-config mail update --smtp-  
host=smtp.example.com --smtp-ssl=true --smtp-port=465 --smtp-  
user=alerts@example.com --smtp-password=***** --smtp-verify-  
enabled=true  
安全配置已保存。
```

11.3.5 動態碼驗證配置 (security-config otp)

`otp` 子命令用於查看、更新登入動態碼（OTP）驗證配置。開啟後，用戶登入需同時輸入密碼與 Authenticator 動態驗證碼。

11.3.5.1 概要

```
Usage: security-config otp [-h] [COMMAND]
```

動態碼驗證配置

`-h, --help` 顯示說明資訊

Commands:

`get` 查看動態碼驗證配置

`update` 更新動態碼驗證配置

```
Usage: security-config otp update --enabled=<true|false> [-h]
```

更新動態碼驗證配置

```
--enabled=<true|false>    動態碼驗證開關：true 或 false。開啟後登入需同時輸入密碼與 Authenticator 動態驗證碼，並自動關閉郵件驗證
--help                    顯示說明資訊
```

11.3.5.2 示例

示例1 查看動態碼驗證配置

```
secure@localhost> security-config otp get
動態碼驗證配置：
  OTP驗證開關：false
```

示例2 開啟/關閉動態碼驗證

動態碼驗證與郵件驗證互斥，開啟其中一個會自動關閉另一個：

```
secure@localhost> security-config otp update --enabled=true
安全配置已保存。

secure@localhost> security-config otp update --enabled=false
安全配置已保存。
```

11.3.6 業務配置備份管理 (security-config backup)

`backup` 子命令用於業務配置的備份與還原，支援定時備份策略配置、手動立即備份、備份查看、還原與刪除。

11.3.6.1 概要

```
Usage: security-config backup [-h] [COMMAND]
```

業務配置備份管理

```
-h, --help    顯示說明資訊
```

Commands:

<code>get</code>	查看業務配置備份設定
<code>update</code>	更新業務配置備份設定
<code>backup-now</code>	立即執行業務配置備份
<code>list</code>	列出業務配置備份
<code>restore</code>	恢復業務配置備份
<code>delete</code>	刪除業務配置備份

`security-config backup update` 選項：

選項	說明
<code>--backup-enabled=</code>	定時備份開關：true 或 false，預設 false
<code>--backup-path=</code>	共享備份目錄，支援系統變數，預設 <code>\${apusic.home}/store/config_bak</code>
<code>--backup-period-months=</code>	定時備份週期（月/次），範圍 1~12，預設 1

`security-config backup restore` 選項：

選項	說明
	備份檔案名，可透過 <code>security-config backup list</code> 查詢
<code>-f, --force</code>	不提示恢復確認
<code>-r, --restart</code>	恢復成功後立即重啟，不再提示確認

11.3.6.2 示例

示例1 查看業務配置備份設定

```
secure@localhost> security-config backup get
```

業務配置備份設置：

定時備份開關：false

共享備份目錄：\${apusic.home}/store/config_bak

定時備份週期：1 月/次

示例2 開啟定時備份

```
secure@localhost> security-config backup update --backup-enabled=true
--backup-period-months=1
```

安全配置已保存。

示例3 立即執行業務配置備份

```
secure@localhost> security-config backup backup-now
```

備份成功：业务配置备份成功

示例4 列出業務配置備份

```
secure@localhost> security-config backup list
```

備份文件名	備份類型	備份時間
config_backup_manual_20260927_175837.zip	手動備份	2026-09-27 17:58:37
36.33 KB		

第 1 / 1 頁, 共 1 條

示例5 恢復業務配置備份

恢復會以備份檔案內容覆蓋當前業務配置，恢復後需重啟伺服器生效：

```
secure@localhost> security-config backup restore
config_backup_manual_20260927_175837.zip
```

示例6 刪除業務配置備份

```
secure@localhost> security-config backup delete
config_backup_manual_20260927_175837.zip -f
備份已刪除。
```

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

