



Command Line Operation Guide

Kingdee Apusic Application Server V10 Standard Edition

Copyright Notice

All intellectual property rights including software copyrights involved in this document have been legally registered and duly owned by Kingdee Apusic Cloud Computing Co., Ltd. Such rights are protected by Copyright Law of the People's Republic of China, Regulations on Protection of Computer Software, Regulations on Intellectual Property Protection, relevant international copyright conventions, laws, regulations and other intellectual property legislations. Unauthorized use is strictly prohibited.

Disclaimer

All copyright information contained herein is legally owned by Kingdee Apusic Cloud Computing Co., Ltd. and protected by applicable laws. The Company shall bear no liability for any information unrelated to Kingdee Apusic contained in this document. You may view, copy and print this document only within the scope permitted by Copyright Law of the People's Republic of China. No entity or individual may use, modify or republish any part or content of this document without written authorization from Kingdee Apusic Cloud Computing Co., Ltd. Any violation shall constitute infringement, and the Company reserves the right to pursue legal liabilities accordingly.

Updates to this document shall not be notified separately. You may consult the Company regarding issues related to this document. All rights not expressly granted by the Company are reserved.

Trademark Notice

 is a registered trademark filed with the National Intellectual Property Administration of PRC by Kingdee Apusic Cloud Computing Co., Ltd. Shenzhen. Exclusive rights to the registered trademark are legally owned by Kingdee Apusic and protected by law. Without written consent of Kingdee Apusic, no entity or individual shall use, copy, modify, disseminate, transcribe, or bundle and sell any part of this trademark with other products in any manner or for any reason. Kingdee Apusic shall pursue legal liabilities against any infringement of its trademark rights. All other trademarks or registered trademarks mentioned in this document belong to their respective owners.

Contents

- 1 Preface
 - 1.1 Intended Audience
 - 1.2 Related Documentation
 - 1.3 Technical Support
- 2 Overview
 - 2.1 Introduction
 - 2.2 Terminology
- 3 System Requirements
 - 3.1 Platform
 - 3.2 Environment
- 4 Package Contents
- 5 Installation and Uninstallation
 - 5.1 Java Environment
 - 5.2 Installation
 - 5.3 Uninstallation
- 6 Licensing
 - 6.1 Standard Licensing
 - 6.2 Centralized Licensing
 - 6.3 Obtaining the Auth Code
 - 6.4 Obtaining the Auth Code in Embedded Mode
- 7 Starting the Server
 - 7.1 On Linux
 - 7.2 On Windows
- 8 Stopping the Server
 - 8.1 On Linux
 - 8.2 On Windows
- 9 The aascli Command Line Tool
 - 9.1 Introduction
 - 9.2 Logging In
 - 9.2.1 Synopsis
 - 9.2.2 Options
 - 9.2.3 Examples

- 9.3 Logging Out
 - 9.3.1 Synopsis
 - 9.3.2 Example
- 9.4 Viewing Available Commands
- 9.5 General Conventions
- 10 System Administrator Module
 - 10.1 Application Management
 - 10.1.1 Application management (app)
 - 10.1.1.1 Synopsis
 - 10.1.1.2 Subcommands
 - 10.1.2 Listing applications (app list)
 - 10.1.2.1 Synopsis
 - 10.1.2.2 Options
 - 10.1.2.3 Examples
 - 10.1.3 Viewing application details (app get)
 - 10.1.3.1 Synopsis
 - 10.1.3.2 Options
 - 10.1.3.3 Example
 - 10.1.4 Updating application configuration (app update)
 - 10.1.4.1 Synopsis
 - 10.1.4.2 Options
 - 10.1.4.3 Examples
 - 10.1.5 Uninstalling an application (app delete)
 - 10.1.5.1 Synopsis
 - 10.1.5.2 Options
 - 10.1.5.3 Examples
 - 10.1.6 Reloading an application (app reload)
 - 10.1.6.1 Synopsis
 - 10.1.6.2 Example
 - 10.1.7 Deploying an application (app deploy)
 - 10.1.7.1 Synopsis
 - 10.1.7.2 Options
 - 10.1.7.3 Examples
 - 10.1.8 Redeploying an application (app redeploy)

- 10.1.8.1 Synopsis
- 10.1.8.2 Example
- 10.1.9 Application class library management (app class-lib)
 - 10.1.9.1 Synopsis
 - 10.1.9.2 Options
 - 10.1.9.3 Examples
- 10.1.10 Application incremental update (app increment)
 - 10.1.10.1 Synopsis
 - 10.1.10.2 Options
 - 10.1.10.3 Examples
- 10.1.11 Backing up an application (app backup)
 - 10.1.11.1 Synopsis
 - 10.1.11.2 Example
- 10.1.12 Application backup management (app-backup)
 - 10.1.12.1 Synopsis
 - 10.1.12.2 Examples
- 10.1.13 Application recycle management (app-recycle)
 - 10.1.13.1 Synopsis
 - 10.1.13.2 Examples
- 10.2 Configuration Management
 - 10.2.1 Virtual host management (host)
 - 10.2.1.1 Synopsis
 - 10.2.1.2 Examples
 - 10.2.2 Thread pool management (threadpool)
 - 10.2.2.1 Synopsis
 - 10.2.2.2 Examples
 - 10.2.3 Session manager management (session-manager)
 - 10.2.3.1 Synopsis
 - 10.2.3.2 Examples
 - 10.2.4 HTTP channel management (httpchannel)
 - 10.2.4.1 Synopsis
 - 10.2.4.2 Examples
 - 10.2.5 JVM configuration management (jvm)
 - 10.2.5.1 Synopsis

- 10.2.5.2 Examples
- 10.2.6 JMX configuration management (jmx)
 - 10.2.6.1 Synopsis
 - 10.2.6.2 Examples
- 10.2.7 Certificate management (cert)
 - 10.2.7.1 Synopsis
 - 10.2.7.2 Examples
- 10.3 Resource Management
 - 10.3.1 Datasource management (datasource)
 - 10.3.1.1 Synopsis
 - 10.3.1.2 Examples
- 10.4 Monitoring Management
 - 10.4.1 Monitoring configuration management (monitor-config)
 - 10.4.1.1 Synopsis
 - 10.4.1.2 Examples
 - 10.4.2 Alert configuration management (alert-config)
 - 10.4.2.1 Synopsis
 - 10.4.2.2 Examples
 - 10.4.3 SNMP agent configuration management (snmp-config)
 - 10.4.3.1 Synopsis
 - 10.4.3.2 Examples
 - 10.4.4 One-click inspection management (inspection)
 - 10.4.4.1 Synopsis
 - 10.4.4.2 Examples
 - 10.4.5 Monitoring snapshot management (snapshot)
 - 10.4.5.1 Synopsis
 - 10.4.5.2 Examples
- 10.5 Log Service
 - 10.5.1 System log configuration (log-config system)
 - 10.5.1.1 Synopsis
 - 10.5.1.2 Examples
 - 10.5.2 Access log configuration (log-config access)
 - 10.5.2.1 Synopsis
 - 10.5.2.2 Examples

- 10.6 User Management
 - 10.6.1 User management (user)
 - 10.6.1.1 Synopsis
 - 10.6.1.2 Examples
- 10.7 Security Service
 - 10.7.1 Security domain management (security-domain)
 - 10.7.1.1 Synopsis
 - 10.7.1.2 Examples
- 10.8 System and Account
 - 10.8.1 User settings (usersettings)
 - 10.8.1.1 Synopsis
 - 10.8.1.2 Options
 - 10.8.1.3 Examples
 - 10.8.2 Changing the current user's password (changepassword)
 - 10.8.2.1 Synopsis
 - 10.8.2.2 Options
 - 10.8.2.3 Examples
- 11 Security Administrator Module
 - 11.1 User Management
 - 11.1.1 User management (user)
 - 11.1.1.1 Synopsis
 - 11.1.2 Listing users (user list)
 - 11.1.2.1 Example
 - 11.1.3 Viewing user details (user get)
 - 11.1.3.1 Example
 - 11.1.4 Resetting a user password (user reset-password)
 - 11.1.4.1 Synopsis
 - 11.1.4.2 Options
 - 11.1.4.3 Example
 - 11.1.5 Deleting a user (user delete)
 - 11.1.5.1 Example
 - 11.1.6 User permission configuration (user permission)
 - 11.1.6.1 Synopsis
 - 11.1.6.2 Options

- 11.1.6.3 Examples
- 11.2 Role Management
 - 11.2.1 Role management (role)
 - 11.2.1.1 Synopsis
 - 11.2.2 Listing roles (role list)
 - 11.2.2.1 Synopsis
 - 11.2.2.2 Example
 - 11.2.3 Viewing role details (role get)
 - 11.2.3.1 Synopsis
 - 11.2.3.2 Example
 - 11.2.4 Listing available permission resources (role permissions)
 - 11.2.4.1 Synopsis
 - 11.2.4.2 Example
 - 11.2.5 Creating a role (role add)
 - 11.2.5.1 Synopsis
 - 11.2.5.2 Options
 - 11.2.5.3 Example
 - 11.2.6 Updating a role (role update)
 - 11.2.6.1 Synopsis
 - 11.2.6.2 Example
 - 11.2.7 Deleting a role (role delete)
 - 11.2.7.1 Synopsis
 - 11.2.7.2 Examples
- 11.3 Security Configuration Management
 - 11.3.1 Synopsis
 - 11.3.2 Password policy configuration (security-config password-policy)
 - 11.3.2.1 Synopsis
 - 11.3.2.2 Examples
 - 11.3.3 Audit log configuration (security-config audit-log)
 - 11.3.3.1 Synopsis
 - 11.3.3.2 Examples
 - 11.3.4 Mail verification configuration (security-config mail)
 - 11.3.4.1 Synopsis
 - 11.3.4.2 Examples

- 11.3.5 OTP verification configuration (security-config otp)
 - 11.3.5.1 Synopsis
 - 11.3.5.2 Examples
- 11.3.6 Business configuration backup management (security-config backup)
 - 11.3.6.1 Synopsis
 - 11.3.6.2 Examples

1 Preface

This document describes the command line operations of Kingdee Apusic Application Server (AAS) V10 Standard Edition, covering system requirements, installation, and daily administration via the aascli command line tool. It is intended for developers using AAS, system administrators in production environments, and O&M personnel.

1.1 Intended Audience

This guide is intended for developers building applications on AAS, system administrators in production environments, application publishers, and technical O&M engineers. The following skills will help you make better use of this guide:

- Familiarity with common Linux commands
- Basic system administration tasks
- Installing and managing software

1.2 Related Documentation

For more information about AAS V10 Standard Edition, refer to the AAS V10 product documentation set:

No.	Documentation	Description
1	Kingdee Apusic Application Server V10 Standard Edition Installation Guide	Provides detailed instructions on how to install AAS on various operating systems, as well as AAS service start/stop operations and the product registration process
2	Kingdee Apusic Application Server V10 Standard Edition Quick Start Guide	Briefly introduces how to quickly get started with AAS
3	Kingdee Apusic Application Server V10 Standard Edition User Manual	Provides detailed instructions on the usage and operation of AAS-related features, guiding users through system installation and deployment, feature operations, parameter configuration, daily operations and maintenance, and troubleshooting
4	Kingdee Apusic Application Server V10 Standard Edition Administration Console User Manual	Provides detailed instructions on the usage and operation of AAS Administration Console features, guiding users to perform operations through the visual administration console
5	Kingdee Apusic Application Server V10 Standard Edition Command-Line Operation	Provides detailed instructions on how to operate and use AAS through the command line

	n Guide	
6	Kingdee Apusic Application Server V10 Standard Edition Monitoring Metrics Documentation	Provides detailed descriptions of various types of AAS monitoring metrics
7	Kingdee Apusic Application Server V10 Standard Edition API Documentation	Provides detailed information about AAS APIs

1.3 Technical Support

Kingdee Apusic Application Server comes with comprehensive technical support services. You can obtain support in the following ways:

- Website: www.apusic.com
- Phone: 400-855-5800
- Email: support@apusic.com
- Kingdee Cloud Community: <https://vip.kingdee.com/?productId=73&productLineId=14&lang=zh-CN>

When contacting technical support, please provide the following information:

1. Your name
2. Company information and contact details
3. Operating system and version
4. Product version number
5. Details of the exception or error, including logs and screenshots

2 Overview

2.1 Introduction

Kingdee Apusic Application Server (AAS) V10 Standard Edition is a JAVAEE-compliant web application server. It is a lightweight application server typically used for efficient application development.

2.2 Terminology

AAS: Kingdee Apusic Application Server

AE: Product package identifier of Kingdee Apusic Application Server Standard Edition

APUSIC_HOME: Installation directory of Kingdee Apusic Application Server

3 System Requirements

3.1 Platform

AAS is compatible with various operating systems, as listed below.

Table 1: Operating systems supported by AAS V10 Standard Edition

Operating System	Description
Windows	Windows 7, Windows 10, Windows Server
Linux	CentOS, Ubuntu, SUSE, Redhat
Domestic platforms	Kylin series, NeoKylin, i-soft, Red Flag, Deepin, etc.

3.2 Environment

The recommended system environment for AAS is as follows.

Table 2: System environment for AAS V10 Standard Edition

Component	Requirement
Java	JDK 8 or later; JDK 8u252 or later is recommended for HTTP/2
Memory	At least 512 MB; 1024 MB or more is recommended
Disk space	At least 1024 MB; 10 GB or more is recommended

4 Package Contents

The product package is usually distributed as `AAMS-[version].tar.gz` or `AAMS-[version].zip`.

In this document, `${APUSIC_HOME}` refers to the installation directory of the application server.

Table 3: Product package of AAS V10 Standard Edition

Package	Description
AAMS-[version].zip / AAMS-[version].tar.gz	Installation package of AAS V10 Standard Edition

Description of the product package contents.

Table 4: Product package contents of AAS V10 Standard Edition

File/Directory	Description
bin	Scripts for starting and stopping the application server
conf	Configuration files of the application server
console	Management console of the application server
lib	JAR files of the application server
logs	Default log directory of the application server
plugins	Plugins of the application server
store	Data repository directory for centralized data storage
webapps	Directory for auto-deployed application packages
webapps-console	Directory for application packages deployed via the console
webapps-jakarta	Directory for JakartaEE application packages
temp	Temporary files of the application server
work	Compiled JSP class files
license.xml	License file of the product package

5 Installation and Uninstallation

5.1 Java Environment

Before installation, verify the Java environment. JDK 1.8.0_201 or later is recommended. Check the JDK version as follows:

```
[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)
```

To specify a Java runtime environment, set `JAVA_HOME` at the beginning of `apusic.sh` or `apusic.bat`, for example:

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

5.2 Installation

Prepare the package `AAMS-V10.0.8-SP12-AE-20260701.tar.gz` or `AAMS-V10.0.8-SP12-AE-20260701.zip`, copy it to a directory with appropriate permissions, and extract it.

Extraction completes the installation.

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

5.3 Uninstallation

Delete all files under the installation directory `${APUSIC_HOME}/` to uninstall AAS. Uninstallation discards all configured data of the current AAS instance. Proceed with caution.

6 Licensing

AAS requires a valid license to work properly. Kingdee Apusic normally issues a license matching the purchased product edition. If the license expires or becomes invalid, contact your Apusic service representative first to apply for a new license.

There are two licensing modes: standard licensing and centralized licensing.

6.1 Standard Licensing

In standard licensing, a `license.xml` file is generated based on the IP address, domain name, etc. Place the license file in the installation directory: `${APUSIC_HOME}/license.xml`.

If the management console is installed, log in to the console as the system administrator admin, click the license icon in the upper right corner, click "Update License", upload a valid license file, and save. Restart AAS for the new license to take effect.

6.2 Centralized Licensing

Centralized licensing connects to a license center for unified authorization. You need to deploy the Kingdee Apusic License Center first. See the Kingdee Apusic License Center User Guide, or contact Kingdee Apusic technical support.

Configure environment variables in the operating system, or create an `acls.properties` file in the root of the AAS installation directory with the following parameters:

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

Connection parameters:

Table 5: License Center connection parameters

Parameter	Description
apusic_acls_enable	Whether to enable license center authentication. Set to true or false; true enables license center authentication. If the parameter is absent or false, license center authentication is disabled.

apusic_acls_authUrls	Address(es) of the license center. Multiple addresses in the format ip1:port1,ip2:port2 are supported; if one fails, the others are tried in turn. Required when license center authentication is enabled. The port is the HTTPS port of the license center.
apusic_acls_ns	Namespace of this instance. Optional; the default is public. Namespaces can be viewed in the license center console under System Management > License Management.
apusic_acls_tenant	Tenant name of this instance. Optional.

AAS connects to the Apusic License Center automatically at startup.

6.3 Obtaining the Auth Code

If the license expires or becomes invalid, contact your Apusic service representative to apply for a new one. When applying, provide the auth code of the product.

In `${APUSIC_HOME}/bin`, run `apusic.sh -ac [ethname or ip]`, where `[ethname or ip]` is an IP address or network interface name, for example:

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

The auth code is printed as follows, Auth Code=:

```
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
```

Provide the auth code when applying for a license file.

6.4 Obtaining the Auth Code in Embedded Mode

When AAS is embedded with Spring Boot, there is no `apusic` command to obtain the auth code; it must be obtained through your application.

If you have a temporary license file that allows the application to start, remove the license file first, then add the JVM parameter `apusic-ac` to the start command, with an IP address or network interface name as its value. In the following example, `eth3` is the network interface name of the test machine; adjust it according to your actual network configuration:

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

Without a license file the application startup will stop, and the auth code will be printed before the process exits, for example:

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

7 Starting the Server

7.1 On Linux

After installation and license update, go to `${APUSIC_HOME}/bin` and run `apusic.sh run`.

You can also run `./startapusic.sh` to start the server.

7.2 On Windows

After installation and license update, go to `${APUSIC_HOME}\bin` and run `apusic.bat run`.

You can also run `startapusic.bat` to start the server.

8 Stopping the Server

8.1 On Linux

1. Ctrl + C

When running in the foreground, press Ctrl + C to stop AAS.

2. stopapusic.sh

Go to `${APUSIC_HOME}/bin` and run `stopapusic.sh`.

3. kill the process

Stop AAS by running `kill [pid]`.

8.2 On Windows

1. Ctrl + C

When running in the foreground, press Ctrl + C to stop AAS.

2. stopapusic.bat

Go to `${APUSIC_HOME}\bin` and run `stopapusic.bat`.

3. Stop the process

Open Task Manager, locate the process, and stop it.

9 The aascli Command Line Tool

9.1 Introduction

AAS provides the command line tool `aascli` for daily administration and O&M of the application server. Its features are consistent with the management console. The tool is located in `${APUSIC_HOME}/bin`:

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

Before using `aascli`, make sure that:

1. `JAVA_HOME` or `JRE_HOME` is configured (JDK 8 or later);
2. The AAS server is running (`aascli` communicates with the server over the console port, 6888 by default);
3. You have an account that can log in to the console.

`aascli` supports three interface languages: `en`, `zh`, and `zh_TW`. Switch the language with `usersettings --lang=<language>`; the change takes effect the next time you log in to `aascli`. Example outputs in this guide are labeled by language.

9.2 Logging In

The `login` command logs in to the AAS console and enters the interactive command space after success. AAS ships with three built-in administrator accounts under the separation-of-duties ("three administrators") model:

Username	Role	Description
admin	sysadmin	System administrator: responsible for applications, configuration, resources, monitoring, logs, and other system O&M
secure	security	Security administrator: responsible for roles, security configuration, and other security management (covered in this guide)
audit	auditor	Audit administrator: responsible for audit log management (not covered in this guide)

9.2.1 Synopsis

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

Log in to the AAS console

```
--file=<loginFile>          Login file path (contains
encrypted login information, generated by --generate-file)
--generate-file=<generateFile>  Verify the credentials,
generate a login file with encrypted login information, and exit
without entering the interactive command space
-h, --help                    Show help information
--no-interactive              Exit after a successful login
without entering the interactive command space (for script automation)
-p, --password=<password>      Password
-u, --username=<username>      Username
```

9.2.2 Options

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

Login username. The default system administrator is admin. Entered interactively if not provided.

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

Login password. Entered interactively if not provided (input is not echoed). Can be specified directly in automation scripts.

```
--no-interactive
```

Exit after a successful login without entering the interactive command space; used to verify account availability in scripts.

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

Verify the credentials, generate a login file containing encrypted login information, and exit. The generated file can be used with `--file` to avoid keeping plaintext passwords in scripts.

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

Log in with a login file containing encrypted login information, generated by `--generate-file`.

9.2.3 Examples

Example 1 Log in as the system administrator

Go to `${APUSIC_HOME}/bin` and run `aascli`; enter the username and password at the interactive prompts (password input is not echoed):

```
[user@test bin]$ ./aascli.sh
aas> login
用户名: admin
密码:
admin logged in
admin@localhost>
```

You can also specify the username and password on the login command:

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

After a successful login you enter the interactive command space with the prompt `username@hostname>`, where you can enter feature subcommands.

Example 2 Prompt on a wrong password

Repeated wrong passwords will lock the account; proceed with caution:

```
Login failed: 用户名或密码错误, 剩余重试次数 4 次
```

Example 3 Generate a login file for script automation

Generate an encrypted login file first, then log in with it to avoid plaintext passwords in scripts:

```
[user@test bin]$ ./aascli.sh login -u admin -p [YOURPASSWORD] --
generate-file=/home/user/aas.login
Login file generated: /home/user/aas.login

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

Example 4 Verify account availability and exit

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

9.3 Logging Out

The `logout` command exits the current login. To leave the interactive command space, use `exit` or Ctrl + D.

9.3.1 Synopsis

```
Usage: logout [-h]
Log out
    -h, --help    Show help information
```

9.3.2 Example

Example 1 Log out

```
admin@localhost> logout
```

9.4 Viewing Available Commands

The `help` command displays the commands available to the current user. The list depends on the role and permissions of the logged-in user. The commands visible to the system administrator are:

```
admin@localhost> help
Application management:
    app          应用列表
    app-backup   应用备份
    app-recycle  应用回收

Configuration management:
    host         Virtual host management
    threadpool   线程池管理
    session-manager 会话管理器管理
    httpchannel  HTTP 通道管理
    jvm          JVM 配置管理
```

```
jmx      JMX 配置管理
cert     证书管理
```

Resource management:

```
datasource 数据源管理
```

Monitoring:

```
monitor-config 监控配置管理
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)
```

Log service:

```
log-config 日志配置管理
```

User management:

```
user      用户管理
```

Security service:

```
security-domain 安全域管理
```

System and account:

```
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
```

The command lists differ by role: see "System Administrator Module" for the system administrator and "Security Administrator Module" for the security administrator. The audit administrator module is not covered in this guide.

9.5 General Conventions

1. **Command structure:** Commands follow the structure `main-command [subcommand] [arguments] [options]`, e.g. `app list --page=2`. Run `main-command --help` or `main-command subcommand --help` to view usage at each level.
2. **Help option:** All commands support `-h` or `--help`.

3. **Confirmation prompts:** Destructive operations prompt for confirmation by default; enter `y` to proceed or anything else to cancel. Specify `-f` or `--force` to skip the confirmation, which is useful in scripts.
4. **Output format:** List queries support `--output=table|json` (default table), `--page=<n>` to select a page, and `--no-paging` to disable paging and output all records at once. Detail queries support `--output=json`.
5. **Taking effect:** Some configuration operations report "Restart server to take effect"; restart AAS for the new configuration to take effect.

10 System Administrator Module

10.1 Application Management

10.1.1 Application management (app)

The `app` command manages applications, including deployment, undeployment, updates, backup, class libraries, and incremental packages. It usually requires a specific subcommand.

10.1.1.1 Synopsis

```
Usage:  app [-h] [COMMAND]
Manage applications
-h, --help    Show help information
Commands:
list          List applications
get           View application details
update        Update application configuration
delete        Uninstall application
reload        Reload application
deploy        Deploy application
redeploy      Redeploy application
class-lib     Application class library management
increment     Application incremental update
backup        Back up application
```

10.1.1.2 Subcommands

`list` — list applications; `get` — view application details; `update` — update application configuration; `delete` — uninstall application; `reload` — reload application; `deploy` — deploy application; `redeploy` — redeploy application; `class-lib` — application class library management; `increment` — application incremental update; `backup` — back up application.

10.1.2 Listing applications (app list)

The `app list` subcommand lists deployed applications, showing application name, context path, virtual host, and other information.

10.1.2.1 Synopsis

```
Usage: app list [--page=<page>] [--output=<format>] [--no-paging] [-h]
List applications
  -h, --help            Show help information
  --page=<page>         Page number
  --no-paging           Disable paging for this command
  --output=<output>     Output format: table or json
```

10.1.2.2 Options

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

Page number to display.

```
--no-paging
```

Disable paging for this command.

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

Output format: table or json. table is the default; json outputs JSON.

10.1.2.3 Examples

Example 1 List deployed applications

```
admin@localhost> app list
| appName | contextPath | virtualHost | deployOrder | deployTime
| classLevel |
+-----+-----+-----+-----+-----+
-+-----+
| testapp | /testapp    | localhost   | 100          | 2026-09-27 17:36
| Secret   |
| ROOT     | /           | localhost   | 100          | 2026-09-27 14:13
| Secret   |
Page 1 / 1, total 2
```

Example 2 List deployed applications in json format

```

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" : "Secret",
    "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 Viewing application details (app get)

The `app get` subcommand shows the configuration details of a specified application.

10.1.3.1 Synopsis

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

View application details

<code><name></code>	Application name
<code>--virtual-host=<virtualHost></code>	Virtual host name; query with <code>aascli host list</code> ; required when multiple applications share the name
<code>--output=<output></code>	Output format: json
<code>--help</code>	Show help information

10.1.3.2 Options

```
<name>
```

Application name; query with `app list`.

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

Virtual host name. Required when applications with the same name exist under different virtual hosts.

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

Output format: json. When not specified, formatted text is shown.

10.1.3.3 Example

Example 1 View application details

```
admin@localhost> app get ROOT
Application name: ROOT
Context path: /
Class loader order:
JSP precompile: no
Virtual host: localhost
Deploy order: 100
Security domain: -
Classification level: Secret
Description: -
```

10.1.4 Updating application configuration (app update)

The `app update` subcommand updates the configuration of a deployed application and can reload the application immediately to apply the changes.

10.1.4.1 Synopsis

```
Usage: app update <NAME> [--virtual-host=<host>] [options] [--reload]
Update application configuration
    <name>                                Application name
    --virtual-host=<virtualHost>          Virtual host name;
query with aascli host list; required when multiple applications share
the name
    --reload                              Reload the application
immediately after a successful update to apply the configuration
    --context-path=<contextPath>          Context path for
accessing the application; defaults to the application name; must be
unique within a virtual host
    --class-loader-order=<classLoaderOrder> Class loader order:
parent-first or child-first (application first)
    --jsp-precompile=<jspPrecompile>      Whether to precompile
all JSP files at deployment: true or false
    --deploy-order=<deployOrder>          Application startup
priority; smaller values start first; range 0-2147483647
    --security-domain=<securityDomain>    Security domain used
by the application; see aascli security-domain list
    --session-manager=<sessionManager>    Session manager bound
to the application; see aascli session-manager list; pass an empty
string to unbind
    --class-level=<classLevel>            Application
classification level: 2=Secret, 3=Confidential, 4=Top secret
    --description=<description>           Application
description, up to 1000 characters
    --deploy-props=<deployProps>         Custom deployment
properties in the form key1=value1,key2=value2; pass an empty string
to clear
```

10.1.4.2 Options

```
--reload
```

Reload the application immediately after a successful update. Otherwise the configuration takes effect the next time the application is loaded.

```
--context-path=<contextPath>
```

Context path for accessing the application; defaults to the application name; must be unique within a virtual host.

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

Class loader order: parent-first or child-first.

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

Whether to precompile all JSP files at deployment: true or false.

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

Application startup priority; smaller values start first; range 0-2147483647.

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

Security domain used by the application.

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

Session manager bound to the application; pass an empty string to unbind.

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

Application classification level: 2=Secret, 3=Confidential, 4=Top secret.

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

Application description, up to 1000 characters.

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

Custom deployment properties in the form key1=value1,key2=value2; pass an empty string to clear.

10.1.4.3 Examples

Example 1 Update the application description and reload

```
admin@localhost> app update testapp --description=cli-guide-demo --
reload
Application configuration updated.
Application reloaded.
```

Example 2 Change the application startup priority

```
admin@localhost> app update testapp --deploy-order=50
Application configuration updated.
```

10.1.5 Uninstalling an application (app delete)

The `app delete` subcommand uninstalls an application. If the recycle bin is enabled, the uninstalled application is moved to the recycle bin; otherwise it is removed directly.

10.1.5.1 Synopsis

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

    <name>                                Application name
    --virtual-host=<virtualHost>          Virtual host name; query with
aascli host list; required when multiple applications share the name
    -f, --force                            Do not prompt for confirmation
    --help                                Show help information
```

10.1.5.2 Options

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

Do not prompt for confirmation. Useful in scripts.

10.1.5.3 Examples

Example 1 Uninstall an application (with confirmation)

The confirmation prompt shows the numeric ID of the application, which can be queried with `app get` :

```
admin@localhost> app delete testapp
Are you sure you want to uninstall application 11? (y/N) y
Application uninstalled.
```

Example 2 Uninstall an application (without confirmation)

```
admin@localhost> app delete testapp -f
Application uninstalled.
```

10.1.6 Reloading an application (app reload)

The `app reload` subcommand reloads an application, typically to apply updated application content.

10.1.6.1 Synopsis

```
Usage: app reload <NAME> [--virtual-host=<host>] [-h]
Reload application
    <name>                                Application name
    --virtual-host=<virtualHost>          Virtual host name; query with
aascli host list; required when multiple applications share the name
    --help                                Show help information
```

10.1.6.2 Example

Example 1 Reload an application

```
admin@localhost> app reload testapp
Application reloaded.
```

10.1.7 Deploying an application (app deploy)

The `app deploy` subcommand deploys a new application from a WAR file or an application directory.

10.1.7.1 Synopsis

```
Usage: app deploy --app-name=<name> --app-path=<path> [options] [-h]
Deploy application
    --app-name=<appName>                  Application name;
letters, digits, underscores, hyphens, or dots only; up to 255
characters
```

```

--app-path=<appPath>                Path of the WAR file
or application directory on the server
--context-path=<contextPath>         Context path for
accessing the application; defaults to the application name; must be
unique within a virtual host
--class-loader-order=<classLoaderOrder> Class loader order:
parent-first or child-first (application first); default child-first
--jsp-precompile=<jspPrecompile>      Whether to precompile
all JSP files at deployment; default false
--virtual-host=<virtualHost>          Virtual host bound to
the application; default localhost; see aascli host list
--deploy-order=<deployOrder>          Application startup
priority; smaller values start first; range 0-2147483647; default 100
--security-domain=<securityDomain>    Security domain used
by the application; see aascli security-domain list
--session-manager=<sessionManager>    Session manager bound
to the application; see aascli session-manager list
--class-level=<classLevel>            Application
classification level: 2=Secret, 3=Confidential, 4=Top secret; default
2
--description=<description>           Application
description, up to 1000 characters
--deploy-props=<deployProps>          Custom deployment
properties in the form key1=value1,key2=value2
--help                                Show help information

```

10.1.7.2 Options

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

Application name; letters, digits, underscores, hyphens, or dots only; up to 255 characters. Required.

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

Path of the WAR file or application directory on the server. Required. The path must be accessible on the server; paths without Chinese characters are recommended.

The remaining options are the same as those of `app update`.

10.1.7.3 Examples

Example 1 Deploy an application directory

```
admin@localhost> app deploy --app-name=testapp --app-path=D:/aas-
testapp --description=cli-demo
Application created, id: 7
Application deployed.
```

Example 2 Deploy a WAR file with context path and classification level

```
admin@localhost> app deploy --app-name=demo --app-
path=/opt/apps/demo.war --context-path=/demo --class-level=3
Application created, id: 8
Application deployed.
```

Note: `--app-path` must be a local path accessible on the server; avoid paths containing Chinese characters or spaces.

10.1.8 Redeploying an application (app redeploy)

The `app redeploy` subcommand redeploy a specified application from a new package; unspecified configuration items remain unchanged.

10.1.8.1 Synopsis

```
Usage: app redeploy <NAME> [--virtual-host=<host>] --app-path=<path>
[options] [-h]
Redeploy application
    <name>                                Application name
    --virtual-host=<virtualHost>          Virtual host name;
query with aascli host list; required when multiple applications share
the name
    --app-path=<appPath>                  Path of the WAR file
or application directory on the server
    --context-path=<contextPath>          Context path for
accessing the application; defaults to the application name; must be
unique within a virtual host
    --class-loader-order=<classLoaderOrder> Class loader order:
```

```

parent-first or child-first (application first)
  --jsp-precompile=<jspPrecompile>           Whether to precompile
all JSP files at deployment: true or false
  --deploy-order=<deployOrder>               Application startup
priority; smaller values start first; range 0-2147483647
  --security-domain=<securityDomain>         Security domain used
by the application; see aascli security-domain list
  --class-level=<classLevel>                 Application
classification level: 2=Secret, 3=Confidential, 4=Top secret
  --description=<description>                Application
description, up to 1000 characters
  --deploy-props=<deployProps>              Custom deployment
properties in the form key1=value1,key2=value2
  --help                                     Show help information

```

10.1.8.2 Example

Example 1 Redeploy an application

```

admin@localhost> app redeploy testapp --app-path=D:/aas-testapp
Application redeployed.

```

10.1.9 Application class library management (app class-lib)

The `app class-lib` subcommand manages the shared class libraries (JAR files) of a specified application: list, add, and delete.

10.1.9.1 Synopsis

```

Usage:  app class-lib [-h] [COMMAND]
Application class library management
  -h, --help    Show help information
Commands:
  list      List application class libraries
  add       Add an application class library
  delete    Delete an application class library

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 Options

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

JAR file path, used by `add`.

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

Class library JAR file name, used by `delete`; query with `app class-lib list`.

10.1.9.3 Examples

Example 1 Add and list application class libraries

```
admin@localhost> app class-lib add testapp --file=D:/aas-h2.jar
Class library uploaded.

admin@localhost> app class-lib list testapp
| Class library name | Size      | Version |
+-----+-----+-----+
| aas-h2.jar        | 2.49 MB  | -       |
Page 1 / 1, total 1
```

Example 2 Delete an application class library

```
admin@localhost> app class-lib delete testapp --jar-name=aas-h2.jar -f
Class library deleted.
```

10.1.10 Application incremental update (app increment)

The `app increment` subcommand manages incremental update packages of a specified application, for scenarios where only part of the application files change.

10.1.10.1 Synopsis

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

```
Application incremental update
```

```
  -h, --help    Show help information
```

```
Commands:
```

```
  list      List application incremental packages
```

```
  add       Add an application incremental package
```

```
  delete    Delete an application incremental package
```

```
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 Options

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

Incremental package file path (zip archive), used by `add`.

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

Incremental package description.

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

Incremental package ID, used by `delete`; query with `app increment list`.

10.1.10.3 Examples

Example 1 Add an incremental package

The package is applied to the target application automatically after upload:

```
admin@localhost> app increment add testapp --file=D:/aas-increment.zip
--description=update-web-xml
Increment package uploaded and applied, id: 2
```

Example 2 List application incremental packages

```
admin@localhost> app increment list testapp
| id | Name | Size | Description | Uploaded time |
+---+-----+-----+-----+-----+
| 2 | aas-increment.zip | 236 B | update-web-xml | 2026-09-27 17:46 |
Page 1 / 1, total 1
```

Example 3 Delete an application incremental package

```
admin@localhost> app increment delete testapp --increment-id=2 -f
Increment package deleted.
```

10.1.11 Backing up an application (app backup)

The `app backup` subcommand performs a manual backup of a specified application. Backup files can be used to restore the application.

10.1.11.1 Synopsis

```
Usage: app backup <NAME> [--virtual-host=<host>] [-h]
Back up application
    <name>                Application name
    --virtual-host=<virtualHost> Virtual host name; query with
aascli host list; required when multiple applications share the name
    --help                Show help information
```

10.1.11.2 Example

Example 1 Back up an application

```
admin@localhost> app backup testapp
Backup created.
```

10.1.12 Application backup management (app-backup)

The `app-backup` command manages application backups: view, download, restore, delete, and backup settings.

10.1.12.1 Synopsis

```
Usage:  app-backup [-h] [COMMAND]
Application backup
    -h, --help    Show help information
Commands:
    list          List application backups
    delete       Delete an application backup
    download      Download an application backup
    restore       Restore an application from a backup
    settings      Backup settings
```

10.1.12.2 Examples

Example 1 List application backups

```
admin@localhost> app-backup list
| Backup name | Application name | Context path |
Backup time | Size |
+-----+-----+-----+-----+
-----+-----+
| testapp_1790501743154.tar.gz | testapp | /testapp |
2026-09-27 17:35 | 769 B |
Page 1 / 1, total 1
```

Example 2 Download an application backup

```
admin@localhost> app-backup download testapp_1790501743154.tar.gz --
file=D:/aas-en-backup.tar.gz
Backup downloaded to: D:\aas-en-backup.tar.gz
```

Example 3 Restore an application from a backup

```
admin@localhost> app-backup restore testapp_1790501778335.tar.gz
Application restored from backup.
```

`restore` can override application settings stored in the backup (context path, virtual host, classification level, etc.); unspecified items use the values from the backup:

```
admin@localhost> app-backup restore testapp_1790501778335.tar.gz --
context-path=/testapp2 --class-level=3
Application restored from backup.
```

Restoring while an application with the same name or context path already exists reports a conflict:

```
admin@localhost> app-backup restore testapp_1790501743154.tar.gz
Restore conflict: an application named "testapp" already exists under
virtual host "localhost".
Restore conflict: an application with context path "/testapp" already
exists under virtual host "localhost".
```

In this case, uninstall the conflicting application first, or use `--force` to force the restore.

Example 4 Delete an application backup

```
admin@localhost> app-backup delete testapp_1790501778335.tar.gz -f
Backup deleted.
```

Example 5 View backup settings

```
admin@localhost> app-backup settings get
Backup settings
  Backup directory: ${apusic.home}/store/app_backup
  Retention days: 7
```

Example 6 Modify backup settings

```
admin@localhost> app-backup settings set --
path=${apusic.home}/store/app_backup --retention-days=7
Backup settings saved.
```

`settings set` options:

Option	Description
<code>--path=</code>	Backup directory
<code>--retention-days=</code>	Backup retention days, range 1-365

10.1.13 Application recycle management (app-recycle)

The `app-recycle` command manages the application recycle bin. When the recycle bin is enabled, uninstalled applications are moved to it and can be restored or permanently deleted.

10.1.13.1 Synopsis

```
Usage:  app-recycle [-h] [COMMAND]
Application recycle
  -h, --help    Show help information
Commands:
  list          List recycled applications
  restore       Restore a recycled application
  delete        Permanently delete a recycled application
  settings      Recycle settings
```

10.1.13.2 Examples

Example 1 View recycle settings and enable the recycle bin

```
admin@localhost> app-recycle settings get
Recycle settings
  Recycle enabled: disabled
  Recycle directory: ${apusic.home}/store/app_recycle
  Retention days: 30

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

`settings set` options:

Option	Description
<code>--enabled=true false</code>	Whether to enable the recycle bin
<code>--path=</code>	Recycle directory
<code>--retention-days=</code>	Recycle retention days, range 1-365

Example 2 List recycled applications

```
admin@localhost> app-recycle list
| Recycled file name      | Application name | Context path | Recycled
time      | Size  |
+-----+-----+-----+-----+
-----+-----+
| testapp_20260927173618 | testapp          | /testapp     | 2026-09-
27 17:36 | 791 B |
Page 1 / 1, total 1
```

Example 3 Restore a recycled application

`restore` requires the full recycled file name (not the application name):

```
admin@localhost> app-recycle restore testapp_20260927173618
Application restored.
```

The restore also supports overriding the context path, virtual host, classification level, and other settings; unspecified items use the values stored in the recycle bin.

Example 4 Permanently delete a recycled application

```
admin@localhost> app-recycle delete testapp_20260927173630 -f
Recycled application permanently deleted.
```

10.2 Configuration Management

10.2.1 Virtual host management (host)

The `host` command manages virtual hosts: view, create, update, delete, and set the default virtual host.

10.2.1.1 Synopsis

```
Usage:  host [-h] [COMMAND]
Virtual host management
  -h, --help    Show help information
Commands:
  list          List virtual hosts
  get           View virtual host details
  add           Create a virtual host
```

```

update      Update a virtual host
delete      Delete a virtual host
set-default Set a virtual host as default

```

host add / host update options:

Option	Description
--name=	Unique business identifier of the virtual host (required for add)
--docBase=	Document root path; defaults to webapps when blank; a document root cannot be shared by multiple virtual hosts
--alias=	Host names or IP addresses, separated by commas
--sso=true false	Single sign-on mode; when enabled, the virtual host uses SSO login
--sso-http-only=true false	Set the HttpOnly attribute on the SSO cookie
--access-ip-whitelist=	Allowed IPs; * wildcard and commas supported; blank means no restriction, e.g. 192.168.*.10.0.0.8
--access-ip-blacklist=	Blocked IPs; same format
--access-domain-whitelist=	Allowed domains; * wildcard and commas supported; blank means no restriction, e.g. *.example.com
--access-domain-blacklist=	Blocked domains; same format
--access-time=	Allow client access only during configured periods; blank means no restriction; format HH:MM:SS-HH:MM:SS, multiple periods separated by commas
--extra-props=<header=value,...>	HTTP response header fields, e.g. X-Frame-Options=SAMEORIGIN

10.2.1.2 Examples

Example 1 List virtual hosts

```

admin@localhost> host list
| name      | alias | docBase | default |
+-----+-----+-----+-----+
| admin     |      | admin   | yes     |

```

```
| localhost | | webapps | yes |
```

Page 1 / 1, total 2

Example 2 View virtual host details

```
admin@localhost> host get localhost
Name: localhost
Doc base: webapps
Host names or IP addresses, separated by commas:
Domain:
Single sign-on mode; when enabled, the virtual host uses SSO login;
otherwise SSO is disabled: disabled
Set HttpOnly attribute on SSO cookie: no
Allowed IPs; * wildcard and commas supported; blank means no
restriction, e.g. 192.168.*,10.0.0.8:
Blocked IPs; * wildcard and commas supported; blank means no
restriction:
Allowed domains; * wildcard and commas supported; blank means no
restriction, e.g. *.example.com:
Blocked domains; * wildcard and commas supported; blank means no
restriction:
Allow client access only during configured periods; blank means no
restriction; format HH:MM:SS-HH:MM:SS, multiple periods separated by
commas:
HTTP response header fields, e.g. X-Frame-Options=SAMEORIGIN; header
name can only contain letters, digits, and !#$%&*+-.^_`|~:
```

Example 3 Create a virtual host

```
admin@localhost> host add --name=test-host --docBase=webapps-test --
alias=www.test.com --sso=false
Virtual host created successfully. Restart server to take effect.
```

Example 4 Update the access IP whitelist of a virtual host

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

```
Virtual host saved successfully. Restart server to take effect.
```

Example 5 Set a virtual host as default

```
admin@localhost> host set-default test-host
Are you sure you want to set virtual host test-host as default? (y/N)
y
Virtual host test-host has been set as default.
```

Example 6 Delete a virtual host

```
admin@localhost> host delete test-host -f
Virtual host deleted successfully. Restart server to take effect.
```

10.2.2 Thread pool management (threadpool)

The `threadpool` command manages thread pools: view, create, update, and delete.

10.2.2.1 Synopsis

```
Usage: threadpool [-h] [COMMAND]
Thread pool management
  -h, --help    Show help information
Commands:
  list      List thread pools
  get       View thread pool details
  add       Create a thread pool
  update    Update a thread pool
  delete    Delete a thread pool
```

`threadpool add` options (for `update` the name is a parameter; other options are the same):

Option	Description
<code>--name=</code>	Thread pool name (required)
<code>--min-threads=</code>	Minimum threads (required, 1-65535)
<code>--max-threads=</code>	Maximum threads (required, 1-65535)
<code>--queue-size=</code>	Queue size (required, 1-65535)

--max-idle-time=	Idle timeout in ms (required, 1-86400000)
--priority=	Thread priority (required, 1-10)
--connection-auto-reclaim=	Datasource connection auto reclaim (required: false/reclaim/detect)
--thread-step-reclaim=	Step reclaim (true or false, default false)
--thread-step-size=	Reclaim count per step (required when thread-step-reclaim=true, 1-65535)
--thread-step-time=	Reclaim interval in ms (required when thread-step-reclaim=true, 1-86400000)

10.2.2.2 Examples

Example 1 List thread pools

```
admin@localhost> threadpool list
| thread pool name | max threads | core threads | queue size | idle
time | priority |
+-----+-----+-----+-----+-----+
-----+-----+
| admin-thread-pool | 100          | 25          | 5000        | 60000
ms   | 5           |
| http-thread-pool  | 200          | 25          | 5000        | 60000
ms   | 5           |
Page 1 / 1, total 2
```

Example 2 View thread pool details

```
admin@localhost> threadpool get http-thread-pool
Thread pool name: http-thread-pool
Min threads: 25
Max threads: 200
Queue size: 5000
Max idle time: 60000 ms
Step reclaim: disabled
Data source connection auto reclaim: Disabled
Thread priority: 5
```

Example 3 Create a thread pool

```
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
Thread pool created successfully. Restart server to take effect.
```

Example 4 Update a thread pool

```
admin@localhost> threadpool update test-pool --max-threads=200
Thread pool saved successfully. Restart server to take effect.
```

Example 5 Delete a thread pool

```
admin@localhost> threadpool delete test-pool -f
Thread pool deleted successfully. Restart server to take effect.
```

10.2.3 Session manager management (session-manager)

The `session-manager` command manages session managers, supporting four storage types: memory, file, Redis, and database.

10.2.3.1 Synopsis

```
Usage: session-manager [-h] [COMMAND]
Session manager management
  -h, --help    Show help information
Commands:
  list         List session managers
  get          View session manager details
  add          Create a session manager
  update       Update a session manager (name and type cannot be changed)
  delete       Delete a session manager (session managers bound to
applications cannot be deleted)
```

`session-manager add` main options (for `update` the name and type cannot be changed; other options are the same):

Option	Description
--------	-------------

--name=	Session manager name (required; letters, digits, underscores, hyphens; up to 64 characters)
--type=	Storage type (required): memory, file, redis, database
--max-sessions=	Maximum sessions; -1 means unlimited; default -1
--persist-path=	memory: persistence file path; default SESSIONS.ser
--save-on-restart=	file/database: save sessions on restart (default true)
--max-idle-backup=	file/database: idle backup threshold in seconds (-1 disables; default -1)
--min-idle-swap=	file/database: minimum idle swap threshold in seconds (-1 disables; default -1)
--max-idle-swap=	file/database: maximum idle swap threshold in seconds (-1 disables; default -1)
--persistence-dir=	file: persistence directory
--data-source-name=	database: datasource JNDI name (required for database)
--local-data-source=	database: use application-local datasource (default false)
--redis-mode=	redis: standalone or sentinel; default standalone
--redis-nodes= host:port[,...]	redis: node addresses; one host:port for standalone, comma-separated for sentinel
--redis-master=	redis: sentinel master name (required for sentinel)
--redis-database= <0-15>	redis: database index; default 0
--redis-password=	redis: password, up to 256 characters
--session-persist-policies=	redis: persistence policies SAVE_ON_CHANGE, ALWAYS_SAVE_AFTER_REQUEST, comma-combined; default empty
--redis-connect-timeout=	redis: connect timeout in ms; default 2000
--redis-max-conn=	redis: maximum connections; default 100
--redis-min-idle=	redis: minimum idle connections; default 10
--redis-max-idle=	redis: maximum idle connections; default 8 when blank
--redis-max-wait=	redis: maximum wait to obtain a connection in ms; default 3000

Advanced session attribute filter options (`--session-attribute-name-filter` , `--session-attribute-value-class-name-filter` , `--warn-on-filter-failure` , `--notify-attribute-on-unchanged` , `--notify-binding-on-unchanged`) are documented in `session-manager add --help` .

10.2.3.2 Examples

Example 1 Create a memory-type session manager

```
admin@localhost> session-manager add --name=test-sm --type=memory
Session manager created successfully.
```

Example 2 List session managers

```
admin@localhost> session-manager list
| Name      | Type   | Created Time          | Updated Time          |
+-----+-----+-----+-----+
| test-sm | Memory | 2026-09-27 17:36:37 | 2026-09-27 17:36:37 |
Page 1 / 0, 0 records in total
```

Example 3 View session manager details

```
admin@localhost> session-manager get test-sm
Name: test-sm
Type: Memory
Max Sessions: Unlimited
Persistence File Path: SESSIONS.ser
Session Attribute Name Filter:
Session Attribute Value Class Name Filter:
Warn on Filter Failure: disabled
Notify on Unchanged Attribute: enabled
Notify on Unchanged Binding: disabled
```

Example 4 Delete a session manager

Session managers bound to applications cannot be deleted; unbind them first:

```
admin@localhost> session-manager delete test-sm -f
Session manager deleted successfully.
```

10.2.4 HTTP channel management (httpchannel)

The `httpchannel` command manages HTTP/HTTPS channels (network listeners): view, create, update, and delete.

10.2.4.1 Synopsis

```
Usage:  httpchannel [-h] [COMMAND]
HTTP channel management
  -h, --help    Show help information
Commands:
  list      List HTTP channels
  get       View HTTP channel details
  add       Create an HTTP channel
  update    Update an HTTP channel
  delete    Delete an HTTP channel
```

`httpchannel add` basic options (for `update` the name is a parameter; other options are the same and default to keeping the current configuration):

Option	Description
<code>--name=</code>	Channel name (required); 1-64 characters; letters, digits, underscores, hyphens
<code>--protocol=</code>	IO mode (required): NIO, NIO2, APR
<code>--host=</code>	Listen address; 0.0.0.0 listens on all network interfaces; default 0.0.0.0
<code>--port=</code>	Listen port (required), 1-65535; ports below 1024 require root
<code>--executor=</code>	Worker thread pool (required); query candidates with aascli threadpool list
<code>--redirect-port=</code>	HTTPS port to redirect to when an HTTP request requires encryption, 1-65535
<code>--receive-buffer-size=</code>	Receive buffer size (TCP SO_RCVBUF); default 8192 bytes
<code>--send-buffer-size=</code>	Send buffer size (TCP SO_SNDBUF); default 8192 bytes
<code>--tcp-no-delay=</code>	Set TCP_NODELAY; default true
<code>--connection-timeout=</code>	Connection timeout (TCP SO_TIMEOUT); default 20000 ms
<code>--accept-count=</code>	TCP backlog; default 100
<code>--max-connections=</code>	Maximum concurrent TCP connections; -1 means unlimited; default 8192

--max-keep-alive-requests=	Maximum requests per TCP connection; default 100
--keep-alive-timeout=	TCP connection idle timeout; default 20 seconds
--uri-encoding=	Character encoding for parsing URL paths; default UTF-8
--compression=	Compression mode on/off/force; default on
--compressible-mime-type=	Only resources of the listed MIME types are GZIP-compressed; comma-separated
--compression-min-size=	Responses smaller than this value are not compressed; default 2048 bytes
--max-post-size=	POST body size limit; -1 means unlimited; default 2097152 bytes
--max-parameter-count=	Maximum parameters per request; default 10000
--max-header-count=	Maximum request header fields; default 100
--max-http-header-size=	Total request header size limit; default 8192 bytes
--response-headers=	Custom HTTP response headers, format Name=Value,Name2=Value2

SSL/HTTPS options:

Option	Description
--ssl-enabled=	Enable SSL; required for HTTPS channels; default false
--gm-enabled=	Enable GM (Chinese national cryptography) authentication; requires GM certificates; default false
--tls-protocols=	TLS versions, comma-separated; default TLSv1.2,TLSv1.3
--client-auth=	Client authentication: false=disabled, want=optional, true=required (mutual); default false
--certificate-custom=	Custom certificate; when off, select a certificate store; when on, enter custom store information
--keystore-name=	Keystore name (when custom certificate is off); query with aascli cert list
--keystore-entry=	Keystore certificate entry; query candidates with aascli cert get
--truststore-name=	Truststore name (when custom certificate is off and client authentication is on)
--truststore-entry=	Truststore certificate entry

<code>--keystore-path=</code>	Keystore file path (when custom certificate is on)
<code>--keystore-password=</code>	Keystore access password (when custom certificate is on)
<code>--certificate-format=</code>	Keystore type: JKS, PKCS12; default JKS
<code>--certificate-alias=</code>	Certificate alias
<code>--key-password=</code>	Protection password of the certificate entry
<code>--truststore-path=</code>	Truststore file path (when custom certificate is on and client authentication is on)
<code>--truststore-password=</code>	Truststore access password
<code>--truststore-type=</code>	Truststore type: JKS, PKCS12; default JKS
<code>--ssl-cipher-suites=</code>	Cipher suites for non-GM SSL; query candidates with aascli cert cipher-suites; comma-separated
<code>--http2-enabled=</code>	HTTP/2 requires SSL first; default false

Advanced options for slow-request monitoring (`--long-thread-*`), SDOS protection (`--sdos-*`), and HTTP/2 details (`--http2-*`) are documented in `httpchannel add --help` .

10.2.4.2 Examples

Example 1 List HTTP channels

```
admin@localhost> httpchannel list
| name          | address | port | IO mode |
+-----+-----+-----+-----+
| admin-http    |         | 6848 | NIO     |
| ams-http      |         | 6888 | NIO     |
Page 1 / 1, total 2
```

Example 2 View HTTP channel details

```
admin@localhost> httpchannel get ams-http
Name: ams-http
IO mode: NIO
Listen address:
Listen port: 6888
```

```
Redirect port: 6443
Executor: http-thread-pool
Receive buffer: 字节
Send buffer: 字节
TCP NoDelay: enabled
Connection timeout: 30000 毫秒
Accept count: 100
Processor cache:
Graceful stop: 0 毫秒
Max connections:
Max keep-alive requests: 100
Keep-alive timeout: 15 秒
IP virtual hosts: disabled
Long thread detection: disabled
SDOS detection: disabled
URI encoding: utf-8
Use body encoding for URI: disabled
Default response encoding:
DNS reverse lookup: disabled
Compression:
Compressible MIME types:
Compression min size:
Allow TRACE: disabled
Powered-By: disabled
Async request timeout: 30000 毫秒
Disable upload timeout: enabled
Connection upload timeout: 毫秒
Max POST size: 104857600
Max parameter count:
Max header count: 100
Max header size: 8192 字节
Custom response headers:
Enable SSL: disabled
Enable HTTP/2: disabled
```

Example 3 Create an HTTP channel

```
admin@localhost> httpchannel add --name=test-channel --protocol=NIO --
port=6999 --executor=http-thread-pool
HTTP channel created successfully. Restart server to take effect.
```

Example 4 Update the maximum connections of an HTTP channel

```
admin@localhost> httpchannel update test-channel --max-
connections=4096
HTTP channel saved successfully. Restart server to take effect.
```

Example 5 Delete an HTTP channel

```
admin@localhost> httpchannel delete test-channel -f
HTTP channel deleted successfully. Restart server to take effect.
```

10.2.5 JVM configuration management (jvm)

The `jvm` command views, updates, and resets JVM startup settings (heap sizes, GC algorithm, debug options, etc.).

10.2.5.1 Synopsis

```
Usage:  jvm [-h] [COMMAND]
JVM configuration management
  -h, --help    Show help information
Commands:
  get          View JVM configuration
  update       Update JVM configuration
  reset        Restore default JVM configuration
```

`jvm update` options:

Option	Description
<code>--xms=</code>	Minimum heap size (-Xms), in MB
<code>--xmx=</code>	Maximum heap size (-Xmx), in MB
<code>--metaspace-size=</code>	Initial metaspace size, in MB

<code>--max-metaspace-size=</code>	Maximum metaspace size, in MB
<code>--gc-type=</code>	Garbage collector: ParallelGC, G1GC, ConcMarkSweepGC, ParallelOldGC, ZGC, SerialGC
<code>--debug=</code>	Enable debugging (true or false)
<code>--debug-options=</code>	Debug options, e.g. - agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=8000
<code>--native-library-path=</code>	Native library path; separate multiple paths with semicolons
<code>--jvm-options=</code>	JVM options; separate multiple options with semicolons

10.2.5.2 Examples

Example 1 View JVM configuration

```
admin@localhost> jvm get
JDK path: D:\jdk\jdk-17.0.7
Java version: 17.0.7
Min heap size (-Xms): 1024
Max heap size (-Xmx): 4096
Metaspace initial size:
Metaspace max size:
Garbage collector:
Debug enabled: disabled
Debug options:
Native library path:
JVM options list:
```

Example 2 Update JVM heap settings

```
admin@localhost> jvm update --xms=1024 --xmx=4096
JVM config saved successfully. Restart server to take effect.
```

Example 3 Restore default JVM configuration

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

10.2.6 JMX configuration management (jmx)

The `jmx` command views, updates, and resets JMX remote monitoring settings.

10.2.6.1 Synopsis

```
Usage:  jmx [-h] [COMMAND]
JMX configuration management
  -h, --help    Show help information
Commands:
  get          View JMX configuration
  update       Update JMX configuration
  reset        Restore default JMX configuration
```

`jmx update` main options:

Option	Description
<code>--enabled=</code>	Enable JMX (true or false)
<code>--host=</code>	Listen address; blank defaults to local 127.0.0.1
<code>--port=</code>	Listen port, 1-65535; blank defaults to 6868
<code>--authenticate=</code>	Enable authentication (true or false)
<code>--password-file=</code>	Password file path, e.g. conf/jmxremote.password
<code>--access-file=</code>	Access control file path, e.g. conf/jmxremote.access
<code>--ssl-enabled=</code>	Enable SSL (true or false)
<code>--client-auth=</code>	Client authentication (true or false)
<code>--certificate-custom=</code>	Custom certificate (true or false)
<code>--key-store-file=</code>	Keystore path, e.g. conf/jmx.keystore
<code>--key-password=</code>	Key password
<code>--key-store-password=</code>	Keystore password
<code>--key-store-type=</code>	Keystore type, JKS or PKCS12
<code>--trust-store-file=</code>	Truststore path, e.g. conf/jmx.truststore

<code>--trust-store-password=</code>	Truststore password
<code>--trust-store-type=</code>	Truststore type, JKS or PKCS12
<code>--enabled-cipher-suites=</code>	Cipher suites, comma-separated; query candidates with <code>aascli cert cipher-suites</code>
<code>--key-store-name=</code>	Keystore name (when custom certificate is off); query candidates with <code>aascli cert list</code>
<code>--trust-store-name=</code>	Truststore name (when custom certificate is off)
<code>--key-store-entry=</code>	Keystore entry alias; query candidates with <code>aascli cert get</code>

10.2.6.2 Examples

Example 1 View JMX configuration

```
admin@localhost> jmx get
Enable JMX: disabled
```

Example 2 Enable JMX

```
admin@localhost> jmx update --enabled=true
JMX config saved successfully. Restart server to take effect.
```

Example 3 Restore default JMX configuration

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

10.2.7 Certificate management (cert)

The `cert` command manages certificate stores (keystores/truststores): create, import, download, delete, and entry management.

10.2.7.1 Synopsis

```
Usage:  cert [-h] [COMMAND]
Certificate management
  -h, --help    Show help information
Commands:
  list          List certificate stores
  get           View certificate store details and entries
```

```

add          Create a certificate store
update       Update a certificate store
delete       Delete a certificate store
delete-entry Delete a certificate entry
import       Import entries from a store file
import-pem   Import a PEM certificate or private key
download     Download a certificate store file
cipher-suites List supported cipher suites

```

`cert add` options:

Option	Description
<code>--name=</code>	Certificate store name (required); up to 64 characters; letters, digits, underscores, hyphens, dots; globally unique
<code>--type=</code>	Store type, JKS or PKCS12; default JKS
<code>--store-type=</code>	Usage: keystore or truststore; default keystore
<code>--password=</code>	Store password, 8-32 characters; entered interactively when omitted

`cert import` options:

Option	Description
	Target certificate store name
<code>--file=</code>	Store file path; supports .jks/.p12/.pfx/.pkcs12; up to 10 MB
<code>--source-password=</code>	Source store password; entered interactively when omitted
<code>--aliases=</code>	Aliases to import, comma-separated; omit to import all
<code>--target-alias=</code>	Alias in the target store after import; blank keeps the source alias
<code>--entry-password=</code>	Entry protection password; only for PKCS12 keystores; blank uses the target store password

`cert import-pem` options:

Option	Description
--------	-------------

	Target certificate store name
--file=	PEM file path containing a certificate or private key (with BEGIN/END markers)
--alias=	Alias in the target store after import
--entry-password=	Entry protection password; only for PKCS12 keystores; blank uses the target store password

10.2.7.2 Examples

Example 1 Create a certificate store

```
admin@localhost> cert add --name=test-store --password=Test12345
Certificate store created, id: 3
```

Example 2 List certificate stores

```
admin@localhost> cert list
| name          | usage      | type | path                                     |
+-----+-----+-----+-----+
| test-store | keystore | JKS  | D:\AAMS-V10.0.8\store\ssl\test-
store.jks |
Page 1 / 0, total 0
```

Example 3 Import entries from a store file

```
admin@localhost> cert import test-store --file=D:/aas-source.jks --
source-password=Source123
Certificate entries imported.
```

Example 4 View certificate store details and entries

```
admin@localhost> cert get test-store
name: test-store
path: D:\AAMS-V10.0.8\store\ssl\test-store.jks
usage: Keystore
type: JKS

Certificate entries:
```

alias	entry type	algorithm	valid from	valid to
demo-cert	私钥	SHA256WITHRSA	2026-09-27	2027-09-27

Entry details:

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

Example 5 Download a certificate store file

```
admin@localhost> cert download test-store --file=D:/aas-en-store.jks
Certificate store file downloaded to: D:\aas-en-store.jks
```

When `--file` is omitted, the file is saved as `<store-name>.<jks|p12>` in the current directory.

Example 6 Delete a certificate entry

```
admin@localhost> cert delete-entry test-store demo-cert -f
Certificate entry deleted.
```

Example 7 Delete a certificate store

```
admin@localhost> cert delete test-store -f
Certificate store deleted.
```

Example 8 List supported cipher suites

When configuring SSL cipher suites for HTTP channels or JMX, query the cipher suites supported by the current JDK:

```
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
... (omitted) ...
```

```
TLS_EMPTY_RENEGOTIATION_INFO_SCSV
Total 23 cipher suites.
```

10.3 Resource Management

10.3.1 Datasource management (datasource)

The `datasource` command manages JDBC datasources: create, view, update, enable/disable, connectivity test, and delete.

10.3.1.1 Synopsis

```
Usage:  datasource [-h] [COMMAND]
Datasource management
  -h, --help    Show help information
Commands:
  list      List datasources
  get       View datasource details
  add       Create a datasource
  update    Update a datasource
  delete    Delete a datasource
  toggle    Enable or disable a datasource
  ping      Test datasource connectivity
```

`datasource add` main options (for `update` the name is a parameter; other options are the same):

Option	Description
<code>--name=</code>	Datasource name (required); letters, digits, dots, slashes, hyphens, underscores; up to 128 characters
<code>--db-type=</code>	Database vendor (required): Oracle, MySQL8, MySQL5, PostgreSQL, DM, Kingbase, GaussDB, HighGo, Vastbase, Other
<code>--resource-type=</code>	Resource type (required): javax.sql.DataSource, javax.sql.XADataSource, javax.sql.ConnectionPoolDataSource (default javax.sql.DataSource)
<code>--driver-class=</code>	Driver class name (required); up to 128 characters
<code>--driver-path=</code>	Absolute path of the driver jar on the server, e.g. /opt/drivers/mysql-connector.jar
<code>--status=</code>	Whether to enable immediately after creation (default true)

--class-level=	Classification level (2=Secret, 3=Confidential, 4=Top secret; default 2)
--host=	Database host name or IP; up to 255 characters
--port=	Database port (required, 1-65535)
--database-name=	Database name or schema; up to 64 characters
--username=	Connection username (required); up to 255 characters
--password=	Connection password (required); up to 255 characters
--url=	JDBC URL (required); must start with jdbc;; up to 500 characters
--custom-props=	Custom connection properties in the form key1=value1,key2=value2
--initial-connections=	Initial connections (default 10, range 0-10000)
--max-connections=	Maximum connections (default 20, range 1-10000)
--min-idle=	Minimum idle connections (default 4, range 0-10000)
--max-idle=	Maximum idle connections (default 10, range 1-10000)
--max-wait-ms=	Maximum wait time in ms; -1 means wait indefinitely (default 15000)
--idle-timeout-ms=	Idle timeout in ms (default 60000, range 1-86400000)
--idle-check-period-ms=	Idle check period in ms; 0 disables checking (default 60000)
--max-lifetime-ms=	Maximum lifetime in ms; 0 means unlimited (default 900000)
--validation-query=	Validation SQL; a read-only SELECT statement is recommended (up to 255 characters; dangerous keywords are rejected)
--test-on-borrow=	Validate on borrow (default true)
--test-on-return=	Validate on return (default false)
--test-while-idle=	Validate while idle (default false)
--test-on-connect=	Validate on connect (default false)
--validation-interval-ms=	Validation interval in ms (default 30000)
--connection-auto-reclaim=	Connection auto reclaim (default false)

<code>--leak-recover=</code>	Reclaim leaked connections (default false)
<code>--leak-detection-threshold-seconds=</code>	Leak detection timeout in seconds (default 30, range 1-1800)
<code>--auto-commit=</code>	Auto-commit transactions (default true)
<code>--log-abandoned=</code>	Log connection leaks (default false)
<code>--sql-trace-enabled=</code>	Slow statement tracing (default false)
<code>--slow-sql-ms=</code>	Slow SQL threshold in ms (default 2000, range 1-300000)
<code>--failed-sql-alert-enabled=</code>	Record failed SQL (default false)

Additional options (`--max-check-connections` , `--remove-abandoned-on-maintenance` , `--abandon-when-percentage-full` , `--ha-probe-enabled`) are documented in `datasource add --help` .

10.3.1.2 Examples

Example 1 Create a datasource

```
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
Datasource created, id: 2
```

Example 2 List datasources

```
admin@localhost> datasource list
| Datasource name | resource type          | classification | Driver
class          | status | Connectivity |
+-----+-----+-----+-----+
-----+-----+-----+
| test-ds        | javax.sql.DataSource | Secret        |
org.h2.Driver   | Enabled | Connected    |
Page 1 / 1, total 1
```

Example 3 View datasource details

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

```
[Basic Configuration]
```

```
ID: 4
```

```
Name: test-ds
```

```
DB type: 其他
```

```
Resource type: javax.sql.DataSource
```

```
Driver class: org.h2.Driver
```

```
Driver path: D:/aas-h2.jar
```

```
Status: enabled
```

```
Classification: Secret
```

```
Host:
```

```
Port: 9092
```

```
Database name:
```

```
Username: sa
```

```
Password: *****
```

```
URL: jdbc:h2:mem:testdb
```

```
Custom properties:
```

```
[Pool Configuration]
```

```
Initial connections: 10
```

```
Max connections: 20
```

```
Min idle: 4
```

```
Max idle: 10
```

```
Max wait: 15000 ms
```

```
Idle timeout: 60000 ms
```

```
Idle check period: 60000 ms
```

```
Max lifetime: 900000 ms
```

```
Max check connections: 5
```

```
[Validation Configuration]
```

```
Validation query:
```

```
Validation timeout: 3 s
```

```
Test on borrow: enabled
```

```
Test on return: disabled
```

```
Test while idle: disabled
```

```

Test on connect: disabled
Validation interval: 30000 ms

[Advanced Configuration]
Connection auto reclaim: disabled
Leak recover: disabled
Remove abandoned on maintenance: disabled
Leak detection threshold: 30 s
Abandon when percentage full: 10 %
Auto commit: enabled
Log abandoned: disabled

[Monitoring & Diagnostics]
SQL trace enabled: disabled
Slow SQL threshold: 2000 ms
Failed SQL alert enabled: disabled
HA probe enabled: disabled

```

Note: Some field values returned by the server (e.g. DB type "其他" for Other) are currently displayed in Simplified Chinese regardless of the CLI language, as captured above.

Example 4 Test datasource connectivity

```

admin@localhost> datasource ping test-ds
Connection succeeded: 连接成功, elapsed 1ms

```

Example 5 Disable/enable a datasource

```

admin@localhost> datasource toggle test-ds --status=false
Datasource disabled.

admin@localhost> datasource toggle test-ds --status=true
Datasource enabled.

```

Example 6 Delete a datasource

An enabled datasource cannot be deleted directly; disable it first:

```

admin@localhost> datasource delete test-ds -f
Datasource operation failed: 数据源已启用, 请先禁用后再删除

admin@localhost> datasource toggle test-ds --status=false
Datasource disabled.

admin@localhost> datasource delete test-ds -f
Datasource deleted.

```

10.4 Monitoring Management

10.4.1 Monitoring configuration management (monitor-config)

The `monitor-config` command views, updates, and resets the server monitoring collection settings.

10.4.1.1 Synopsis

```

Usage:  monitor-config [-h] [COMMAND]
Monitoring configuration management
    -h, --help    Show help information
Commands:
    get          View monitoring configuration
    update       Update monitoring configuration
    reset        Restore monitoring configuration defaults

```

`monitor-config update` main options:

Option	Description
<code>--monitor-enabled=</code>	Monitoring service master switch (true/false)
<code>--mbean-monitor-enabled=</code>	MBean monitoring switch (true/false)
<code>--collect-interval-seconds=</code>	Collection interval (seconds), range 10~3600, default 60
<code>--jvm-monitor-enabled=</code>	JVM monitoring switch (true/false)
<code>--thread-pool-monitor-enabled=</code>	Thread pool monitoring switch (true/false)
<code>--datasource-monitor-enabled=</code>	Datasource monitoring switch (true/false)

--http-monitor-enabled=	HTTP channel monitoring switch (true/false)
--session-monitor-enabled=	Application session monitoring switch (true/false)
--system-resource-monitor-enabled=	System resource monitoring switch (true/false)
--fullgc-monitor-enabled=	FullGC monitoring switch (true/false)
--cpu-monitor-enabled=	CPU monitoring switch (true/false)
--memory-monitor-enabled=	Memory monitoring switch (true/false)
--max-data-points=	Total monitoring data points, range 1~2147483647
--replay-sample-interval=	Replay sampling interval (seconds), range 1~86400

10.4.1.2 Examples

Example 1 View monitoring configuration

```
admin@localhost> monitor-config get
Monitoring Configuration:
Monitoring service: disabled
Monitor MBean: disabled
Collection interval (seconds): 60
JVM monitoring: disabled
Thread pool monitoring: disabled
Datasource monitoring: disabled
HTTP channel monitoring: disabled
Application session monitoring: disabled
System resource monitoring: disabled
FullGC monitoring: disabled
CPU monitoring: disabled
Memory monitoring: disabled
Max data points: 11000
Replay sample interval (seconds): 30
```

Example 2 Enable monitoring and change the collection interval

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

```
Monitoring configuration updated.
```

Example 3 Restore monitoring configuration defaults

```
admin@localhost> monitor-config reset
Monitoring configuration restored to defaults.
```

10.4.2 Alert configuration management (alert-config)

The `alert-config` command views, updates, and resets alert settings, including the alert switch, notification channels (email/SNMP/alert platform), and thresholds.

10.4.2.1 Synopsis

```
Usage: alert-config [-h] [COMMAND]
Alert configuration management
  -h, --help    Show help information
Commands:
  get          View alert configuration
  update       Update alert configuration
  reset        Restore alert configuration defaults
```

`alert-config update` main options:

Category	Option	Description
Alert switch	--enabled=	Enable alerts (true/false)
	--interval-minutes=	Alert interval (minutes), range 1~527040
Email	--email-enabled=	Enable email notification (true/false)
	--email-recipients=	Recipients; separate multiple addresses with semicolons
	--email-smtp-host=	SMTP server, e.g. smtp.example.com
	--email-smtp-ssl=	Enable email SSL (true/false)
	--email-smtp-port=	SMTP port, range 1~65535
	--email-smtp-user=	Sender mailbox address

	--email-smtp-password=	Sender mailbox password; blank keeps the current value
SNMP	--snmp-enabled=	Enable SNMP notification (true/false)
	--snmp-version=	SNMP version: v2c or v3
	--snmp-community=	SNMP community (v2c)
	--snmp-address=	SNMP trap target address, e.g. 172.24.4.108:162
	--snmp-security=	SNMP security level (v3): none, authNoPriv, authPriv
	--snmp-username=	SNMP username (v3)
	--snmp-auth-method=	SNMP authentication (v3): MD5 or SHA
	--snmp-auth-password=	SNMP authentication password (v3); blank keeps the current value
	--snmp-cipher=	SNMP encryption (v3): DES, 3DES, AES128, AES192, AES256
	--snmp-priv-password=	SNMP privacy password (v3); blank keeps the current value
Alert platform	--platform-enabled=	Enable alert platform notification (true/false)
	--platform-level=	Alert level, e.g. Warning, Critical
	--platform-url=	Alert platform URL; must start with http:// or https://
Thresholds	--cpu-threshold= / --cpu-consecutive=	CPU usage alert threshold / consecutive occurrences (≥ 1)
	--memory-threshold= / --memory-consecutive=	Memory usage alert threshold / consecutive occurrences
	--fullgc-count-threshold= / --fullgc-count-consecutive=	FullGC count threshold / consecutive occurrences (fixed 1)
	--fullgc-duration-threshold= / --fullgc-duration-consecutive=	FullGC duration threshold (ms) / consecutive occurrences

	--http-timeout-threshold= / --http-timeout-consecutive=	HTTP timeout request threshold / consecutive occurrences
	--http-waiting-threshold= / --http-waiting-consecutive=	HTTP waiting task threshold / consecutive occurrences
	--http-error-threshold= / --http-error-consecutive=	HTTP error threshold / consecutive occurrences
	--jdbc-waiting-threshold= / --jdbc-waiting-consecutive=	JDBC waiting request threshold / consecutive occurrences
	--jdbc-leak-threshold= / --jdbc-leak-consecutive=	JDBC leaked connection threshold / consecutive occurrences

10.4.2.2 Examples

Example 1 View alert configuration

```
admin@localhost> alert-config get
Alert Configuration:
Alert switch: disabled
Alert interval (minutes): 30

Email Notification
Email notification switch: disabled
Recipients:
SMTP server:
Email SSL: enabled
SMTP port:
Sender mailbox:
Sender mailbox password:

SNMP Notification
SNMP notification switch: disabled
SNMP version: v2c
SNMP community: public
SNMP address:
SNMP security level: 无
SNMP username:
```

```
SNMP auth method: SHA
SNMP auth password:
SNMP cipher: AES128
SNMP privacy password:
SNMP engine ID: 0x8000137001AC1806A0FB9B64B9
```

```
Alert Platform Notification
Platform notification switch: disabled
Alert level: 警告
Platform URL:
```

Alert Metrics

Metric	Threshold	Unit	Consecutive count
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

Note: Some values in the alert configuration (security level, alert level, metric names) are returned by the server and are currently displayed in Simplified Chinese regardless of the CLI language, as captured above.

Example 2 Enable alerts and configure email notification

```
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=*****
Alert configuration updated.
```

Example 3 Change the CPU alert threshold

```
admin@localhost> alert-config update --cpu-threshold=90 --cpu-
consecutive=5
Alert configuration updated.
```

Example 4 Restore alert configuration defaults

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

10.4.3 SNMP agent configuration management (snmp-config)

The `snmp-config` command views, updates, and resets the built-in SNMP agent settings, which allow upper-layer NMS systems to collect monitoring data via SNMP.

10.4.3.1 Synopsis

```
Usage: snmp-config [-h] [COMMAND]
SNMP agent configuration management
  -h, --help    Show help information
Commands:
  get          View SNMP agent configuration
  update       Update SNMP agent configuration
  reset        Restore SNMP configuration defaults
```

`snmp-config update` options:

Option	Description
<code>--enabled=</code>	Enable the SNMP service (true/false)
<code>--host=</code>	Bind address, e.g. 0.0.0.0, 192.168.1.1; up to 128 characters
<code>--port=</code>	Bind port, range 1~65535; the server probes port usage on change
<code>--protocol=</code>	Transport protocol: udp or tcp
<code>--version=</code>	SNMP version: v2c or v3
<code>--community=</code>	SNMP community (v2c)
<code>--security=</code>	SNMP security level (v3): none, authNoPriv, authPriv
<code>--username=</code>	SNMP username (v3)

--auth-method=	SNMP authentication (v3): MD5 or SHA
--auth-password=	SNMP authentication password (v3); blank keeps the current value
--cipher=	SNMP encryption (v3): DES, 3DES, AES128, AES192, AES256
--priv-password=	SNMP privacy password (v3); blank keeps the current value

10.4.3.2 Examples

Example 1 View SNMP agent configuration

```
admin@localhost> snmp-config get
SNMP Agent Configuration:
SNMP service switch: disabled
Bind address: 0.0.0.0
Bind port: 161
Transport protocol: udp
SNMP version: v2c
SNMP community: public
```

Example 2 Enable the SNMP agent (v2c)

```
admin@localhost> snmp-config update --enabled=true --host=0.0.0.0 --
port=161 --protocol=udp --version=v2c --community=public
SNMP configuration updated.
```

Example 3 Restore SNMP configuration defaults

```
admin@localhost> snmp-config reset
SNMP configuration restored to defaults.
```

10.4.4 One-click inspection management (inspection)

The `inspection` command runs a one-click health inspection of the server, generates inspection reports, and supports viewing, downloading, deleting reports, and viewing inspection system snapshots.

10.4.4.1 Synopsis

```
Usage: inspection [-h] [COMMAND]
One-click inspection management
```

```

-h, --help    Show help information

Commands:
list          List inspection reports
get           View inspection report details
run           Start a one-click inspection
status        Query inspection report execution status
summary       View inspection report summary
download      Download inspection report archive (only successful
reports)
delete        Delete an inspection report
rerun         Re-run an inspection (only completed or abnormal
reports)
snapshot      View or download inspection system snapshots (jstack,
lsof, netstat, top; lsof is Linux only)
has-running   Query whether an inspection task is running

```

`inspection list` filter options:

Option	Description
<code>--name=</code>	Filter by report name
<code>--status=</code>	Filter by status: RUNNING, COMPLETED, TIMEOUT, ABNORMAL
<code>--start-time=</code>	Start time, format yyyy-MM-dd HH:mm:ss
<code>--end-time=</code>	End time, format yyyy-MM-dd HH:mm:ss

`inspection run` options:

Option	Description
<code>--timeout-seconds=</code>	Inspection timeout (seconds), range 1~172800, default 600

`inspection snapshot` arguments:

Argument	Description
	Report ID; query with aascli inspection list
	Snapshot type: jstack, lsof, netstat, top (lsof is Linux only)

`--file=`

Save path; when omitted, the snapshot content is printed directly

10.4.4.2 Examples

Example 1 Query whether an inspection task is running

```
admin@localhost> inspection has-running
No inspection task is running
```

Example 2 Start a one-click inspection

Inspections run asynchronously in the background; query the progress with the report ID:

```
admin@localhost> inspection run
Inspection submitted (ID: 2, name: InspectionReport_20260927173757),
running in background
Check progress with: aascli inspection status 2
```

Example 3 Query inspection status

```
admin@localhost> inspection status 2
Inspecting

admin@localhost> inspection status 2
Completed
```

Example 4 List inspection reports

```
admin@localhost> inspection list
| ID | Inspection result | Status | Timeout
(seconds) | Date |
+---+-----+-----+-----+
--+-----+
| 2 | InspectionReport_20260927173757 | Inspecting | 600
| 2026-09-27 17:37:57 |
Page 1 / 1, total 1
```

Example 5 View the inspection report summary

```
admin@localhost> inspection summary 2
Instances with CPU load over 80%: 0
Instances with memory usage over 80%: 0
Instances with heap usage over 80%: 0
Instances with disk usage over 80%: 1
```

Example 6 Download the inspection report archive

```
admin@localhost> inspection download 2 --file=D:/aas-en-inspection.zip
Downloaded: D:\aas-en-inspection.zip
```

Example 7 Re-run an inspection

Only completed or abnormal reports can be re-inspected:

```
admin@localhost> inspection rerun 2
```

Example 8 View an inspection system snapshot

```
admin@localhost> inspection snapshot 2 jstack
```

Example 9 Delete an inspection report

```
admin@localhost> inspection delete 2 -f
Inspection report deleted.
```

10.4.5 Monitoring snapshot management (snapshot)

The `snapshot` command manages monitoring snapshots: manual creation, download, deletion, and automatic generation policies.

10.4.5.1 Synopsis

```
Usage:  snapshot [-h] [COMMAND]
Monitoring snapshot management (manual creation, download, deletion,
automatic configuration)
    -h, --help    Show help information
Commands:
    list          List monitoring snapshots
```

```

get      Query snapshot details by exact name
create   Create a monitoring snapshot manually (generated
asynchronously)
delete   Delete a monitoring snapshot
download Download monitoring snapshot archive (only completed
snapshots)
config   Automatic snapshot configuration

```

`snapshot list` filter options:

Option	Description
<code>--name=</code>	Filter by snapshot name
<code>--status=</code>	Filter by status: pending, running, done, failed
<code>--start-time=</code>	Start time, format yyyy-MM-dd HH:mm:ss
<code>--end-time=</code>	End time, format yyyy-MM-dd HH:mm:ss

`snapshot create` options:

Option	Description
<code>--contents=<a,b,...></code>	Snapshot contents (required), comma-separated: accessLog, gcLog, systemConfig, systemLog, hostMonitor, instanceMetrics, coreDump, threadDump, heapDump
<code>--thread-stack-count=</code>	Thread stack capture count, range 1~10 (recommended when threadDump is included)
<code>--heapdump-mode=</code>	Heapdump mode: all or live (recommended when heapDump is included)

`snapshot config` subcommands:

```

Usage:  snapshot config [-h] [COMMAND]
Automatic snapshot configuration
  -h, --help  Show help information
Commands:
  get      View automatic snapshot configuration
  update   Update automatic snapshot configuration
  reset    Restore automatic snapshot configuration defaults

```

`snapshot config update` main options:

Category	Option	Description
Basic	<code>--auto-enabled=</code>	Enable automatic snapshots (true/false)
	<code>--max-snapshots=</code>	Maximum snapshots to keep, range 1~1000
	<code>--interval-minutes=</code>	Generation interval (minutes), range 1~1440
Contents	<code>--content-access-log=</code>	Collect access logs (true/false)
	<code>--content-gc-log=</code>	Collect GC logs (true/false)
	<code>--content-system-config=</code>	Collect system configuration (true/false)
	<code>--content-system-log=</code>	Collect system logs (true/false)
	<code>--content-host-monitor=</code>	Collect host monitoring data (true/false)
	<code>--content-instance-metrics=</code>	Collect instance metrics (true/false)
	<code>--content-core-dump=</code>	Collect system coredumps (true/false)
	<code>--content-thread-dump=</code>	Collect thread stacks (true/false)
	<code>--thread-stack-count=</code>	Thread stack capture count, range 1~10
	<code>--content-heap-dump=</code>	Collect heapdumps (true/false)
	<code>--heapdump-mode=</code>	Heapdump mode: all or live
Trigger rules	<code>--rule-cpu-threshold= / --rule-cpu-occurrences=</code>	CPU usage threshold (%) / consecutive occurrences
	<code>--rule-memory-threshold= / --rule-memory-occurrences=</code>	Memory usage threshold (%) / consecutive occurrences
	<code>--rule-fullgc-count-threshold= / --rule-fullgc-count-occurrences=</code>	FullGC count threshold / consecutive occurrences
	<code>--rule-fullgc-time-threshold= / --rule-fullgc-time-occurrences=</code>	FullGC time threshold (ms) / consecutive occurrences
	<code>--rule-http-timeout-threshold= / --rule-http-timeout-occurrences=</code>	HTTP slow request threshold / consecutive occurrences
	<code>--rule-http-wait-threshold= / --rule-http-wait-occurrences=</code>	HTTP waiting task threshold / consecutive occurrences

	--rule-http-error-threshold= / --rule-http-error-occurrences=	HTTP error threshold / consecutive occurrences
	--rule-jdbc-wait-threshold= / --rule-jdbc-wait-occurrences=	JDBC waiting request threshold / consecutive occurrences
	--rule-jdbc-leak-threshold= / --rule-jdbc-leak-occurrences=	JDBC suspected leaked connection threshold / consecutive occurrences

10.4.5.2 Examples

Example 1 Create a monitoring snapshot manually

Snapshots are generated asynchronously; check the progress in the list:

```
admin@localhost> snapshot create --contents=systemLog,threadDump --
thread-stack-count=1
Snapshot task submitted and is being generated in the background.
Use aascli snapshot list to check the generation progress.
```

Example 2 List monitoring snapshots

```
admin@localhost> snapshot list
| ID | Name | Type | Status | Contents
| Created At |
+-----+-----+-----+-----+-----+
-----+-----+
| 2 | snapshot-20260927-173806-manual | Manual | Done |
systemLog,threadDump | 2026-09-27 17:38:06 |
Page 1 / 1, total 1
```

Example 3 Query snapshot details by exact name

```
admin@localhost> snapshot get snapshot-20260927-173806-manual
```

Example 4 Download a monitoring snapshot archive

```
admin@localhost> snapshot download 2 --file=D:/aas-en-snapshot.zip
Snapshot downloaded: D:\aas-en-snapshot.zip
```

Example 5 Delete a monitoring snapshot

```
admin@localhost> snapshot delete 2 -f
Snapshot deleted.
```

Example 6 View automatic snapshot configuration

```
admin@localhost> snapshot config get
Automatic Snapshot Config:

Automatic Snapshot
Auto snapshot switch: disabled
Max snapshots: 10
Interval (minutes): 5

Snapshot Contents
Access log: enabled
GC log: disabled
System config: disabled
System log: enabled
Host monitor data: disabled
Instance metrics: disabled
System coredump: disabled
Thread stack: enabled
Thread stack count: 3
Heapdump: enabled
Heapdump mode: live

Trigger Rules
CPU usage threshold (%): 80.0
CPU consecutive occurrences: 3
Memory usage threshold (%): 80.0
Memory consecutive occurrences: 3
FullGC count threshold: 5
FullGC count consecutive occurrences: 1
FullGC time threshold (ms): 1000
FullGC time consecutive occurrences: 2
HTTP slow request threshold: 1
```

```

HTTP slow request consecutive occurrences: 2
HTTP waiting task threshold: 1
HTTP waiting task consecutive occurrences: 3
HTTP error threshold: 3
HTTP error consecutive occurrences: 2
JDBC waiting request threshold: 1
JDBC waiting request consecutive occurrences: 3
JDBC suspected leaked connection threshold: 1
JDBC suspected leaked connection consecutive occurrences: 2

```

Example 7 Update automatic snapshot configuration

```

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

```

Example 8 Restore automatic snapshot configuration defaults

```

admin@localhost> snapshot config reset

```

10.5 Log Service

10.5.1 System log configuration (log-config system)

The `log-config system` command views and updates the server system log configuration.

10.5.1.1 Synopsis

```

Usage:  log-config system [-h] [COMMAND]
System log configuration
  -h, --help    Show help information
Commands:
  get          View system log configuration
  update       Update system log configuration

```

`log-config system update` main options:

Category	Option	Description
----------	--------	-------------

Basic	--global-level=	Global log level; default INFO; options: SEVERE/WARNING/INFO/CONFIG/FINE/FINER/FINEST/ALL
	--class-loggers=	Custom class-level settings; independent log levels for packages or classes; comma-separated; format: package.Class=LEVEL
	--console-enabled=	Output logs to the console; default true
	--file-enabled=	Write logs to files; default true
File	--file-prefix=	Log file name prefix; default apusic.; up to 100 characters
	--file-suffix=	Log file name suffix; default .log; up to 100 characters
	--encoding=	Log file encoding; default UTF-8; options: UTF-8/GBK
	--file-timestamp-format=	Date format in log file names; default yyyy-MM-dd
Pattern	--console-pattern=	Console log pattern, e.g. "%d{yyyy-MM-dd HH:mm:ss.SSS} %level [%t] %logger %msg%n"
	--file-pattern=	File log pattern; same format
Rotation	--retention-count=	Retain log files by day; older logs are deleted automatically; default 30; 0 or negative means unlimited
	--max-size=	Rotate a log file when it reaches this size, in MB; default 100; 0 or negative means unlimited
	--rotate-by-date=	Create a new log file every day; default true
Compression	--compress-enabled=	Enable automatic log compression and archiving; default false
	--compress-time=	Time of day to compress logs; default 02:00; format HH:mm
	--compress-dir=	Directory for compressed files; default \${apusic.base}/logs/archive; variables such as \${apusic.base} supported

10.5.1.2 Examples

Example 1 View system log configuration

```

admin@localhost> log-config system get

[Basic Configuration]
Global log level: FINE
Custom class-level loggers:
Console log output: enabled
File log output: enabled

[File Configuration]
Log file prefix: apusic.
Log file suffix: .log
Log file encoding: UTF-8
Log file date format: yyyy-MM-dd

[Pattern Configuration]
Console log pattern: %d{yyyy-MM-dd HH:mm:ss.SSS} %level [%t] %logger
%msg%n
File log pattern: %d{yyyy-MM-dd HH:mm:ss.SSS} %level [%t] %logger
%msg%n

[Rotation Configuration]
Retention days: 30
Rotate by size: 0 MB
Rotate by date: enabled

[Compression Configuration]
Log compression: disabled

```

Example 2 Change the global log level

```

admin@localhost> log-config system update --global-level=INFO
Log configuration updated successfully.

```

Example 3 Set an independent log level for a package

```

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

```

```
Log configuration updated successfully.
```

10.5.2 Access log configuration (log-config access)

The `log-config access` command views and updates the HTTP access log configuration.

10.5.2.1 Synopsis

```
Usage: log-config access [-h] [COMMAND]
Access log configuration
  -h, --help    Show help information
Commands:
  get          View access log configuration
  update       Update access log configuration
```

`log-config access update` main options:

Category	Option	Description
Basic	--enabled=	Enable access logs (true/false)
	--virtual-hosts=	Application virtual hosts, comma-separated; query candidates with aascli host list
	--directory=	Log file path; variables such as \${apusic.base} supported
	--encoding=	Log file encoding: UTF-8/GBK; default UTF-8
	--buffer-enabled=	Write buffer (true/false)
	--buffer-size=	Buffer size in bytes; default 32768
	--file-date-format=	Date format in log file names, e.g. .yyyy-MM-dd
	--max-days=	Retention days; default 30; 0 or negative means unlimited
Format	--custom-format-enabled=	Enable custom format (true/false)
	--log-content=	Log content fields, comma-separated; options: time, remoteAddr, method, protocol, status, uri, bytesSent, sessionId, threadName,

		serverName, localPort
	--pattern=	Custom access log format using placeholders such as %a (remote IP), %t (time), e.g. "%h %l %u %t \"%r\" %s %b"
	--date-format=	Date format for the time field (the xxx in %{xxx}t); default dd/MM/yyyy:HH:mm:ss
	--request-headers=	Request headers to log, comma-separated
	--response-headers=	Response headers to log, comma-separated
	--cookies=	Cookies to log, comma-separated
	--session-attrs=	Session attributes to log, comma-separated
Rotation & compression	--rotate-by-date=	Rotate daily (true/false)
	--max-size=	Rotate by size, in MB; 0 means unlimited
	--compress-enabled=	Log compression (true/false)
	--compress-time=	Compression time, format HH:mm; default 00:00
	--compress-dir=	Compression directory; variables such as \${apusic.base}/logs/archive supported

10.5.2.2 Examples

Example 1 Enable access logs

```
admin@localhost> log-config access update --enabled=true
Log configuration updated successfully.
```

Example 2 View access log configuration

```
admin@localhost> log-config access get
[Basic Configuration]
Enable access log: enabled
Virtual hosts: localhost
```

```

Log file path: ${apusic.base}/logs/localhost_access_log
Log file encoding: UTF-8
Write buffer: enabled
Buffer size: 32768
Log file date format: .yyyy-MM-dd
Retention days: 30

[Log Pattern Configuration]
Enable custom format:
Log content: time, remoteAddr, method, uri, status
Date format: dd/MM/yyyy:HH:mm:ss
Request headers:
Response headers:
Cookies:
Session attributes:

[Log Rotation Configuration]
Rotate by date: enabled
Rotate by size: 30 MB

[Log Compression Configuration]
Log compression: disabled

```

Example 3 Disable access logs

```

admin@localhost> log-config access update --enabled=false
Log configuration updated successfully.

```

10.6 User Management

10.6.1 User management (user)

The `user` command manages console users: create, view, delete, and view permissions. Role assignment and permission configuration are performed by the security administrator; the system administrator can view user permission settings.

10.6.1.1 Synopsis

```
Usage:  user [-h] [COMMAND]
User management
  -h, --help    Show help information
Commands:
  list          List users
  get           View user details
  add           Create a user
  delete        Delete a user
  permission    User permission configuration
```

`user add` options:

Option	Description
<code>--username=</code>	Username (required); 1-64 characters; letters, digits, underscores, hyphens
<code>--password=</code>	Login password; can be specified in scripts; entered interactively with confirmation when omitted
<code>--status=</code>	User status: 0=disabled, 1=normal, 2=locked; default 1
<code>--email=</code>	Email address; valid email format; up to 255 characters
<code>--phone=</code>	Mobile number; Chinese mobile number 1[3-9]XXXXXXXXXX
<code>--description=</code>	Description; up to 255 characters

10.6.1.2 Examples

Example 1 Create a user

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

Note: A newly created user has no role assigned yet; logging in at this point fails with `Login failed: 用户未授权, 请联系管理员`. The security administrator must assign a role before the user can log in.

Example 2 List users

```

admin@localhost> user list
| username | status | roles      | classification | expire date |
privileged |
+-----+-----+-----+-----+-----+-----+
-----+
| testuser | Enabled |           | Secret         | Permanent   | no
|
| admin    | Enabled | sysadmin  | Top secret     | Permanent   | yes
|
| secure   | Enabled | security  | Top secret     | Permanent   | yes
|
| audit    | Enabled | auditor   | Top secret     | Permanent   | yes
|
Page 1 / 1, total 4

```

Example 3 View user details

```

admin@localhost> user get testuser
ID: 8
Username: testuser
Status: Enabled
Classification: Secret
Privileged identifier: no
Roles:
Email:
Phone:
OTP: no
Expire date: Permanent
Created time: 2026-09-27 17:46:14
Description: CLI-test-user

```

Example 4 View user permission settings

```

admin@localhost> user permission get testuser
User Permission Configuration:
Username: testuser

```

```
Status: Enabled
Assigned Role:
Expire date: Permanent
Allowed IPs:
Access Start Time:
Access End Time:
Data Class Level: Secret
Phone:
Email:
Description: CLI-test-user
```

Example 5 Delete a user

```
admin@localhost> user delete testuser -f
User deleted successfully.
```

10.7 Security Service

10.7.1 Security domain management (security-domain)

The `security-domain` command manages security domains (Realms), which provide authentication and authorization data sources for applications. Four types are supported: datasource, user database, JAAS, and JNDI (LDAP, etc.).

10.7.1.1 Synopsis

```
Usage:  security-domain [-h] [COMMAND]
Security domain management
  -h, --help    Show help information
Commands:
  list        List security domains
  get         View security domain details
  add         Create a security domain
  update      Update a security domain
  delete      Delete a security domain
```

`security-domain add` basic options (for `update` the name is a parameter; the type stays unchanged when not provided):

Option	Description
--name=	Security domain name (required); letters, digits, underscores, hyphens; up to 128 characters
--type=	Security domain type (required): DataSourceRealm, UserDatabaseRealm, JAASRealm, JNDIRealm
--description=	Description; up to 500 characters; < and > are not allowed
--password-hash-algorithm=	Password comparison algorithm: blank (plaintext), MD5, SHA-1, SHA-256, SHA-512
--lock-enabled=	Enable login failure lockout (true/false; default false)
--max-failed-attempts=	Allowed failed attempts before locking when lockout is enabled (default 5)
--lock-timeout=	Lockout timeout (seconds) before automatic unlock (default 300)

Type-specific options:

Type	Main options
DataSourceRealm	--data-source-jndi (datasource JNDI name, required), --user-table, --username-field, --password-field, --role-table, --role-field
UserDatabaseRealm	--resource-name (UserDatabase resource configured in apusic.conf, required)
JAASRealm	--user-principal-class (required), --role-principal-class (required), --jaas-module (LoginModule configuration: loginModuleClass@controlFlag@options; separate multiple modules with semicolons)
JNDIRealm	--context-factory, --jndi-connection-url (required, e.g. ldap://localhost:389), --connection-username, --connection-password, --user-search-base, --user-search-filter (e.g. (uid={0})), --role-search-base, --role-search-filter (e.g. (member={0})), --user-subtree, --role-subtree, etc.

Complete option descriptions for each type are available in `security-domain add --help`.

10.7.1.2 Examples

Example 1 Create a UserDatabase security domain

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

Example 2 List security domains

```
admin@localhost> security-domain list
| Security domain name | Type | Bound apps | Lock enabled |
|
+-----+-----+-----+-----+
| test-realm | UserDatabaseRealm | -- | disabled |
|
Page 1 / 1, total 1
```

Example 3 View security domain details

```
admin@localhost> security-domain get test-realm
[Basic Configuration]
ID: 3
Security domain name: test-realm
Security domain type: UserDatabaseRealm
Description:
Password hash algorithm:

[Lock Mechanism]
Lock enabled: disabled

[UserDatabase Configuration]
Resource name: UserDatabase
```

Example 4 Update a security domain

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

Example 5 Delete a security domain

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

Example 6 Bind a security domain to an application

After creating a security domain, bind it to an application with `app update` :

```
admin@localhost> app update testapp --security-domain=test-realm --
reload
Application configuration updated.
Application reloaded.
```

10.8 System and Account

10.8.1 User settings (usersettings)

The `usersettings` command views or modifies the personal settings of the current user: paging, language, history size, and profile.

10.8.1.1 Synopsis

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

User settings

<code>-h, --help</code>	Show help information
<code>--history-size=<historySize></code>	History size
<code>-l, --list</code>	View current settings
<code>--lang=<language></code>	Language: en / zh / zh_TW
<code>--no-paging</code>	Disable paging output
<code>--page-size=<pageSize></code>	Page size

Commands:

`profile` View or modify personal profile

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

View or modify personal profile

<code>--description=<description></code>	Description
<code>--email=<email></code>	Email address
<code>-h, --help</code>	Show help information
<code>--phone=<phone></code>	Mobile number

10.8.1.2 Options

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

View current settings.

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

Interface language of the CLI: en / zh / zh_TW. Takes effect the next time you log in to aascli.

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

Records per page for list queries.

```
--no-paging
```

Disable paging output; list commands output all records at once.

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

Number of history commands to keep.

10.8.1.3 Examples

Example 1 View current settings

```
admin@localhost> usersettings -l
Current configuration:
  Paging: enabled
  Page size: 20
  Language: en
  History size: 1,000
```

Example 2 Change page size and history size

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

Example 3 View profile

```
admin@localhost> usersettings profile
Username: admin
Phone: -
```

```
Email: -
Description: -
```

Example 4 Modify profile

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

10.8.2 Changing the current user's password (changepassword)

The `changepassword` command changes the password of the current logged-in user.

10.8.2.1 Synopsis

```
Usage: changepassword [-h] [--new-password=<newPassword>] [--old-
password=<oldPassword>]
Change the password of the current logged-in user
  -h, --help                Show help information
  --new-password=<newPassword> New password; entered
interactively when not provided
  --old-password=<oldPassword> Old password; entered
interactively when not provided
```

10.8.2.2 Options

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

The current user's old password. Entered interactively when not provided (input is not echoed).

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

New password. Entered interactively with confirmation when not provided (input is not echoed). The new password must satisfy the password complexity policy.

10.8.2.3 Examples

Example 1 Change the password interactively

```
admin@localhost> changepassword
旧密码:
```

```
新密码:  
确认新密码:  
密码修改成功, 请重新 login 登录。
```

The current session becomes invalid after a successful change; log in again with `login` .

Example 2 Specify the old and new passwords

```
admin@localhost> changepassword --old-password=Old@12345 --new-  
password=New@67890  
密码修改成功, 请重新 login 登录。
```

11 Security Administrator Module

The security administrator logs in to aascli with the built-in account `secure` (role security) and is responsible for role management, user authorization, and security policy configuration. The commands visible to the security administrator are:

```
secure@localhost> help

User management:
  user          用户管理

Role management:
  role          角色管理

Security service:
  security-config 安全配置管理

System and account:
  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
```

`usersettings`, `changepassword`, `logout`, and `help` are common account commands with the same usage as for the system administrator; see "System and Account".

The division of user management duties between the system administrator and the security administrator:

Operation	System administrator (admin)	Security administrator (secure)
Create user (user add)	Yes	No
List/view users	Yes	Yes
Delete user (user delete)	Yes	Yes
Reset user password (user reset-password)	No	Yes
Assign/unassign user roles (user permission update)	No (view only)	Yes

11.1 User Management

11.1.1 User management (user)

The `user` command manages users. The security administrator can view and delete users, reset user passwords, and configure user permissions (role binding, access control, etc.).

11.1.1.1 Synopsis

```
Usage:  user [-h] [COMMAND]
User management
  -h, --help    Show help information
Commands:
  list          List users
  get           View user details
  delete        Delete a user
  reset-password Reset a user password
  permission    User permission configuration
```

11.1.2 Listing users (user list)

Same usage as for the system administrator, showing username, status, roles, classification, and other information.

11.1.2.1 Example

Example 1 List users

```
secure@localhost> user list
| username | status | roles      | classification | expire date |
privileged |
+-----+-----+-----+-----+-----+-----+
-----+
| admin   | Enabled | sysadmin  | Top secret    | Permanent   | yes
|
| secure  | Enabled | security  | Top secret    | Permanent   | yes
|
| audit   | Enabled | auditor   | Top secret    | Permanent   | yes
|
Page 1 / 1, total 3
```

11.1.3 Viewing user details (user get)

Same usage as for the system administrator.

11.1.3.1 Example

Example 1 View user details

```
secure@localhost> user get admin
ID: 1
Username: admin
Status: Enabled
Classification: Top secret
Privileged identifier: yes
Roles: sysadmin
Email:
Phone:
OTP: no
Expire date: Permanent
Created time: 2026-09-27 11:23:52
Description:
```

11.1.4 Resetting a user password (user reset-password)

The `user reset-password` subcommand resets the password of a specified user, for use when a user forgets the password or a password is compromised.

11.1.4.1 Synopsis

```
Usage: user reset-password <ID> [--password=<password>] [-h]
Reset a user password
    <id>                User ID
    --password=<password> New password; can be specified in
scripts; entered interactively with confirmation when omitted
    --help              Show help information
```

11.1.4.2 Options

```
<ID>
```

User ID; query with `user get <username>` . Note the parameter is the user ID, not the username.

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

New password; must satisfy the current password policy. Entered interactively with confirmation when not specified (input is not echoed).

11.1.4.3 Example

Example 1 Reset a user password

```
secure@localhost> user reset-password 12 --password=New@67890
Password reset successfully.
```

11.1.5 Deleting a user (user delete)

Same usage as for the system administrator. Unbind the user's role before deleting.

11.1.5.1 Example

Example 1 Delete a user

```
secure@localhost> user delete testuser2 -f
User deleted successfully.
```

11.1.6 User permission configuration (user permission)

The `user permission` subcommand configures user permissions: role binding, user status, account validity period, access IP whitelist, access time window, data classification level, etc. It is the core operation of the security administrator.

11.1.6.1 Synopsis

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

```
User permission configuration
```

```
-h, --help    Show help information
```

```
Commands:
```

```
get          View user permission configuration
```

```
update       Update user permission configuration
```

```
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]
Update user permission configuration
    <username>                Username; query with aascli user
list
    --status=<status>          User status: 0=disabled,
1=normal, 2=locked
    --role=<role>              Role code to assign; see `aascli
role list`, or the user's current role code via `aascli user
permission get <USERNAME>`
    --unbind-role              Unbind the user's current role;
mutually exclusive with --role
    --expire-date=<expireDate> Expiry time, format yyyy-MM-dd
HH:mm:ss; blank means permanent
    --ip-whitelist=<ipWhitelist> Allowed IPs, comma-separated;
blank means no restriction
    --start-time=<startTime>    Access start time, format yyyy-
MM-dd HH:mm:ss
    --end-time=<endTime>        Access end time, format yyyy-MM-
dd HH:mm:ss
    --class-level=<classLevel>  Data classification level:
2=Secret, 3=Confidential, 4=Top secret
    --phone=<phone>             Mobile number; Chinese mobile
number 1[3-9]XXXXXXXXXX
    --email=<email>             Email address; valid email
format; up to 255 characters
    --description=<description> Description; up to 255 characters
    --help                     Show help information

```

11.1.6.2 Options

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

User status: 0=disabled, 1=normal, 2=locked. Disabled users cannot log in.

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

Role code to assign; query available role codes with `role list`. Only users with an assigned role can log in to the console or aascli.

```
--unbind-role
```

Unbind the user's current role; mutually exclusive with `--role`. The user cannot log in after unbinding.

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

Account expiry time, format yyyy-MM-dd HH:mm:ss; blank means permanent. Expired accounts cannot log in. Date-time values containing spaces must be quoted.

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

Access IP whitelist, comma-separated; blank means no restriction. Only IPv4/IPv6 addresses or * wildcards (e.g. 192.168.1.*) are supported; CIDR notation (e.g. 192.168.1.0/24) is not.

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

Access start/end times, format yyyy-MM-dd HH:mm:ss, restricting the user to the specified time window.

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

User data classification level: 2=Secret, 3=Confidential, 4=Top secret.

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

Contact information and description. Unlike `usersettings profile`, these options can maintain other users' information.

11.1.6.3 Examples

Example 1 Assign a role and set the account validity period and classification level

After the system administrator creates a user, the security administrator must assign a role before the user can log in:

```
secure@localhost> user permission update testuser2 --role=test-role --
expire-date="2027-12-31 23:59:59" --class-level=3
```

```
User permission configuration saved.

secure@localhost> user permission get testuser2
User Permission Configuration:
Username: testuser2
Status: Enabled
Assigned Role: test-role
Expire date: 2027-12-31 23:59:59
Allowed IPs:
Access Start Time:
Access End Time:
Data Class Level: Confidential
Phone:
Email:
Description:
```

Example 2 Set the user access IP whitelist

Only IPv4/IPv6 addresses or * wildcards are supported:

```
secure@localhost> user permission update testuser2 --ip-
whitelist=192.168.1.*,10.0.0.8
User permission configuration saved.
```

CIDR notation is rejected with a message:

```
secure@localhost> user permission update testuser2 --ip-
whitelist=192.168.1.0/24
User operation failed: 允许访问IP格式非法, 只允许IPv4、IPv6地址或*通配符
```

Example 3 Disable/enable a user

```
secure@localhost> user permission update testuser2 --status=0
User permission configuration saved.

secure@localhost> user permission update testuser2 --status=1
User permission configuration saved.
```

Example 4 Unbind the user's role

```
secure@localhost> user permission update testuser2 --unbind-role
User permission configuration saved.
```

11.2 Role Management

11.2.1 Role management (role)

The `role` command manages roles: create, view, update, delete, and query available permission resources. A role is a collection of permissions and is bound to users via `user permission update --role`.

11.2.1.1 Synopsis

```
Usage:  role [-h] [COMMAND]
Role management
  -h, --help    Show help information
Commands:
  list          List roles
  get           View role details
  add           Create a role
  update        Update a role
  delete        Delete a role
  permissions   List available permission resources
```

11.2.2 Listing roles (role list)

11.2.2.1 Synopsis

```
Usage:  role list [--page=<page>] [--output=<format>] [--no-paging] [-h]
List roles
  --page=<page>      Page number
  --help            Show help information
  --no-paging        Disable paging for this command
  --output=<output>  Output format: table or json
```

11.2.2.2 Example

Example 1 List roles

```
secure@localhost> role list
| Role name | Bound users | Privileged | Description
|
+-----+-----+-----+-----+
-----+
| sysadmin | admin      | yes       | System maintenance and user
creation |
| security | secure     | yes       | Security policy and
authorization |
| auditor  | audit      | yes       | Operation supervision and
audit      |
Page 1 / 1, total 3
```

sysadmin, security, and auditor are built-in privileged roles that cannot be modified or deleted.

11.2.3 Viewing role details (role get)

11.2.3.1 Synopsis

```
Usage: role get <ROLE_CODE> [--output=<format>] [-h]
View role details
    <roleCode>           Role code; query with aascli role list
    --output=<output>    Output format: json
    --help               Show help information
```

11.2.3.2 Example

Example 1 View role details

```
secure@localhost> role get test-role
ID: 6
Role code: test-role
Role name: test-role
Description: CLI-test-role
Builtin: no
Data class level: Public
Permissions: dashboard:view, app_list:view
```

11.2.4 Listing available permission resources (role permissions)

The `role permissions` subcommand lists all available permission resources and their codes in the system. The values of the `--permissions` parameter used when creating or updating roles come from this list.

11.2.4.1 Synopsis

```
Usage: role permissions [--output=<format>] [-h]
List available permission resources
    --help                Show help information
    --output=<output>     Output format: json
```

11.2.4.2 Example

Example 1 List available permission resources

The output is a JSON tree of permission resources; each resource contains `code` (permission code), `name` (name), and `children` (operation-level permissions such as view, create, update, delete):

```
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",
  ...
}, ... ]
```

Top-level permission resource codes:

Code	Name
------	------

dashboard	Home overview
user_mgmt	User management
app_mgmt	Application management
role_mgmt	Role management
config_mgmt	Configuration management
monitor_mgmt	Monitoring management
resource_mgmt	Resource management
log_service	Log service
security_service	Security service
system_mgmt	System management
audit_log	Audit log
security_config	Security configuration
ops_mgmt	O&M inspection

11.2.5 Creating a role (role add)

11.2.5.1 Synopsis

```
Usage: role add --name=<name> --permissions=<codes> [--description=
<desc>] [-h]
Create a role
      --name=<name>                Role name, required; letters,
digits, underscores, hyphens; up to 64 characters
      --permissions=<permissions>  Permission resource codes, comma-
separated, required; see `aascli role permissions`
      --description=<description>  Description; up to 255 characters
      --help                        Show help information
```

11.2.5.2 Options

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

Role name (role code); letters, digits, underscores, hyphens; up to 64 characters.

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

Permission resource codes, comma-separated, required. Both top-level resource codes (e.g. app_mgmt) and operation-level codes (e.g. app_list:view) can be used.

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

Role description; up to 255 characters.

11.2.5.3 Example

Example 1 Create a role

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

11.2.6 Updating a role (role update)

11.2.6.1 Synopsis

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

Update a role

<roleCode>	Role code
--permissions=<permissions>	Permission resource codes, comma-separated; see `aascli role permissions`
--description=<description>	Description; up to 255 characters
--help	Show help information

11.2.6.2 Example

Example 1 Update role permissions and description

`role update` must include `--permissions`, and the permission list fully replaces the previous one (permissions not included are removed):

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

Specifying only `--description` without `--permissions` is rejected:

```
secure@localhost> role update test-role --description=updated-desc
Role operation failed: 请至少为角色分配一个权限资源
```

11.2.7 Deleting a role (role delete)

11.2.7.1 Synopsis

```
Usage: role delete <ROLE_CODE> [-f] [-h]
Delete a role
    <roleCode>    Role code
    -f, --force    Do not prompt for confirmation
    --help        Show help information
```

11.2.7.2 Examples

Example 1 Delete a role

```
secure@localhost> role delete test-role -f
Role deleted successfully.
```

Example 2 Deleting a built-in role is rejected

```
secure@localhost> role delete sysadmin -f
Builtin roles cannot be deleted.
```

11.3 Security Configuration Management

The `security-config` command configures security policies: password policy, audit log settings, business configuration backup, mail verification, and OTP verification.

11.3.1 Synopsis

```
Usage: security-config [-h] [COMMAND]
Security configuration management
    -h, --help    Show help information
Commands:
    password-policy Password policy configuration
```

```

audit-log      Audit log configuration
backup         Business configuration backup management
mail           Mail verification configuration
otp            OTP verification configuration

```

11.3.2 Password policy configuration (security-config password-policy)

The `password-policy` subcommand views and updates the password policy for console accounts.

11.3.2.1 Synopsis

```

Usage:  security-config password-policy [-h] [COMMAND]
Password policy configuration
  -h, --help    Show help information
Commands:
  get          View password policy configuration
  update       Update password policy configuration

```

`security-config password-policy update` options:

Option	Description
<code>--password-min-length=</code>	Minimum password length, range 8~32, default 8
<code>--password-valid-days=</code>	Mandatory password change cycle (days), range 1~90, default 30
<code>--password-max-failed-attempts=</code>	Maximum consecutive login failures, range 1~5, default 5
<code>--password-complexity=</code>	Password complexity: normal (at least 2 of letters/digits/special characters), complex (all 3 required); default normal
<code>--password-history-limit=</code>	Number of historical passwords that cannot be reused, range 1~10, default 5
<code>--password-lock-duration=</code>	Account lock duration (minutes), range 1~1440, default 15
<code>--weak-passwords=</code>	Weak password list, comma-separated, up to 255 characters, default empty

11.3.2.2 Examples

Example 1 View password policy configuration

```
secure@localhost> security-config password-policy get
Password policy:
  Password length: 8 characters
  Password valid days: 30 days
  Password retry attempts: 5 times
  Password complexity: normal
  Recent password repeat limit: 5 times
  User lock duration: 15 minutes
  Weak password setting configuration:
```

Example 2 Raise password strength requirements

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

secure@localhost> security-config password-policy get
Password policy:
  Password length: 10 characters
  Password valid days: 30 days
  Password retry attempts: 5 times
  Password complexity: complex
  Recent password repeat limit: 5 times
  User lock duration: 15 minutes
  Weak password setting configuration:
```

11.3.3 Audit log configuration (security-config audit-log)

The `audit-log` subcommand views and updates the retention and scheduled backup policies of audit logs (login logs and operation logs). Querying and exporting audit logs is performed by the audit administrator.

11.3.3.1 Synopsis

```
Usage: security-config audit-log [-h] [COMMAND]
Audit log configuration
  -h, --help    Show help information
Commands:
```

```
get      View audit log configuration
update  Update audit log configuration
```

```
security-config audit-log update options:
```

Option	Description
--login-log-max-count=	Login log count limit (entries), range 1000~100000, default 10000
--login-log-retention=	Login log retention (days), range 30~365, default 180
--operation-log-max-count=	Operation log count limit (entries), range 1000~100000, default 10000
--operation-log-retention=	Operation log retention (days), range 30~365, default 180
--audit-backup-period-days=	Audit log scheduled backup period (days), range 1~365, default 30
--audit-backup-retention-days=	Audit log backup retention (days); -1 means keep forever; default -1

11.3.3.2 Examples

Example 1 View audit log configuration

```
secure@localhost> security-config audit-log get
Audit log configuration:
  Login log max count: 10000 entries
  Login log retention: 180 days
  Operation log max count: 10000 entries
  Operation log retention: 180 days
  Scheduled backup period: 30 days/time
  Scheduled backup retention: forever
```

Example 2 Change log retention

```
secure@localhost> security-config audit-log update --login-log-
retention=90 --operation-log-retention=90
Security configuration saved.
```

11.3.4 Mail verification configuration (security-config mail)

The `mail` subcommand views and updates login mail verification settings. When enabled, users must enter an email verification code at login.

11.3.4.1 Synopsis

```
Usage: security-config mail [-h] [COMMAND]
Mail verification configuration
  -h, --help    Show help information
Commands:
  get          View mail verification configuration
  update       Update mail verification configuration
```

`security-config mail update` options:

Option	Description
<code>--smtp-verify-enabled=</code>	Mail verification switch: true or false; default false. When enabled, login requires an email verification code, and OTP verification is disabled automatically
<code>--smtp-host=</code>	SMTP server, e.g. smtp.example.com; default empty
<code>--smtp-ssl=</code>	SMTP SSL switch: true or false; default false
<code>--smtp-port=</code>	SMTP port, range 1~65535; default 25 (usually 465 with SSL)
<code>--smtp-user=</code>	Sender mailbox account; default empty
<code>--smtp-password=</code>	Sender mailbox password or client authorization code; default empty
<code>--smtp-license-email=</code>	Recipients of license reminders, semicolon-separated; default empty

11.3.4.2 Examples

Example 1 View mail verification configuration

```
secure@localhost> security-config mail get
Mail verification configuration:
  Mail verification switch: false
  SMTP server:
  Enable SSL: false
  SMTP port: 25
  Sender email account:
```

```
Sender email password:
License reminder email:
```

Example 2 Configure and enable mail verification

Before enabling mail verification, configure a working SMTP service; otherwise no user will be able to log in:

```
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
Security configuration saved.
```

11.3.5 OTP verification configuration (security-config otp)

The `otp` subcommand views and updates login OTP (one-time password) verification settings. When enabled, users must enter both the password and an Authenticator OTP code at login.

11.3.5.1 Synopsis

```
Usage: security-config otp [-h] [COMMAND]
OTP verification configuration
  -h, --help    Show help information
Commands:
  get          View OTP verification configuration
  update       Update OTP verification configuration

Usage: security-config otp update --enabled=<true|false> [-h]
Update OTP verification configuration
  --enabled=<true|false>  OTP verification switch: true or false.
When enabled, login requires both the password and an Authenticator
OTP code, and mail verification is disabled automatically
  --help                Show help information
```

11.3.5.2 Examples

Example 1 View OTP verification configuration

```
secure@localhost> security-config otp get
OTP configuration:
  OTP switch: false
```

Example 2 Enable/disable OTP verification

OTP verification and mail verification are mutually exclusive; enabling one automatically disables the other:

```
secure@localhost> security-config otp update --enabled=true
Security configuration saved.

secure@localhost> security-config otp update --enabled=false
Security configuration saved.
```

11.3.6 Business configuration backup management (security-config backup)

The `backup` subcommand manages backup and restore of business configuration: scheduled backup policy, immediate manual backup, backup listing, restore, and deletion.

11.3.6.1 Synopsis

```
Usage: security-config backup [-h] [COMMAND]
Business configuration backup management
  -h, --help    Show help information
Commands:
  get           View business configuration backup settings
  update        Update business configuration backup settings
  backup-now    Run a business configuration backup immediately
  list          List business configuration backups
  restore       Restore a business configuration backup
  delete        Delete a business configuration backup
```

`security-config backup update` options:

Option	Description
<code>--backup-enabled=</code>	Scheduled backup switch: true or false; default false
<code>--backup-path=</code>	Shared backup directory; system variables supported; default <code>\${apusic.home}/store/config_bak</code>

--backup-period-months=

Scheduled backup period (months), range 1~12, default 1

security-config backup restore options:

Option	Description
	Backup file name; query with <code>security-config backup list</code>
<code>-f, --force</code>	Do not prompt for restore confirmation
<code>-r, --restart</code>	Restart immediately after a successful restore without asking

11.3.6.2 Examples

Example 1 View business configuration backup settings

```
secure@localhost> security-config backup get
Business config backup settings:
  Scheduled backup switch: false
  Shared backup directory: ${apusic.home}/store/config_bak
  Scheduled backup period: 1 months/time
```

Example 2 Enable scheduled backup

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

Example 3 Run a business configuration backup immediately

```
secure@localhost> security-config backup backup-now
Backup succeeded: 业务配置备份成功
```

Example 4 List business configuration backups

```
secure@localhost> security-config backup list
| Backup file name | Backup type | Backup time |
| File size |
+-----+-----+-----+
-----+-----+
```

```
| config_backup_manual_20260927_174930.zip | Manual | 2026-09-27  
17:49:30 | 36.33 KB |  
Page 1 / 1, total 1
```

Example 5 Restore a business configuration backup

Restoring overwrites the current business configuration with the backup content; restart the server after the restore:

```
secure@localhost> security-config backup restore  
config_backup_manual_20260927_174930.zip
```

Example 6 Delete a business configuration backup

```
secure@localhost> security-config backup delete  
config_backup_manual_20260927_174930.zip -f  
Backup deleted.
```

UNIFIED SERVICE HOTLINE

4008-555-800

Kingdee Apusic Cloud Computing Co., Ltd. (referred to as “Kingdee Apusic”) was founded in 2000. Its predecessor was “Kingdee Middleware Company.” It is a basic software platform provider under Kingdee Group, a national standards-setting enterprise for cloud computing, and a core software enterprise in the national IT application innovation industry. Kingdee Apusic is an enterprise undertaking the National 863 Key R&D Program and the National Major Science and Technology Project for Core Electronic Components, High-end General-purpose Chips and Fundamental Software, and is also a basic platform provider for the national key projects of “Two Networks, One Website, Four Databases, and Twelve Golden Projects.” Its products are widely used in key industries such as government, military, finance, and energy, and it has cumulatively served more than 100,000 customers.



National standard setter for cloud computing
Basic software enterprise under Kingdee Group
Core enterprise for IT application innovation
Website: www.apusic.com

