



APUSIC
固若长城
睿比世界

快速使用手册

金蝶Apusic分布式缓存 V2.0.5

版权声明

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

免责声明

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

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

商标声明

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

目录

- 1 前言
 - 1.1 适用对象
 - 1.2 相关文档
 - 1.3 技术支持
- 2 缓存核心快速开始
 - 2.1 前提条件
 - 2.2 快速上手步骤
 - 2.2.1 获取AMDC产品包
 - 2.2.2 上传至目标服务器并解压
 - 2.2.3 导入授权文件
 - 2.2.4 启动AMDC服务
 - 2.2.5 使用amdc-cli工具进行数据读写
 - 2.2.6 使用Java语言进行数据读写
 - 2.2.7 安全加固（强烈建议在首次启动后立即配置）
- 3 Web管控制台快速开始
 - 3.1 登录控制台
 - 3.2 1. 资源准备
 - 3.2.1 1.1 上传安装包
 - 3.2.2 1.2 上传 License
 - 3.2.3 1.3 添加目标机器
 - 3.2.4 1.4 确认配置模板
 - 3.3 2. 自动部署新服务
 - 3.3.1 2.1 基础配置
 - 3.3.2 2.2 拓扑配置
 - 3.3.3 2.3 配置模板
 - 3.3.4 2.4 确认部署
 - 3.4 3. 导入已有实例
 - 3.4.1 3.1 进入导入页面
 - 3.4.2 3.2 填写导入信息
 - 3.4.3 3.3 关联机器
 - 3.5 4. 服务验证

1 前言

本文档为金蝶Apusic分布式缓存（AMDC）V2.0.5产品的快速使用手册，提供产品介绍，安装部署产品的基本步骤及验证方式。

1.1 适用对象

本文档适用于AMDC产品运维工程师、IT系统运维工程师、开发工程师、软件架构师及研发经理等人员。

1.2 相关文档

了解更多AMDC V2.0.5产品相关的信息，请参阅以下AMDC V2.0.5产品手册文档集：

序号	手册文档	说明
1	金蝶Apusic分布式缓存 V2.0.5 快速使用手册	简单介绍了如何快速上手使用AMDC。
2	金蝶Apusic分布式缓存 V2.0.5 安装手册	详细介绍如何在各操作系统上安装AMDC，以及AMDC服务启停操作和产品注册过程。
3	金蝶Apusic分布式缓存 V2.0.5 缓存核心用户手册	详细介绍 AMDC 相关功能的使用、配置、管理及配套工具的使用方法。
4	金蝶Apusic分布式缓存 V2.0.5 管控台用户手册	详细介绍AMDC管控台相关功能的使用和操作说明。
5	金蝶Apusic分布式缓存 V2.0.5 开发手册	详细介绍基于各开发语言进行AMDC客户端应用开发的说明。
6	金蝶Apusic分布式缓存 V2.0.5 迁移手册	详细介绍AMDC历史版本迁移升级到V2.0.5版本的说明，以及Redis迁移到AMDC的说明。
7	金蝶Apusic分布式缓存 V2.0.5 运维手册	详细介绍AMDC的监控、运维、安全加固等运维说明。
8	金蝶Apusic分布式缓存 V2.0.5 性能优化手册	详细介绍AMDC性能调优的说明。

1.3 技术支持

AMDC产品提供全面的技术支持服务，您可以通过以下方式获得技术支持：

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

您在取得技术支持时, 请提供如下信息:

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

2 缓存核心快速开始

本快速使用指南将帮助您在几分钟内：

1. 成功部署并启动 AMDC 服务；
2. 使用默认配置连接到 AMDC 实例；
3. 执行基本的读写操作验证功能；
4. 使用Java 客户端连接AMDC服务。

完成本指南后，您将拥有一个可运行的 AMDC实例，并能通过命令行或客户端进行基础交互。

2.1 前提条件

请确保您的系统满足以下要求：

- 操作系统：Linux（Windows Subsystem for Linux（WSL）推荐）
- CPU架构：x86_64、ARM64
- Java 17+（如 AMDC 客户端为 Java 编写）或对应运行时环境（根据实际技术栈调整）
- 网络端口 6359（AMDC 默认端口）未被占用

2.2 快速上手步骤

2.2.1 获取AMDC产品包

- x86_64 CPU架构： `AMDC-V2.0.5-20260604-amd64.tar.gz`
- ARM64 CPU架构： `AMDC-V2.0.5-20260604-arm64.tar.gz`

请根据目标服务器CPU架构选择对应的产品包。

注：产品包名称中【20260604】日期可能不一致，请以实际日志为准。

2.2.2 上传至目标服务器并解压

下面以产品包： `AMDC-V2.0.5-20260604-amd64.tar.gz` 为例

- 上传 `AMDC-V2.0.5-20260604-amd64.tar.gz` 至目标服务器的 `/opt` 目录下
- 进入/opt目录： `cd /opt`
- 解压： `tar -zxvf AMDC-V2.0.5-20260604-amd64.tar.gz`
- 进入AMDC目录： `cd amdc`

2.2.3 导入授权文件

AMDC产品需要授权文件才能顺利启动，请联系对应的AMDC销售人员/服务支持人员获取最新的测试授权文件。

下面以测试授权文件压缩包：**金蝶Apusic分布式缓存软件2.0.zip** 为例

- 获取授权文件压缩包：`金蝶Apusic分布式缓存软件2.0.zip`
- 上传至目标服务器的 `/opt/amdc` 目录下
- 解压授权文件：`unzip 金蝶Apusic分布式缓存软件2.0.zip`
- 将KBC开头的授权文件名修改为 `license.lic`：`mv KBCLIC-251216-000155.lic license.lic`

2.2.4 启动AMDC服务

- 启动AMDC服务: `./amdc-server amdc.yaml`

```
root@root:/opt/amdc$ ./amdc-server amdc.yaml
[2026-06-04 14:09:59.775 NOTICE] AMDC version=2.0.5, bits=64,
commit=00000000, modified=1, pid=324619, just started
[2026-06-04 14:09:59.777 NOTICE] AMDC KBC authorization code:
'SZTY1952469739' for mac: AC-DC-CA-7B-2E-02
[2026-06-04 14:09:59.779 NOTICE] KBC: license used to test users
[2026-06-04 14:09:59.779 NOTICE] License file currently loaded by
server: /opt/changzhi/AMDC./src/license.lic
[2026-06-04 14:09:59.779 NOTICE] License restriction detected,
overwrite maxmemory configuration from 0B to 5.59G due to license
limitations
[2026-06-04 14:09:59.779 NOTICE] licensee=深圳市金蝶天燕云计算股份有限公
司, duration=2026-05-06 00:00:00 to 2026-08-04 23:59:59, ip=0,
maxclients=0, maxmemory=6G, CPUs=0

      _____ _                               Apusic In-Memory Data Cache
    /   |         _          ( ) _____
  /  /| | /  ___ \ / / / /  ___ / /  ___ /
00:00:00                                     License start: 2026-05-06
    /  ___ | /  _/ / /  _/ (___ ) /  _/
23:59:59                                     License end: 2026-08-04
    /_ /   |_ / . ___ /\ __ , _ / ___ /_ / \ ___ /
      /_ /                                         IP granted: 0
                                           Licensee: 深圳市金蝶天燕云计算股
份有限公司

Version: v2.0.5   Bind: 127.0.0.1 / -::1   Port: 6359   PID:
324619
```

```
[2026-06-04 14:09:59.779 NOTICE] Server initialized
[2026-06-04 14:09:59.779 NOTICE] Loading RDB produced by version
2.0.5
[2026-06-04 14:09:59.779 NOTICE] RDB age 53 seconds
[2026-06-04 14:09:59.779 NOTICE] RDB memory usage when created 0.78
Mb
[2026-06-04 14:09:59.779 NOTICE] Done loading RDB, keys loaded: 0,
keys expired: 0.
[2026-06-04 14:09:59.779 NOTICE] DB loaded from disk: 0.000 seconds
[2026-06-04 14:09:59.780 NOTICE] Ready to accept connections tcp
```

如下所示，AMDC监听端口为6359，AMDC服务启动成功。

2.2.5 使用amdc-cli工具进行数据读写

- 启动amdc-cli工具： `./amdc-cli`

```
root@root:/opt/amdc$ ./amdc-cli
127.0.0.1:6359> info    # 输入info命令，查看AMDC服务信息
# Server
amdc_version:2.0.5
amdc_git_sha1:00000000
amdc_git_dirty:1
amdc_build_id:4d9754b8e1459059
amdc_build_date:2026-05-21
amdc_pid:324619
amdc_mode:standalone
os:Linux 5.10.0-281.0.0.184.oe2203sp4.aarch64 aarch64
arch_bits:64
monotonic_clock:ARM CNTVCT @ 100 ticks/us

# 以下内容略
```

- 创建一个key： `set test "hello world"`

```
127.0.0.1:6359> set test "hello world"
OK
```

- 获取key的值: `get test`

```
127.0.0.1:6359> get test
"hello world"
```

2.2.6 使用Java语言进行数据读写

- AMDC兼容Redis SDK: Jedis
- 添加依赖

```
<dependency>
  <groupId>redis.clients</groupId>
  <artifactId>jedis</artifactId>
  <version>4.3.1</version>
</dependency>
```

- 建立连接，并写入数据

```
package com.robin.test;

import redis.clients.jedis.Jedis;

public class JedisDemo1 {

    private Jedis jedis;

    // 测试链接
    void SetUp() {
        // 1.建立链接
```

```
jedis = new Jedis("127.0.0.1", 6359);  
}  
  
// 测试String  
void testString(){  
  
    String res = jedis.set("name", "apusic");  
    System.out.println("res"+res);  
  
    String name = jedis.get("name");  
    System.out.println("name="+name);  
  
}  
  
// 释放资源  
void tearDown(){  
  
    if(jedis!=null){  
        jedis.close();  
    }  
}  
  
public static void main(String[] args) {  
  
    JedisDemo1 demo1 = new JedisDemo1();  
    demo1.SetUp();  
    demo1.testString();  
    demo1.tearDown();  
  
}  
}
```

- 其他编程语言可参考Redis客户端SDK使用说明

2.2.7 安全加固（强烈建议在首次启动后立即配置）

AMDC 默认无密码、绑定所有接口，**严禁直接暴露于公网或非信任网络。**

1. 设置访问密码

- 编辑 `amdc.yaml`（编辑 `requirepass`）：

```
SECURITY:
  # user: "<user> <acl rules...>"
  user: ""
  # ACL 日志的最大条目数
  acllog-max-len: 128
  # 外部 ACL 用户文件路径，与配置文件内用户配置不可混用
  aclfile: ""
  # 默认用户的密码（等价于 ACL 的 default 用户密码）
  requirepass: "PASSWORD" # 修改此行的密码
```

- 重启AMDC服务
- 测试： `./amdc-cli -a PASSWORD`

2. 限制监听地址

- 编辑 `amdc.yaml`（编辑 `bind`）：

```
NETWORK:
  # 监听的网络地址，默认仅监听本地回环地址，注释此行可监听所有接口
  （暴露在公网有安全风险）
  bind:
    - "127.0.0.1" # 修改此行的IP地址为当前机器IP
    - "-::1"
```

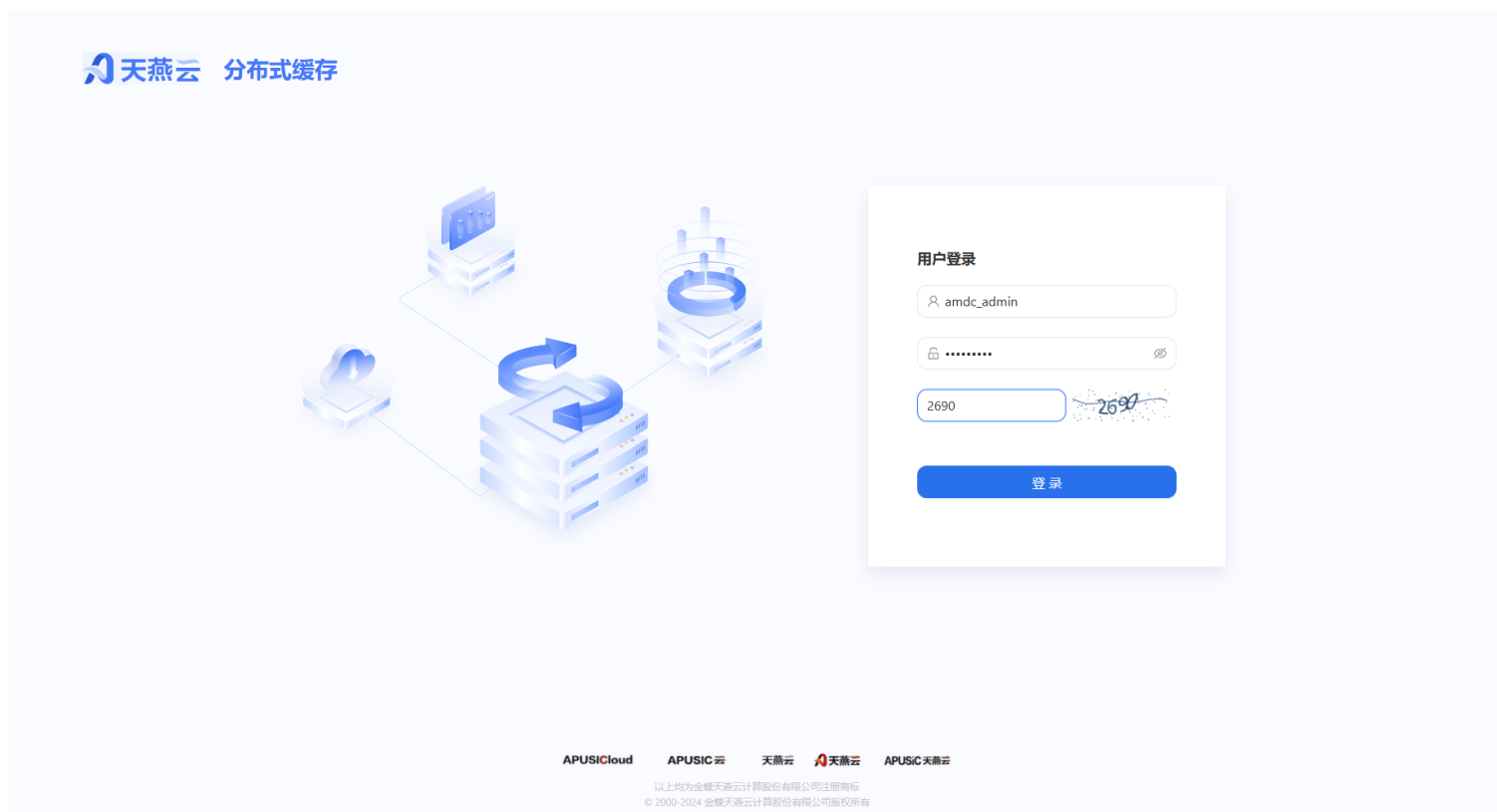
- 重启AMDC服务
- 测试： `./amdc-cli -h IP`

3 Web管控台快速开始

下面以 **AMDC 管理员** 视角，演示服务上线与已有实例导入的快速操作流程。按本流程操作，一名不了解管控台的实施人员也能独立完成部署与导入验证。

3.1 登录控制台

1. 打开浏览器，访问 `http://<serverIP>:9001/`，进入登录页面。
2. 输入账号 `amdc_admin` 和对应密码：
 - 管控台首次启动时，会自动生成所有预置管理员账号的初始密码（包括 `amdc_admin`、三员管理员及租户管理员），并写入安装目录下的 `./resource/.initial_passwords` 文件。
 - 打开该文件，找到 `amdc_admin:` 后的 12 位随机密码作为首次登录密码。
 - 首次登录成功后，系统会强制要求按当前密码策略修改密码，修改完成后方可进入控制台主界面。密码过期后再次登录也会触发强制修改。
3. 点击【登录】进入控制台首页。



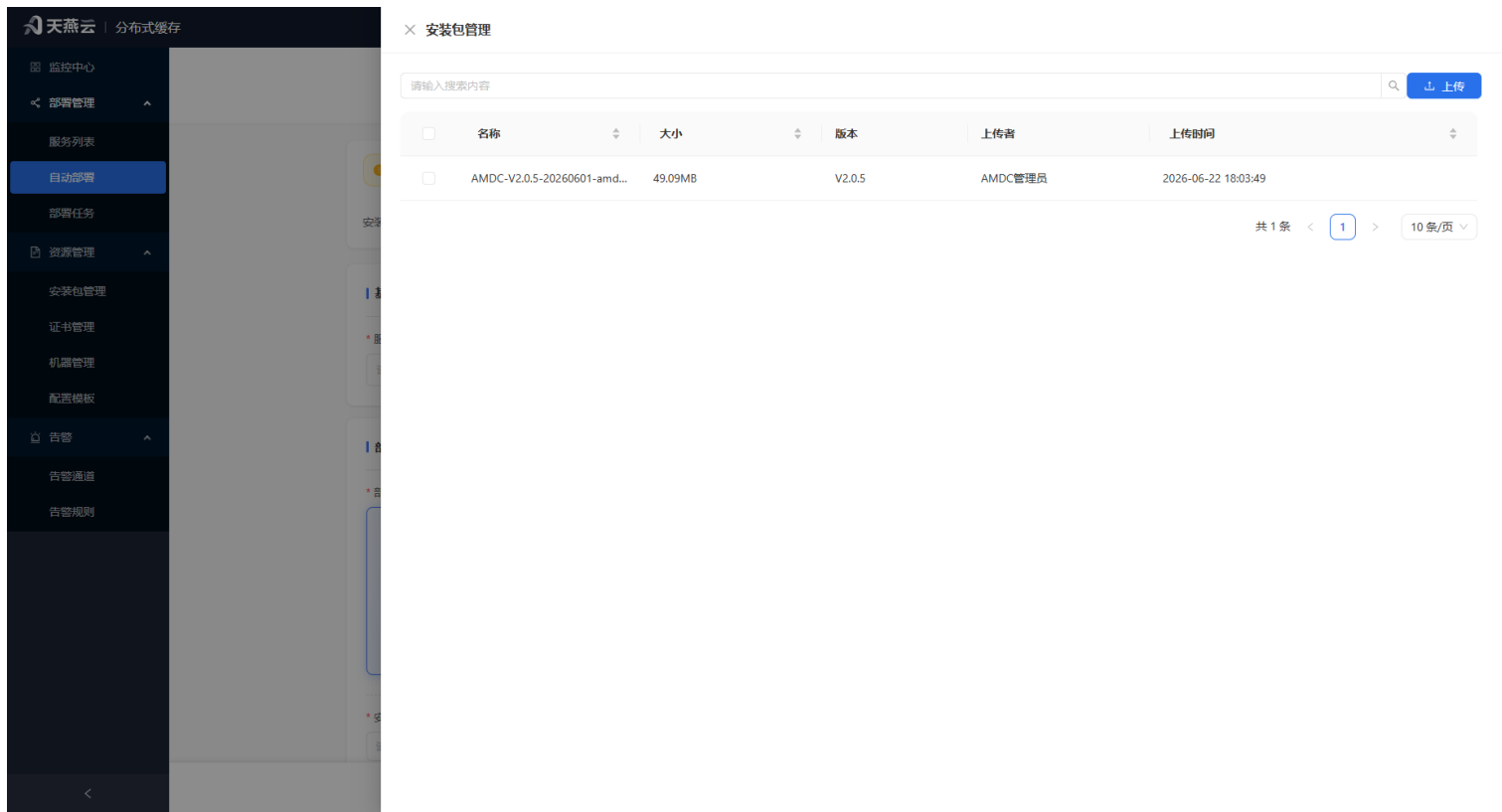
说明：本文档后续操作均基于 **AMDC 管理员** 角色。其他角色（系统管理员、安全管理员、审计管理员、租户管理员）的菜单范围可能不同，请以实际登录角色为准。

3.2 1. 资源准备

部署服务前，需要先准备好安装包、License、机器和配置模板。在自动部署流程中，可直接在【基础配置】页面顶部点击对应【查看】按钮，在弹出的抽屉中完成上传或添加，无需提前进入【资源管理】菜单。

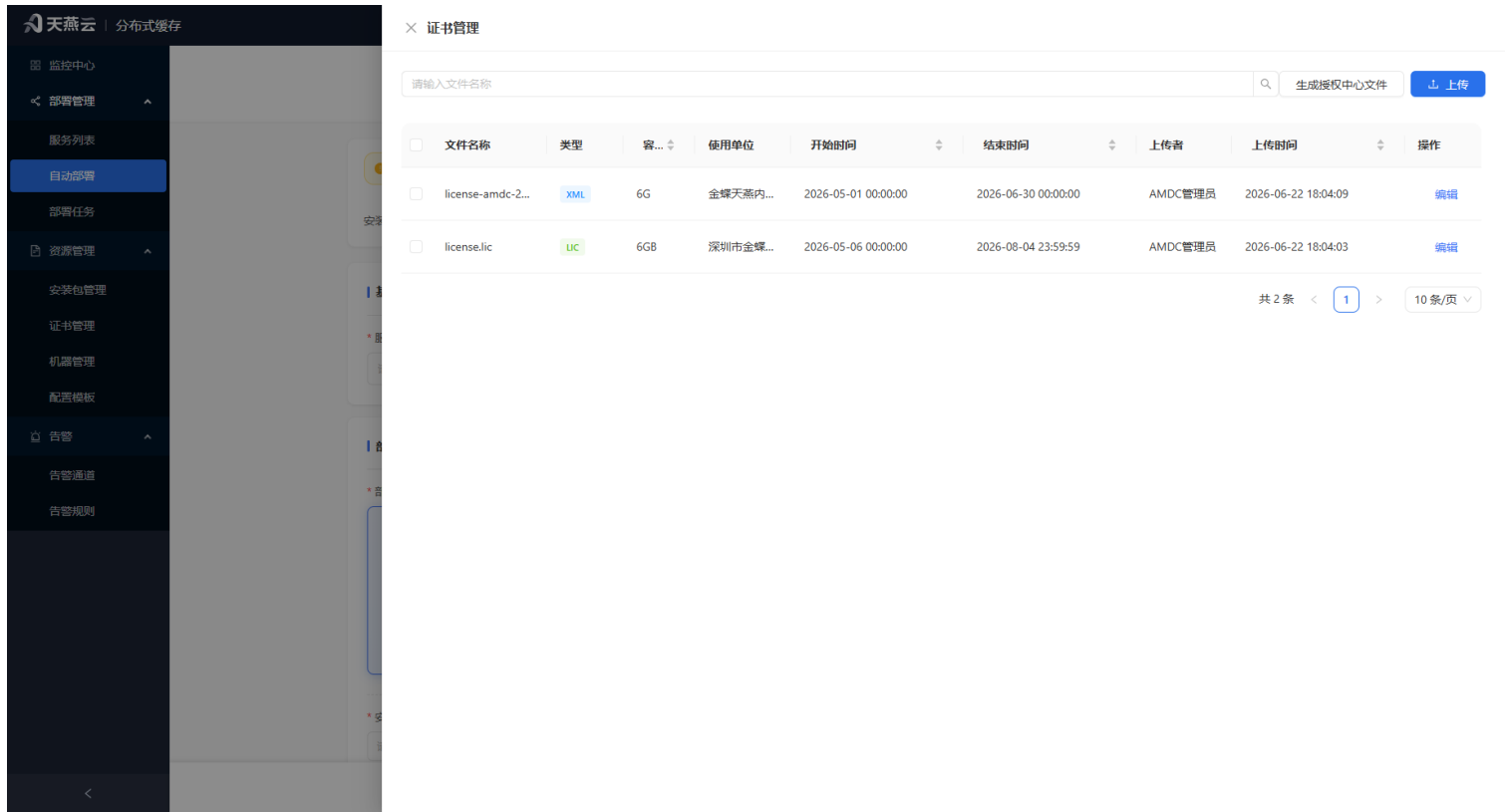
3.2.1 1.1 上传安装包

1. 进入【部署管理】>【自动部署】，在【基础配置】步骤点击顶部**安装包**旁的【查看】按钮。
2. 在弹出的"安装包管理"抽屉中，点击【上传】按钮，选择目标版本的 AMDC 安装包文件（如 `AMDC-V2.0.5-xxxx-amd64.tar.gz`）。
3. 点击【确定】完成上传。



3.2.2 1.2 上传 License

1. 在自动部署【基础配置】页面，点击顶部**证书**旁的【查看】按钮。
2. 在弹出的"证书管理"抽屉中，点击【上传】按钮，选择有效的 License 文件，可同时上传多个。
3. 上传后系统会自动解析，无效的 License 将无法保存。



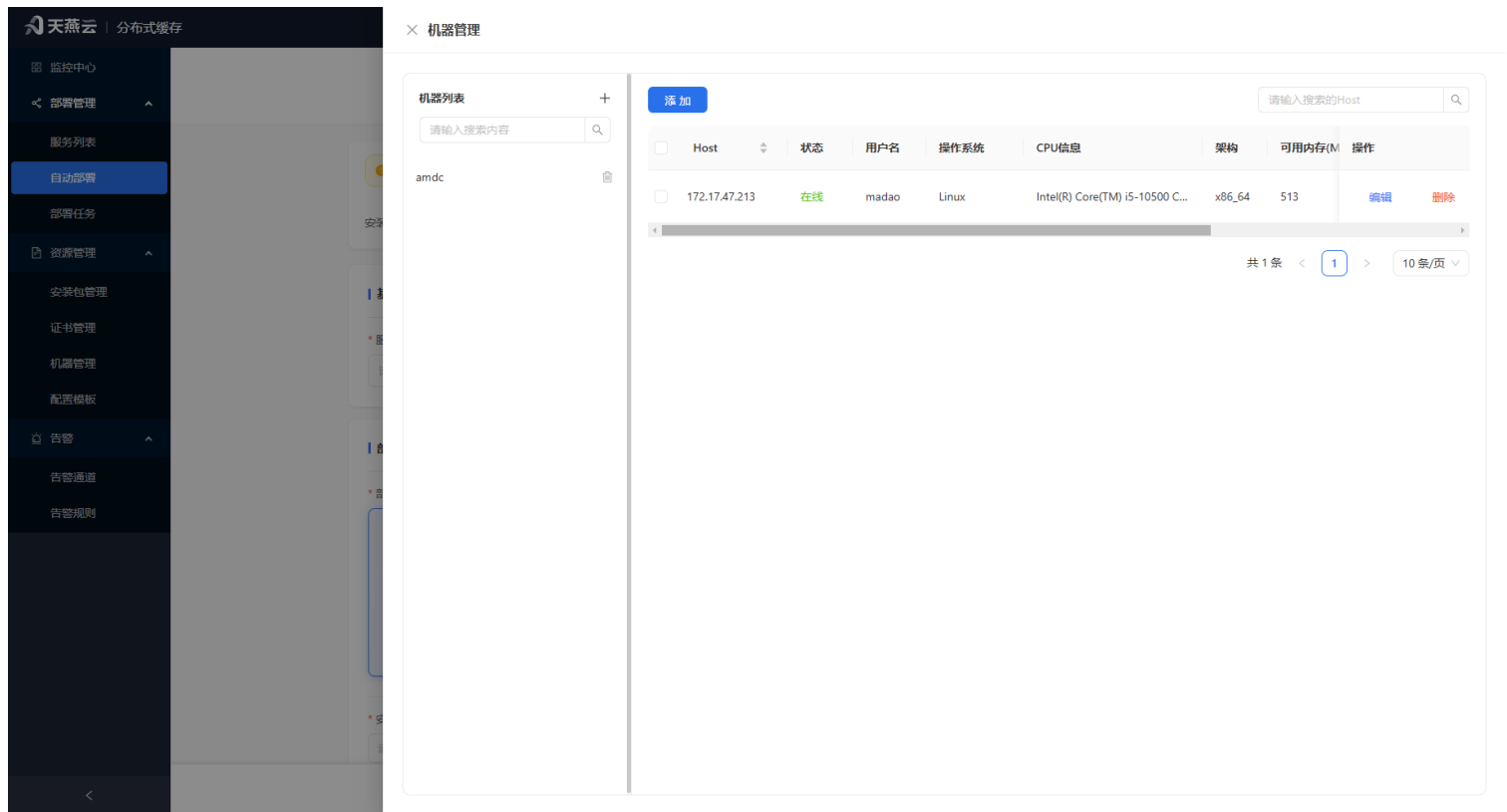
3.2.3 1.3 添加目标机器

1. 在自动部署【基础配置】页面，点击顶部**机器管理**旁的【查看】按钮。
2. 在弹出的"机器管理"抽屉中，先点击左侧机器组旁的【+】按钮创建机器组。
3. 选中机器组，点击左上方【添加】按钮，填写远程服务器信息：

参数名	示例值	说明
用户名	root	登录远程机器的用户名
密码	-	登录远程机器的密码
SSH 端口	22	远程 SSH 端口
Host	172.17.47.213	远程机器 IP
备注	-	可选

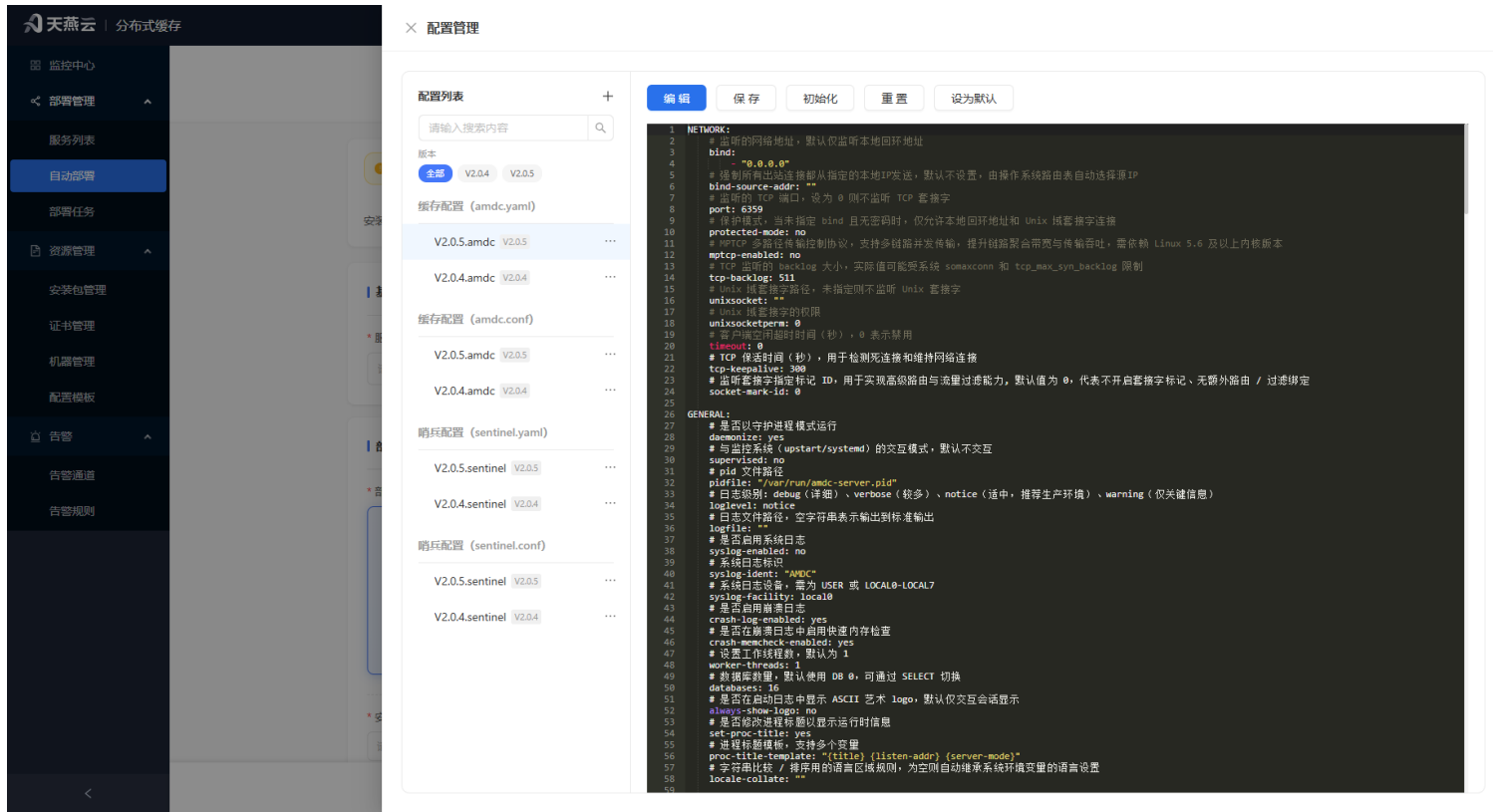
4. 确认目标机器连接状态正常后再进行后续部署。

注意：同一个机器组只能添加相同架构芯片的机器；自动部署需要目标机器具备 `tar`、`ss` 命令。



3.2.4 1.4 确认配置模板

1. 在自动部署【基础配置】页面，点击顶部**配置管理**旁的【查看】按钮。
2. 在弹出的"配置管理"抽屉中，确认已有缓存核心配置模板和哨兵配置模板（如使用哨兵模式）。
3. 若需要自定义，可在此修改；IP、端口等非公共参数不会生效。



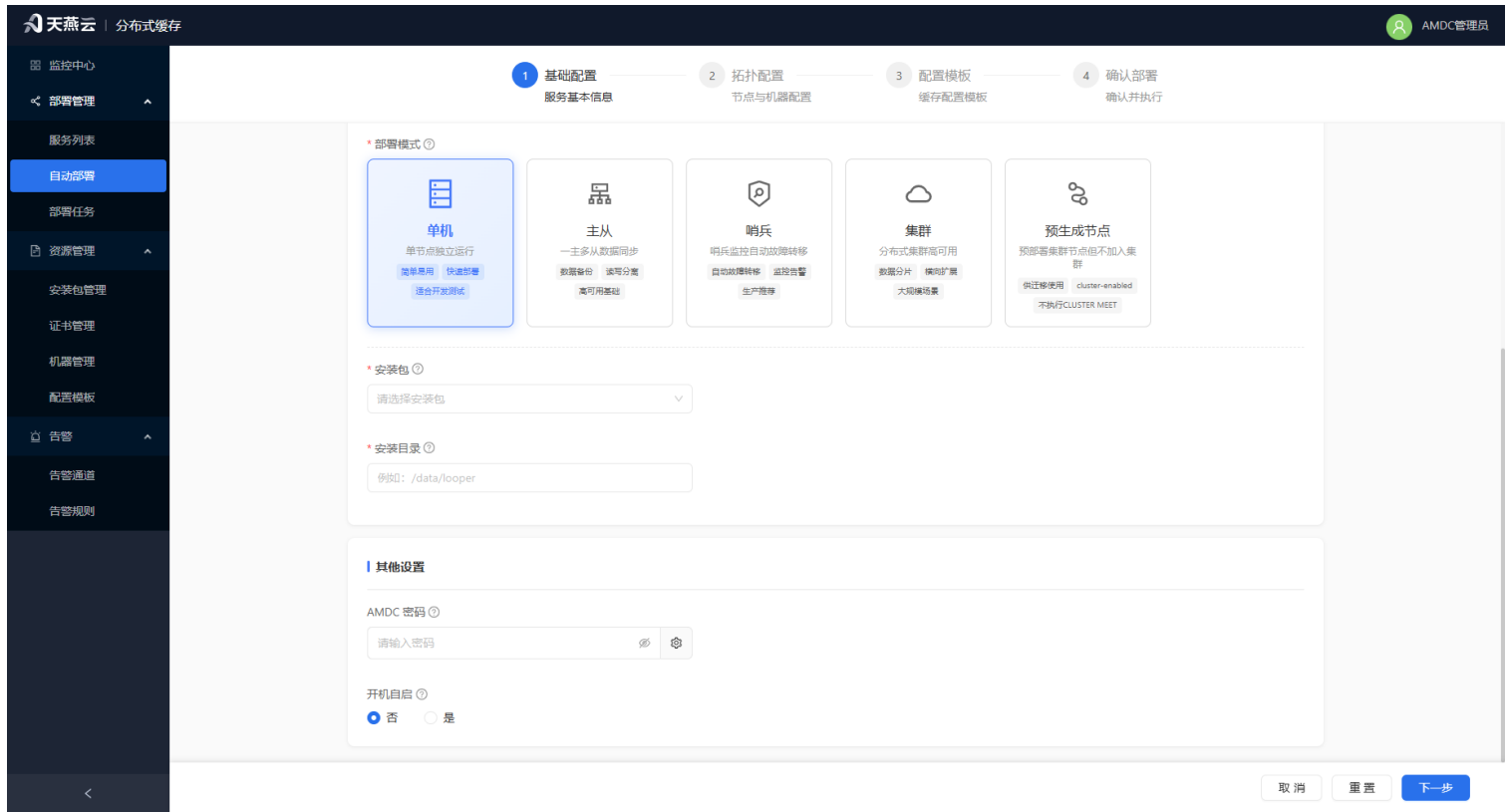
3.3 2. 自动部署新服务

资源准备完成后，继续在自动部署向导中按 4 步流程完成服务部署。

3.3.1 2.1 基础配置

在【基础配置】步骤填写：

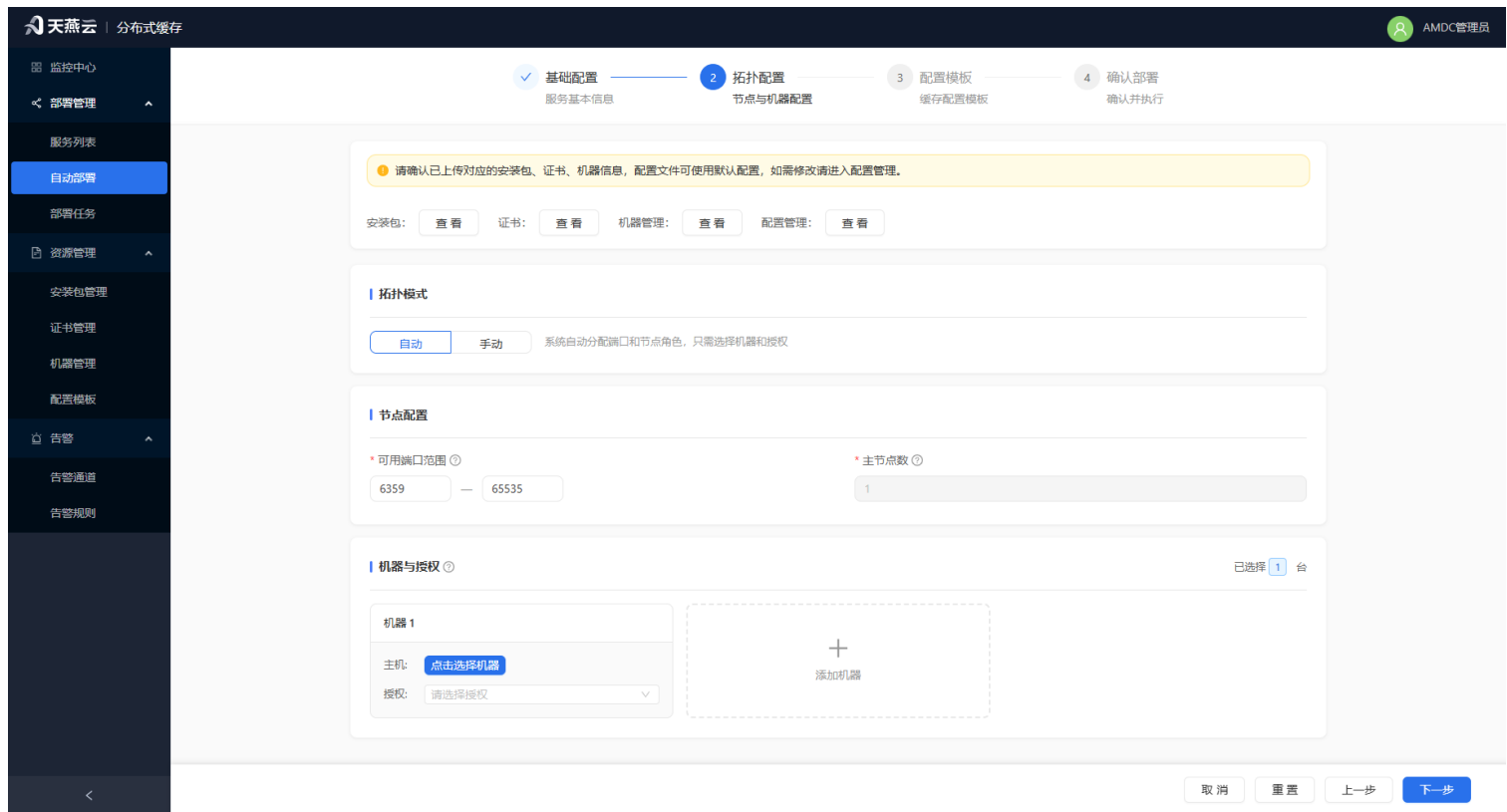
- **服务名称**：唯一标识，仅支持英文、数字、下划线、中划线，不超过 24 个字符。
- **部署模式**：选择单机 / 主从 / 哨兵 / 集群 / 预生成节点。
- **安装包**：选择已上传的安装包。
- **安装目录**：如 `/data/looper`，节点实际位于 `{seat}/{port}`。
- **AMDC 密码**：缓存核心管理密码，可点击设置按钮配置复杂度规则。
- **开机自启**：是否随系统启动自动运行。
- **sudo**：是否使用 sudo 权限执行安装命令。



3.3.2 2.2 拓扑配置

1. 选择【自动】或【手动】拓扑模式：
 - 自动：系统自动分配端口和节点角色，只需选择机器和授权。
 - 手动：逐节点配置机器、端口、角色和授权。
2. 配置端口范围（默认 6359–65535）和节点数量。
3. 在【机器与授权】区域选择目标机器，并为每台机器选择 License 授权。

注意：所有机器需在同一机器组下；非哨兵节点必须选择授权。

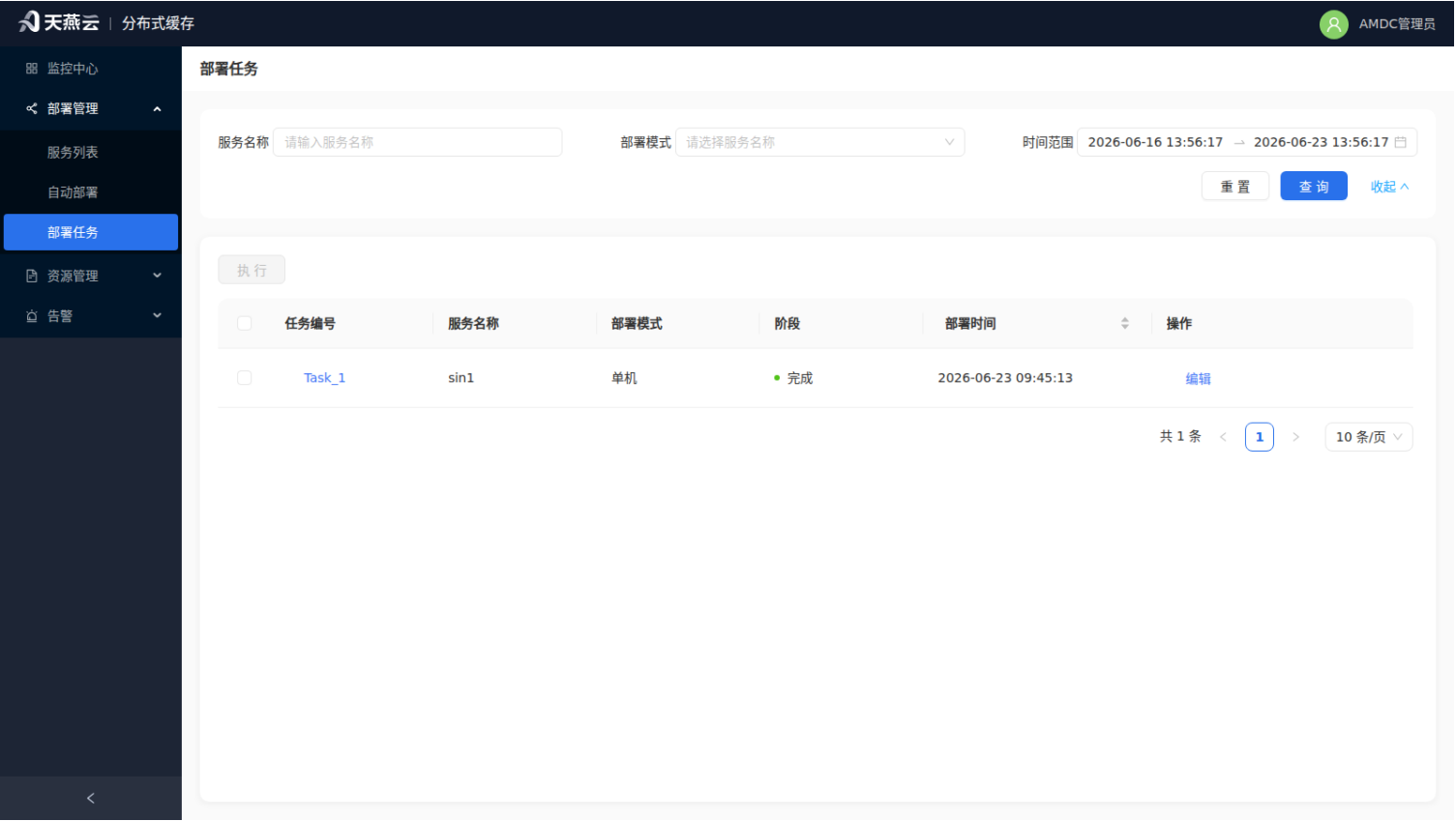


3.3.3 2.3 配置模板

选择缓存核心配置模板；若部署模式为哨兵，还需选择哨兵配置模板。

3.3.4 2.4 确认部署

1. 确认部署信息无误后，点击【执行任务】立即部署，或点击【保存任务】仅保存待后续执行。
2. 部署过程中可进入【部署管理】>【部署任务】查看任务状态。
3. 等待任务状态变为“完成”。



3.4 3. 导入已有实例

对于已通过其他方式部署的 AMDC 缓存核心，可快速导入控制台进行统一管理。

3.4.1 3.1 进入导入页面

进入【部署管理】>【服务列表】，点击左上角【导入】按钮，打开“服务导入”弹窗。

3.4.2 3.2 填写导入信息

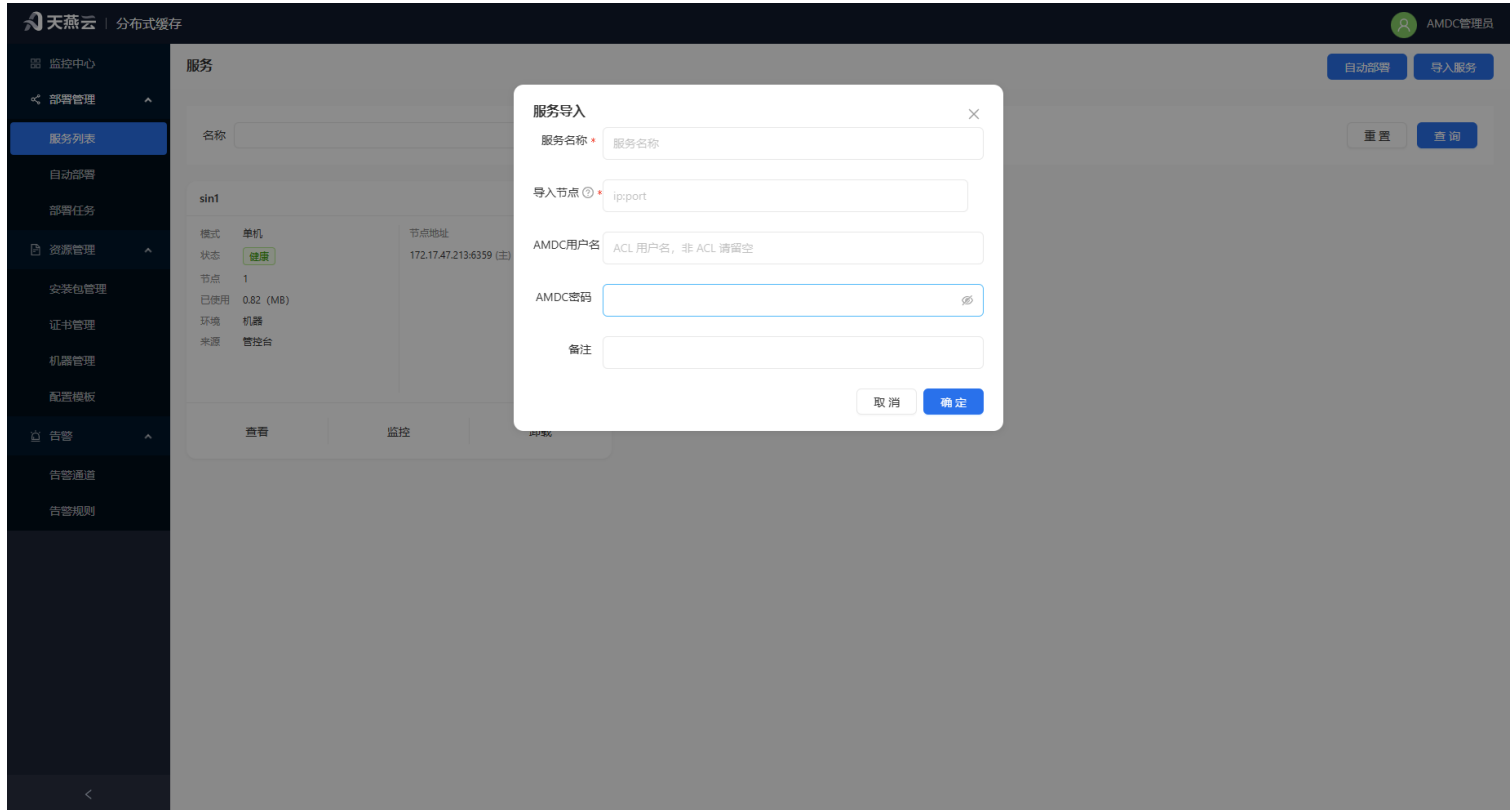
按实际实例模式填写以下信息：

参数名	示例值	说明
服务名称	imported_cluster	自定义服务名称
导入节点	172.17.47.213:6359	实例节点的 IP:Port，多个节点可点击【创建节点】添加
AMDC 用户名	-	ACL 用户名；未启用 ACL 请留空
AMDC 密码	-	缓存核心访问密码
备注	-	可选

注意：请按实例模式导入对应节点：

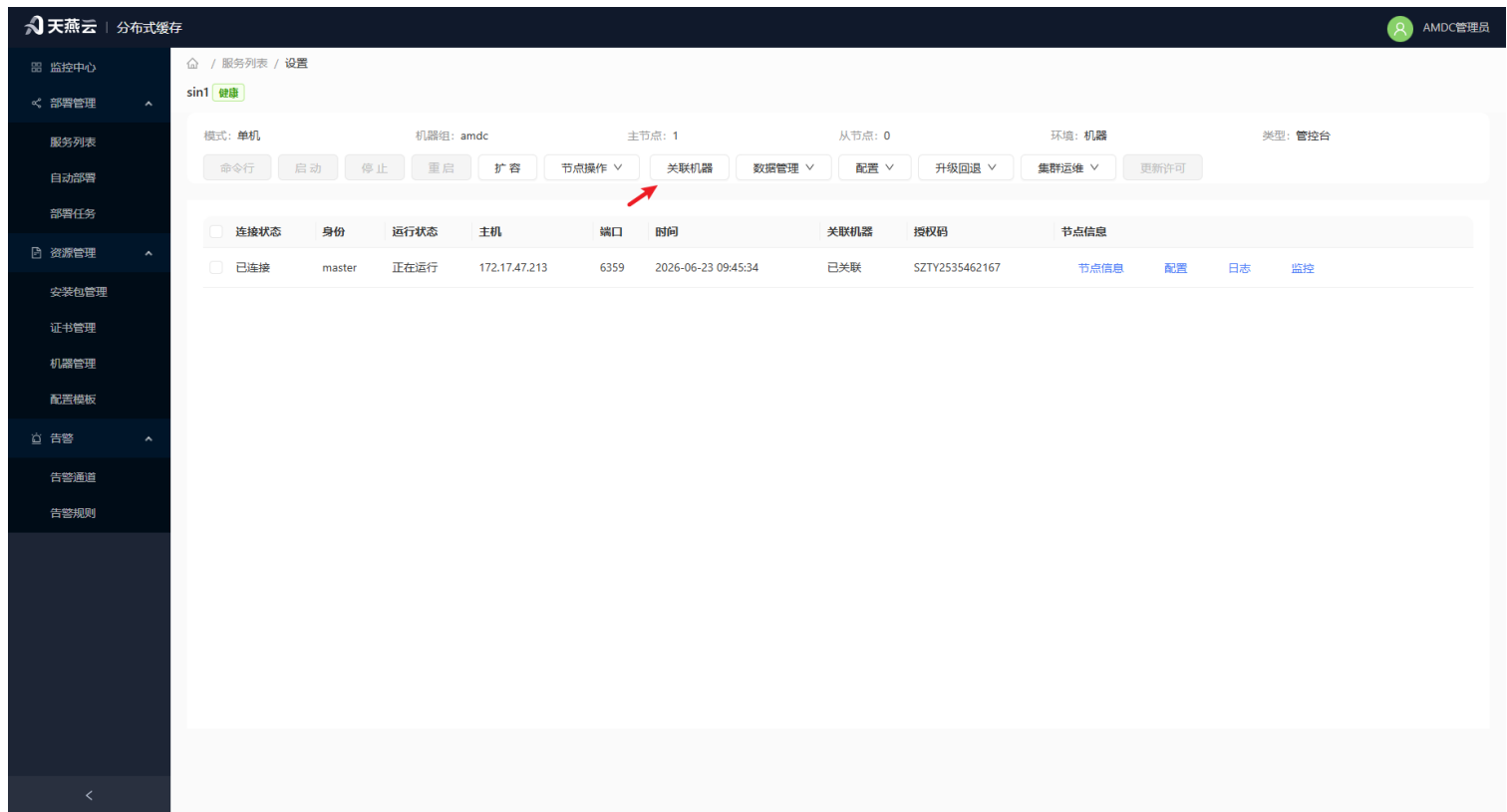
- **单机/主从模式**：导入主节点。
- **哨兵模式**：导入哨兵节点。
- **集群模式**：导入集群节点。

填写完成后点击【确定】，导入成功后服务列表会显示该实例。



3.4.3 3.3 关联机器

导入的节点默认未关联机器信息。为使用控制台提供的安装包升级、配置下发等完整管理功能，需进入服务详情 > 【设置】，选中节点后点击【关联机器】，将节点与机器管理中的目标机器绑定。



3.5 4. 服务验证

完成自动部署或导入后，进入【部署管理】>【服务列表】，点击服务名称进入服务详情，即可查看节点状态，或点击顶部【命令行】按钮执行 AMDC 命令验证数据写入/读取。

全国统一服务热线
4008-555-800



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



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

