



## 附录

金蝶Apusic中间件云平台v8.0.5

## 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 附录

- 1.1 操作命令
  - 1.1.1 PostgreSQL 操作命令
  - 1.1.2 MiniIO 操作命令
- 1.2 Apusiclet 介绍
  - 1.2.1 主要功能
  - 1.2.2 工作流程
    - 1.2.2.1 操作说明
    - 1.2.2.2 apusiclet 制品包的制作
    - 1.2.2.3 关于 apusiclet 升级
    - 1.2.2.4 手动安装 apusiclet
    - 1.2.2.5 apusiclet 制品库管理

# 1 附录

## 1.1 操作命令

### 1.1.1 PostgreSQL 操作命令

```
# 启动
./bin/maas.sh middleware start postgresql

# 停止
./bin/maas.sh middleware stop postgresql

# 查看状态
./bin/maas.sh middleware status postgresql

# 查看日志
./bin/maas.sh middleware logs postgresql
```

### 1.1.2 MinIO 操作命令

```
# 启动
./bin/maas.sh middleware start minio

# 停止
./bin/maas.sh middleware stop minio

# 查看状态
./bin/maas.sh middleware status minio

# 查看日志
./bin/maas.sh middleware logs minio
```

## 1.2 Apusiclet 介绍

### 1.2.1 主要功能

1. 对进程操作和管理
  - apusiclet 通过对进程的纳管，确保在节点上对进程进行启动、停止、重启等操作。
2. 对 peasecod 资源的管理
  - 定时同步与 ACP 平台进程的状态，虚拟机状态。
3. 健康检查 (Liveness/Readiness) :
  - apusiclet 会定期检查容器的健康状况，执行 liveness (存活) 和 readiness (就绪) 探针。如果容器健康检查失败，apusiclet 会重新启动容器。
4. 日志收集
  - apusiclet 会处理进程日志的收集。
5. 资源监控
  - apusiclet 会定期报告节点资源使用情况 (如 CPU、内存、磁盘等) 给 ACP 控制平面，以便集群管理员可以监控节点健康和负载。
6. 与 ACP 服务器通信:
  - apusiclet 定期向 ACP API 服务器报告节点和进程的状态，并接收新的调度指令。它负责与 API 服务器的实时通信，确保进程与集群的状态一致。

### 1.2.2 工作流程

#### 1.2.2.1 操作说明

1. 获取当前虚拟机中所有的进程对象。

名词解释：

1. pc (进程组) : 由一个或者多个进程组成的集合。
2. NAMESPACE: ACP 中租户 ID。
3. CONTEXT: 当前实例集群的名称。
4. READY: 进程组正常能访问的个数 (正常进程个数/进程组总数) 。
5. RESTART: 重启次数。当被纳管的进程关闭或自动退出时，`apusiclet` 会自动拉起该进程，重启次数依次递增，最大重启时间为 5min。
6. STATUS: 当前进程组的状态，可能值包括 INITIALIZED (初始化)、CREATED (已创建)、RUNNING (运行中)、CHECKING (状态校验)、FAILED (已失败)、UPGRADE (升级中)、EXISTED (进程退出)、SUCCESS (成功)、STOPPED (已停止) 。

```
$ apusiclet pc ls
```

| ID  | NAME                     | CONTEXT                     | NAMESPACE |
|-----|--------------------------|-----------------------------|-----------|
| 0   | demo-exporter-ayynukaq-0 | 示例集群                        | 1         |
| 1/1 | 0                        | SUCCESS 2024-11-25 15:25:09 |           |
| 1   | demo-app-omgyciiw-0      | 示例集群                        | 1         |
| 1/1 | 0                        | SUCCESS 2024-11-25 15:25:03 |           |

2. 获取当前进程组资源状态及对象 YAML，使用上下方向键选择目标进程组。

```
$ apusiclet pc get
```

Use the arrow keys to navigate: ↓ ↑ → ← and / toggles search  
select peasecod Output format. One of: yaml

```
🐱 demo-exporter-ayynukaq-0 示例集群
```

```
1
```

```
demo-app-omgyciiw-0 1
```

```
----- Info -----
```

```
Name: demo-exporter-ayynukaq-0
```

```
Context: 示例集群
```

3. 获取当前实例的进程日志。

步骤与 `apusiclet pc get` 类似，使用上下方向键选择目标进程组后即可查看对应日志。

```
$ apusiclet logs
```

Use the arrow keys to navigate: ↓ ↑ → ← and / toggles search  
select peasecod Output format. One of: yaml

```
🐱 demo-exporter-ayynukaq-0 示例集群
```

```
1
```

demo-app-omgyciw-0

1

----- Info -----

Name: demo-exporter-ayynukaq-0

Context: 示例集群



Select:demo-exporter-ayynukaq-0

server address is 127.0.0.1:2185

exporter\_port is 9145

2024/11/25 15:25:09 info: zookeeper hosts: [127.0.0.1:2185]

2024/11/25 15:25:09 info: serving metrics at :9145/metrics

server address is 127.0.0.1:2185

exporter\_port is 9145

2024/11/25 15:25:09 info: zookeeper hosts: [127.0.0.1:2185]

2024/11/28 14:02:33 info: serving metrics at :9145/metrics

2024/12/02 14:53:50 info: zookeeper hosts: [127.0.0.1:2185]

2024/12/02 14:53:50 info: serving metrics at :9145/metrics

2024/12/02 14:53:50 fatal: shutting down exporter: listen tcp :9145: bind: address already in use

#### 4. 查看当前 volumes 挂载的卷

\$ apusiclet volume ls

| ID | NAME | PATH | MOUNT | SIZE | STATUS | TIME |
|----|------|------|-------|------|--------|------|
|----|------|------|-------|------|--------|------|

### 1.2.2.2 apusiclet 制品包的制作

方式一:

#### 1. 构建 apusiclet 与 apusic 制品包

\$ apusiclet preset build &lt;REPOSITORY\_NAME&gt;:&lt;VERSION&gt;

✓ Name (软件包名称): minio

```

✓ Version (软件包版本): 20210422
✓ Maintainer (作者): apusic
Preset (软件包): minio-20210422-linux-amd64.tar.gz
WorkDir (工作路径): /root/minio-test
Volumes (Bind mount a volume [hostPath:containerPath],
[hostPath1:containerPath1]): ./data:./data
Environment (环境变量): MINIO_ROOT_USER=
[YOUR_USERNAME],MINIO_ROOT_PASSWORD=[YOUR_PASSWORD]
Command (执行命令(输入 'exit' 退出))
请输入command:: ./minio server ./data/minio
请输入command:: exit

```

⚠ 安全提示: 生产环境部署必须修改为强密码, 避免使用默认凭证。

方式二:

1. 初始化 `metadata.json`, 初始化成功后会在当前目录生成一个 `metadata.json` 文件。

```
apusiclet preset init
```

2. 修改当前 `metadata.json`

```

{
  "metadata": {},
  "raw": {
    "name": "minio",
    "version": "20210422",
    "maintainer": "apusic"
  },
  "images": {
    "preset": "minio-20210422-linux-amd64.tar.gz"
  },
  "os": "linux",
  "command": [
    "./minio server /root/minio-dev/data"
  ]
}

```

```
],
  "create_time": "0001-01-01T00:00:00Z"
}
```

### 3. 构建 preset 中间件安装包

```
apusiclet preset build minio:v1 -f metadata.json
=> naming to minio:v1
```

#### 1.2.2.3 关于 apusiclet 升级

```
# 查看当前虚拟机中的 apusiclet 当前的版本
apusiclet version

APUSICLET
apusic 中间件生命周期管理
Version:      8.0.5.23
GitRevision:  075de3cd90e7d279657267cfb1d47a7093cc877a
GolangVersion: go1.23.4
BuildStatus:  develop
GitTag:       v8.0.5.23
Platform:     linux/amd64
BuildDate:    2025-06-17-14:31:53

# 如果需要更新到最新版本, 可以通过页面上更新 apusiclet, 也可以通过命令行更新

apusiclet upgrade
```

#### 1.2.2.4 手动安装 apusiclet

```
# 手动卸载 apusiclet 软件包
systemctl stop apusiclet
yum remove -y apusiclet
```

```
# 下载并安装 apusiclet
wget https://<internal-nexus-url>/repository/file-
server/acp/apusiclet/apusiclet-8.0.5.23-Linux-amd64.rpm
rpm -ivh apusiclet-8.0.5.23-Linux-amd64.rpm
rm apusiclet-8.0.5.23-Linux-amd64.rpm -rf

systemctl daemon-reload
systemctl start apusiclet
```

如果更新过程中出现问题，可使用以下命令查看日志

```
$ cat /var/log/messages | grep upgrade.sh
```

### 1.2.2.5 apusiclet 制品库管理

#### 1. 查询制品包列表

```
$ apusiclet preset ls
```

| REPOSITORY                 | TAG    | ID         | CREATED |
|----------------------------|--------|------------|---------|
| SIZE                       |        |            |         |
| 172.24.3.XX:9010/acp/minio | latest | c4d70f148d | 2025-   |
| 03-11 15:50:11 17 MB       |        |            |         |

#### 2. 删除本地制品包

```
$ apusiclet preset rm 172.24.3.XX:9010/acp/minio:latest
remove preset down ...
```

#### 3. 拉取仓库的制品包

```
$ apusiclet pull 172.24.3.XX:9010/acp/minio:latest
Connecting to 172.24.3.XX:9010 ... connected.
Length: 16690390 (17 MB) [application/x-rpm]
```

```
minio-latest.tar.gz 100% [=====] (616
MB/s)
2025-03-11 18:15:16 - 'minio-latest.tar.gz' saved
[16690390:16690390]
```

#### 4. 推送仓库的制品包

```
$ apusiclet push 172.24.3.XX:9010/acp/minio:latest
Length: 16690388 (17 MB) [application/octet-
stream]
minio-latest.tar.gz 100% [=====] (204
MB/s)
2025-03-11 18:15:59 - 'minio-latest.tar.gz' pushed...
[16690388:16690388]
```

#### 5. 推送制品包和拉取制品包时，需要登录对应的仓库

```
$ apusiclet login 172.24.3.XX:9010
Username:: [YOUR_USERNAME]
✓ Password:: [YOUR_PASSWORD]
Login Succeeded
```

 **安全提示：**生产环境部署必须修改为强密码，避免使用默认凭证。

#### 6. 运行指定的制品包服务

```
$ apusiclet preset run --name minio-demo
172.24.3.XX:9010/acp/minio:latest
peasecod minio-demo created

# 查询当前服务运行状态
$ apusiclet pc ls
```

| ID | NAME | CONTEXT | NAMESPACE |
|----|------|---------|-----------|
|----|------|---------|-----------|

```

READY    RESTART STATUS    TIME
0        minio-demo          minio-demo          default
1/1      0              SUCCESS 2025-03-11 18:20:38

```

# 获取当前服务的日志

```
# apusiclet pc logs <NAME> <NAMESPACE>
```

```
$ apusiclet pc logs minio-demo default
```

```

Endpoint: http://172.24.3.XX:9000  http://192.168.x.x:9000
http://192.168.x.x:9000  http://192.168.x.x:9000
http://192.168.x.x:9000  http://192.168.x.x:9000
http://192.168.x.x:9000  http://192.168.x.x:9000
http://192.18.0.x:9000  http://127.0.0.1:9000

```

Browser Access:

```

http://172.24.3.XX:9000  http://192.168.x.x:9000
http://192.168.x.x:9000  http://192.168.x.x:9000
http://192.168.x.x:9000  http://192.168.x.x:9000
http://192.168.x.x:9000  http://192.168.x.x:9000
http://192.18.0.x:9000  http://127.0.0.1:9000

```

Object API (Amazon S3 compatible):

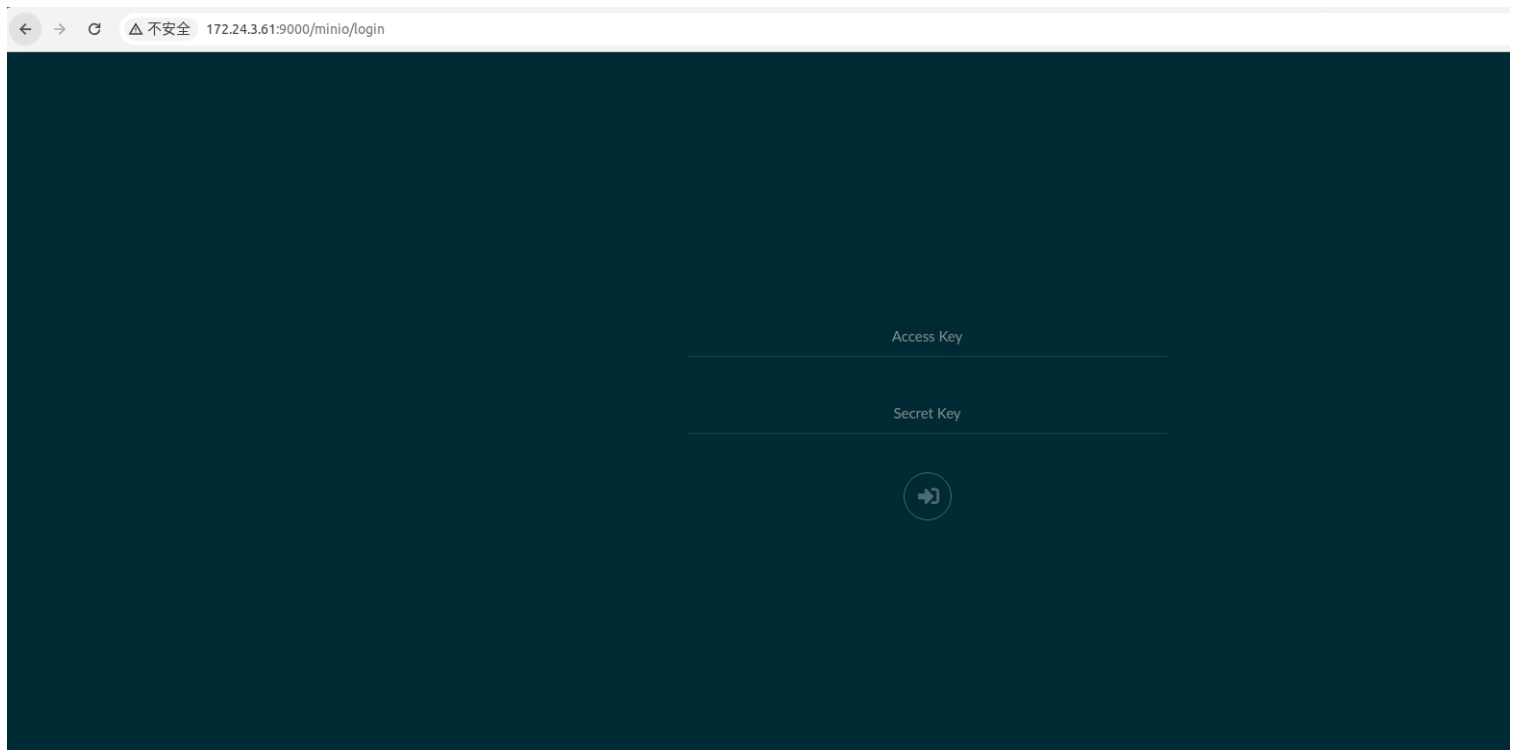
```

Go:          https://docs.min.io/docs/golang-client-quickstart-
guide
Java:        https://docs.min.io/docs/java-client-quickstart-
guide
Python:      https://docs.min.io/docs/python-client-quickstart-
guide
JavaScript:  https://docs.min.io/docs/javascript-client-
quickstart-guide
.NET:        https://docs.min.io/docs/dotnet-client-quickstart-
guide

```

```
Detected default credentials 'minioadmin:minioadmin', please  
change the credentials immediately using 'MINIO_ROOT_USER' and  
'MINIO_ROOT_PASSWORD'  
IAM initialization complete
```

访问当前 MinIO 是否正常运行：



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](http://www.apusic.com)

