

# PDMAasEE 升级手册（2.3.2）

注意 1：为了防止覆盖自行修改过模板的内容，本次升级将数据库模板相关的更新单独提取至 `template-2.3.1-2.3.2.sql`，升级时可自行选择是否执行该脚本

## 1. Docker 安装版升级方法

### 1.1. 服务停止

将正在运行的前后端镜像停止：

```
docker stop pdmaas-service
docker stop pdmaas-web
```

### 1.2. 容器删除

注：以当前运行的镜像版本号为 2.3.1 为例，如不是 2.3.1 本，需要将下方命令的 2.3.1 替换成当前的版本

```
docker rm pdmaas-service
docker rmi pdmaas-service:2.3.1
docker rm pdmaas-web
docker rmi pdmaas-web:2.3.1
```

### 1.3. 更新镜像

#### 1.3.1. 下载地址

##### 1.3.1.1. ARM

1. pdmaas-service:

<https://pdmaas-ent-download.oss-cn-hangzhou.aliyuncs.com/docker/pdmaas-service.tar>

2. pdmaas-web.tar:

<https://pdmaas-ent-download.oss-cn-hangzhou.aliyuncs.com/docker/pdmaas-web.tar>

##### 1.3.1.2. AMD

1. pdmaas-service:

<https://pdmaas-ent-download.oss-cn-hangzhou.aliyuncs.com/dockeramd/pdmaas-service.tar>

## 2. pdmaas-web.tar:

<https://pdmaas-ent-download.oss-cn-hangzhou.aliyuncs.com/dockeramd/pdmaas-web.tar>

### 1.3.2.2. 导入离线包

```
docker load -i pdmaas-service.tar
docker load -i pdmaas-web.tar
```

## 1.4. 下载升级包

升级包: <https://pdmaas-ent-download.oss-cn-hangzhou.aliyuncs.com/upgradeDocker.zip>

### 1.4.1. 执行增量脚本

根据当前使用的版本和需要升级的版本按顺序执行升级包内 increment.sql 增量脚本，例如如果当前是 2.0.0，需要升级到 2.0.1，则需要执行 pdmaas\_ee\_boot-2.0.0-2.0.1-increment.sql，如果当前是 2.0.0，需要升级到 2.0.2，则还需要执行 pdmaas\_ee\_boot-2.0.1-2.0.2-increment.sql 脚本，以此类推按顺序执行即可。

## 1.5. 启动容器

将 docker-compose.yml 文件内的镜像名后的版本号修改成最新的 2.3.2

```
version: '3'
services:
  pdmaas:
    image: pdmaas-service:2.3.2 ←
    container name: pdmaas-service #不建议更改
    networks:
      - pdmaas-network #pdmaas容器间通信方式
    env file:
      - pdmaas.env #pdmaas-service环境变量配置信息
    ports:
      - "9080:9080" #端口映射，如果觉得暴露端口存在安全隐患，可注释端口映射
    volumes:
      - D:/dockerconfig/test/service/logs:/opt/pdmaas/logs/ #日志文件映射，请更改冒号前的路径为宿主机日志地址

  pdmaas-web:
    image: pdmaas-web:2.3.2 ←
    container name: pdmaas-web #不建议更改
    networks:
      - pdmaas-network #pdmaas容器间通信方式
    ports:
      - "80:80" #页面访问端口，修改冒号前的端口为浏览器访问地址对应端口号
    volumes:
      - D:/dockerconfig/test/web:/var/log/nginx/ #日志文件映射，请更改冒号前的路径为宿主机日志地址
```

执行 `docker-compose up -d`

## 2. 独立安装版升级方法

### 2.1. 下载升级包

升级包: <https://pdmaas-ent-download.oss-cn-hangzhou.aliyuncs.com/upgrade.zip>

### 2.2. 执行增量脚本

根据当前使用的版本和需要升级的版本按顺序执行升级包内 `increment.sql` 增量脚本, 例如如果当前是 2.0.0, 需要升级到 2.0.1, 则需要执行 `pdmaas_ee_boot-2.0.0-2.0.1-increment.sql`, 如果当前是 2.0.0, 需要升级到 2.0.2, 则还需要执行 `pdmaas_ee_boot-2.0.1-2.0.2-increment.sql` 脚本, 以此类推按顺序执行即可。

### 2.3. 替换前后端

#### 2.3.1. 替换前端

```
1 upstream pdmaas-app {
2   #ip_hash;
3   server 127.0.0.1:9080;
4 }
5
6
7 upstream pdmaas-ws {
8   #ip_hash;
9   server 0.0.0.0:9000;
10 }
11
12
13 server {
14   listen 80;
15   server_name localhost;
16   ssl_prefer_server_ciphers on;
17   location / {
18     if ($request_filename ~* .*\.?(?:htm|html)$)
19     {
20       add_header Cache-Control "private, no-store, no-cache, must-revalidate, proxy-revalidate";
21     }
22     root /opt/PDMAas/pdmaas-ee-web;
23     try_files $uri $uri/ /index.html;
24     index index.html index.htm;
25   }
26   location /webscoket {
27     proxy_pass http://pdmaas-ws;
28     proxy_http_version 1.1;
29     proxy_set_header Upgrade $http_upgrade;
30     proxy_set_header Connection "upgrade";
31     proxy_set_header Host $host;
32   }
33
34   location /pdmaas-ee {
35     proxy_pass http://pdmaas-app/pdmaas-ee;
36   }
37   error_page 500 502 503 504 /50x.html;
38   location = /50x.html {
39     root html;
40   }
41 }
42 }
43 }
44 }
45 }
46 }
47 }
```

根据当前正在运行的前端服务路径将升级包内的 `pdmaas-ee-web` 替换, 然后重启 `nginx` 即可

### 2.3.2. 修改 Nginx 配置【如在上次升级中修改过此处，则直接忽略】

1. 修改 PDMaas\_V2.conf 文件，新增下方两行配置（如下图）【如该属性已经存在，请忽略】

```
proxy_buffering off;  
proxy_cache off;
```

```
57  
58  
59     location /pdmaas-ee {  
60         proxy_buffering off;  
61         proxy_cache off;  
62         proxy_pass http://pdmaas-app/pdmaas-ee;  
63     "PDMaas_V2.conf" 49L, 1014C
```

2. 修改 PDMaas\_V2.conf 文件，新增下方一行配置（如下图）【如该属性已经存在，请忽略】

```
client_max_body_size 200M;
```

```
client_max_body_size 200M;  
upstream pdmaas-app {  
    #ip_hash;  
    server 127.0.0.1:9080;  
}  
  
upstream pdmaas-ws {  
    #ip_hash;
```

### 2.3.3. 修改后端配置文件【如该属性已经存在，请忽略】

```
Spring:
6   neo4j:
7     password: Y123456!
8
9   data:
10    redis:
11     port: 6379
12     host: 192.168.3.206
13     database: 0
14     lettuce:
15      pool:
16       max-active: 8 # 最大活跃连接数
17       max-idle: 8 # 最大空闲连接数
18       min-idle: 0 # 最小空闲连接数
19       max-wait: 1000ms # 获取连接的最大等待时间
20
21     jedis:
22      pool:
23       max-active: 8 # 连接池最大连接数（使用负值表示没有限制）
24       max-idle: 8 # 连接池中的最大空闲连接
25       min-idle: 0 # 连接池中的最小空闲连接
26       max-wait: -1ms # 连接池最大阻塞等待时间（使用负值表示没有限制）
27
28     connect-timeout: 20000
29     client-type: lettuce
30
31   datasource: # 数据源配置
32    type: com.zayxon.hikari.HikariDataSource
```

将 application-public.yml 文件按照红框内的内容修改

## 2.3.4. 替换后端

```
pdmaas-restart.sh
1 #!/bin/bash
2
3 # 获取脚本所在目录
4 BASE_DIR=$(dirname $(realpath $0))
5
6 # 定义变量
7 LOG_DIR="$BASE_DIR/../log"
8 PID_FILE="$LOG_DIR/pdmaas.pid"
9 JAR_FILE="$BASE_DIR/../pdmaas-ee-boot.jar"
10 CONFIG_FILE="$BASE_DIR/../conf/application-public.yml"
11
12 # 检查 PID 文件是否存在
13 if [ -f "$PID_FILE" ]; then
14     # 读取 PID
15     PID=$(cat $PID_FILE)
16     echo "Stopping application with PID $PID"
17
18     # 停止进程
19     kill $PID
20
21     # 检查进程是否已停止
22     if [ $? -eq 0 ]; then
23         echo "Application stopped successfully"
24         # 删除 PID 文件
25         rm -f $PID_FILE
26
27         # 启动应用程序，日志追加到现有文件
28         java -jar $JAR_FILE --spring.config.additional-location=$CONFIG_FILE >> $LOG_DIR/pdmaas.log 2>&1 &
29
30         # 获取应用程序的 PID
31         NEW_PID=$!
32         echo "Application started with PID $NEW_PID. Logs are being written to $LOG_DIR/pdmaas.log"
33
34         # 将 PID 写入文件
35         echo $NEW_PID > $PID_FILE
36
37         # 安排日志清理
38         echo "Log cleanup scheduled. Logs older than one month will be deleted."
39         find $LOG_DIR -type f -name "*.log" -mtime +30 -exec rm -f {} \;
40     else
41         echo "Failed to stop the application"
42     fi
43 else
44     echo "PID file not found. Is the application running?"
45 fi
46
```

根据当前正在运行的后端服务路径将升级包内的 `pdmaas-ee-boot.jar` 替换，然后 `pdmaas-restart.sh` 重启后端服务即可

## 2.3.5. 内置驱动（可选）【如在上次升级中修改过此处，则直接忽略】

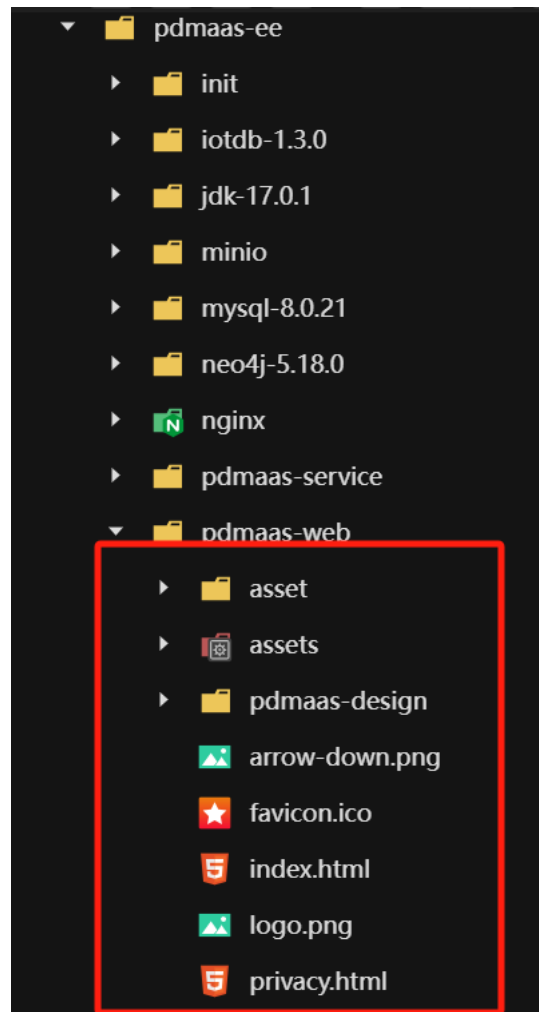
升级包内的 `jdbc` 目录包含了目前系统支持解析的所有 `jdbc` 驱动，可以选择将该目录拷贝至 `pdmaas-restart.sh` 脚本所在的 `jdbc` 目录内，也可以自行在系统中上传

# 3. 一键安装版升级方法

## 3.1. 下载升级包

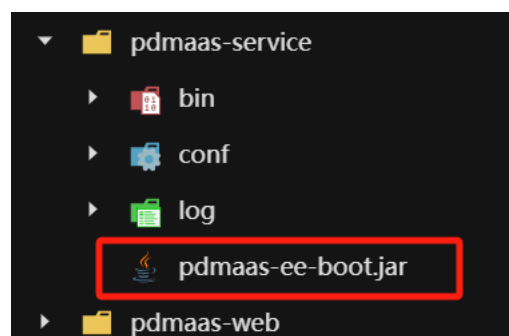
升级包: <https://pdmaas-ent-download.oss-cn-hangzhou.aliyuncs.com/upgrade.zip>

### 3.2. 替换前端



把该目录下的数据用升级包内的内容全部替换。然后执行以下命令重启前端服务：`systemctl restart pdmaas-web`

### 3.3. 替换后端



将 jar 替换即可。然后执行以下命令重启后端服务：`systemctl restart pdmaas-service`

### 3.4. 执行增量脚本

可在终端执行以下命令进入到 mysql: `/opt/pdmaas-ee/mysql-8.0.21/bin/mysql -uroot -p --socket=/opt/pdmaas-ee/mysql-8.0.21/mysql.sock`

根据当前使用的版本和需要升级的版本按顺序执行升级包内 `increment.sql` 增量脚本，例如如果当前是 2.0.0，需要升级到 2.0.1，则需要执行 `pdmaas_ee_boot-2.0.0-2.0.1-increment.sql`，如果当前是 2.0.0，需要升级到 2.0.2，则还需要执行 `pdmaas_ee_boot-2.0.1-2.0.2-increment.sql` 脚本，以此类推按顺序执行即可。