

- 5GC OMC安装指南
 - 1 安装环境及依赖包
 - 1.1 安装数据库
 - 1.2 安装WEB服务(可选)
 - 2 安装配置OMC
 - 2.1 安装OMC软件包
 - 2.2 修改nginx服务的OMC配置(可选)
 - 2.3 修改OMC Web前端配置
 - 2.4 修改OMC后端配置
 - 2.5 数据库配置
 - 2.6 防火墙配置
 - 2.7 配置被管网元的SSH信任关系
 - 3 启动和停止OMC
 - 3.1 系统服务启动
 - 3.2 手动启动/查询/停止OMC
 - 4 问题定位跟踪
 - 5 管理核心网系统

5GC OMC安装指南

1 安装环境及依赖包

X86_64通用服务器:

计算机或者虚拟机环境(最低配置): CPU: 2, Memory: 8.00 GB, Disk: 200 GB
操作系统环境: Ubuntu 18.04
数据库环境: mariadb-10.3.38-GA
WEB服务: nginx 1.14.0 (ubuntu)
安装包名称格式: OMC 5GC16.1.1.amd64.deb (其中, '5GC16.1.1'为版本号, 'amd64'为硬件平台名称)

麒麟arm64服务器:

计算机或者虚拟机环境(最低配置): CPU: 2, Memory: 8.00 GB, Disk: 200 GB
操作系统环境: 银河麒麟 kylin server v10
数据库环境: mariadb-10.3.35-1.p01.ky10.aarch64
WEB服务: nginx 1.16.1-11.p01.ky10.aarch64

安装包名称格式：OMC-5GC16.1.1.arm64.rpm（其中，'5GC16.1.1'为版本号，'arm64'为硬件平台名称）

注意事项：

- * 以下操作以银河麒麟arm64硬件平台为例
- * 全新安装方式，数据库的数据会被全部清除

1.1 安装数据库

注意：非第一次安装或者全新安装请跳过此步骤

mariadb版本：mariadb-server-3:10.3.35-1.p01.ky10.aarch64

软件安装

银河麒麟OS(CentOS)

```
# yum clean all
# yum makecache
# yum -y update
# yum install -y mariadb
# yum install -y mariadb-server-3:10.3.35-1.p01.ky10.aarch64
```

配置修改

修改/etc/my.cnf.d/mariadb-server.cnf 增加或者修改如下三行：

```
bind-address=0.0.0.0
port=33066
max_connections=1000
```

验证测试

```
# systemctl enable mariadb
# systemctl restart mariadb
# systemctl status mariadb
```

正常显示结果如下：

```
[root@ems omc]# systemctl status mariadb
● mariadb.service - MariaDB 10.3.35 database server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; disabled; vendor preset: disabled)
   Active: active (running) since Sun 2023-04-30 11:09:56 CST; 7h ago
     Docs: man:mysql(8)
           https://mariadb.com/kb/en/library/systemd/
 Main PID: 5127 (mysqld)
    Status: "Taking your SQL requests now..."
     Tasks: 37
  Memory: 101.0M
    CGroup: /system.slice/mariadb.service
            └─5127 /usr/libexec/mysqld
```

修改mysql数据库root密码:

!!!注意!!!: 修改后的密码需要更新到/usr/local/omc/etc/restconf.yaml, crontask.yaml和nbi_alarm.json等配置文件中的数据库信息配置项

```
[root@ems ~]# mysql -u root
MariaDB [none]> use mysql

MariaDB [mysql]> update user set password=PASSWORD('*****') where User='root';

MariaDB [mysql]> flush privileges;

MariaDB [mysql]> GRANT ALL PRIVILEGES ON *.* TO 'root'@'%' IDENTIFIED BY '*****'

MariaDB [mysql]> quit
```

通过以下登录命令确认修改成功

```
[root@ems ~]# mysql -u root -h 192.168.0.229 -P 33066 -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 95
Server version: 10.3.35-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

1.2 安装WEB服务(可选)

注意：非第一次安装或者全新安装请跳过此步骤

OMC Web服务可通过nginx代理进行访问，也可不需要代理进行访问

WEB服务: nginx-1.16.1-11.p01.ky10.aarch64

- 软件安装

```
# yum clean all
# yum makecache
# yum -y update
# yum install -y nginx
```

- 验证测试

```
# systemctl enable nginx
# systemctl daemon-reload
# systemctl start nginx
# systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; vendor preset:
disabled)
   Active: active (running) since Tue 2023-05-02 20:57:48 CST; 3h 12min ago
 Main PID: 23163 (nginx)
    Tasks: 5
   Memory: 19.3M
    CGroup: /system.slice/nginx.service
            └─23163 nginx: master process /usr/sbin/nginx
               └─23164 nginx: worker process
                  └─23165 nginx: worker process
```

2 安装配置OMC

2.1 安装OMC软件包

安装OMC软件包

```
# rpm -ivh OMC\ 5GC16.1.1.arm64.rpm
```

安装后的软件包目录树（实际目录树会随版本不同有变化，以下仅供参考）：

```
# cd /usr/local
# tree omc -L 2
omc
├── backup
├── bin
│   ├── cpsshkey.sh
│   ├── crontask
│   ├── importdb.sh
│   ├── nehosts
│   ├── omcsvc.sh
│   ├── restagent
│   └── tables.txt
├── database
├── etc
│   ├── cm
│   ├── crontask.yaml
│   ├── db
│   ├── default
│   ├── restconf.yaml
│   ├── schema
│   └── tasks.yaml
├── htdocs
│   └── front
├── license
├── log
│   ├── crontask.log -> /usr/local/omc/log/crontask.log-20230629
│   ├── crontask.log-20230628
│   ├── crontask.log-20230629
│   ├── nbi_alarm
│   ├── restagent.log -> /usr/local/omc/log/restagent.log-20230629
│   ├── restagent.log-20230628
│   └── restagent.log-20230629
├── software
└── upload
#
```

2.2 修改nginx服务的OMC配置(可选)

安装软件包成功后可以根据实际环境对nginx的OMC配置进行修改，前端Web的默认侦听端口是8888，除了侦听端口，其它不用修改

```
# cd /etc/nginx/conf.d
# cat omc.conf
server {
    listen 8888 default_server;
    root /usr/local/omc/htdocs/front/;
    index index.html index.htm;
    server_name localhost;
```

```
location / {
    try_files $uri $uri/ =404;
}
}
```

2.3 修改OMC Web前端配置

根据实际环境修改如下文件的IP地址和端口，端口需要和OMC的配置文件一致，OMC后端ipv4服务侦听端口默认为3030，ipv6的默认侦听端口为6060：

```
# sudo sed -i.bak 's/192.168.2.119:/192.168.0.229:/g'
/usr/local/omc/htdocs/front/config.js
```

2.4 修改OMC后端配置

说明：

- * 目录/usr/local/omc/etc/default下的文件是OMC程序的初始配置文件，不要修改
- * 拷贝default目录下的文件至/usr/local/omc/etc目录，根据使用环境修改配置文件

```
# cd /usr/local/omc/etc
# cp default/* ../
```

根据实际的运行环境对配置文件restconf.yaml, crontask.yaml, nbi_alarm.json, nbi_agent.json, 4a_agent.yaml及tasks.yaml进行修改，特别是侦听端口和数据库配置信息(包括端口/用户/密码)

2.5 数据库配置

注意：第一次或全新安装执行该步骤，该步骤会清除掉原有的omc数据库

```
# /usr/local/omc/bin/importdb.sh install
```

升级安装执行该步骤

```
# /usr/local/omc/bin/importdb.sh upgrade
```

2.6 防火墙配置

如果系统默认是打开防火墙，需要添加配置文件中配置的服务和端口(端口要和实际环境配置一致)，如下：

```
# firewall-cmd --add-port=33066/tcp --permanent
# firewall-cmd --add-port=8443/tcp --permanent
# firewall-cmd --add-port=8080/tcp --permanent
# firewall-cmd --add-port=4443/tcp --permanent
# firewall-cmd --add-port=8888/tcp --permanent
# firewall-cmd --add-port=3030/tcp --permanent
# firewall-cmd --add-port=4040/tcp --permanent
# firewall-cmd --add-port=5050/tcp --permanent
# firewall-cmd --add-port=6060/tcp --permanent
# firewall-cmd --add-port=31232-31271/tcp --permanent
# firewall-cmd --add-port=2222/tcp --permanent
# firewall-cmd --add-port=6379/tcp --permanent
# firewall-cmd --add-port=2152/udp --permanent
# firewall-cmd --reload
```

2.7 配置被管网元的SSH信任关系

为了便于OMC管理被管网元(包括登录网元，安装软件，备份配置文件和执行命令等操作)，第一次安装时需要配置被管网元与OMC的SSH信任关系，便于无密码SSH登录：

- 修改/usr/local/omc/bin目录下的nehosts文件(*****替换为明文密码)，根据现网环境增加被管网元的ip, 用户和密码信息列表
- 运行脚本cpsshkey.sh, 将公钥拷贝至被管网元
- 运行ssh root@192.168.4.133验证是否可以无密码登录
- 执行成功后, 为了防止密码泄露，清除掉nehosts文件的密码信息

```
[root@ems bin]# cat nehosts
# host user password
# Example: 192.168.4.133 root password
192.168.4.133 root *****
192.168.4.134 root *****
[root@ems bin]# ./cpsshkey.sh
[root@ems bin]# ssh root@192.168.4.133
```

3 启动和停止OMC

3.1 系统服务启动

```
# systemctl enable restagent.service
# systemctl enable crontask.service
# systemctl enable sshsvc.service
# systemctl enable captrace.service
# systemctl daemon-reload
```

执行上述命令后，它被设置为作为 **systemd** 服务运行。

3.2 手动启动/查询/停止OMC

您可以按如下方式启动，查询状态，停止和重新启动**OMC**，以下命令可通过 `/usr/local/omc/bin/omcsvc.sh` 脚本一次性执行：

```
# cd /usr/local/omc/bin
# ./omcsvc.sh start/stop/restart/status
```

单个**OMC**进程的执行命令如下，启动/状态/停止/重启：

```
# systemctl start restagent.service
# systemctl start crontask.service
# systemctl start sshsvc.service
# systemctl start captrace.service
```

```
# systemctl status restagent.service
# systemctl status crontask.service
# systemctl status sshsvc.service
# systemctl status captrace.service
```

```
# systemctl stop restagent.service
# systemctl stop crontask.service
# systemctl stop sshsvc.service
# systemctl stop captrace.service
```



```
# systemctl restart restagent.service
# systemctl restart crontask.service
# systemctl restart sshsvc.service
# systemctl restart captrace.service
```

4 问题定位跟踪

OMC `restagent`程序的配置文件在`restagent.yaml`，`trace`级别的日志通常在调试系统时使用，正常运行级别为`debug`，如下配置项所示：

```
logger:
  file: /usr/local/omc/log/restagent.log
  level: debug
  duration: 24
  count: 90
```

OMC `crontask`程序的配置文件在`crontask.yaml`，`trace`级别的日志通常在调试系统时使用，正常运行级别为`debug`，如下配置项所示：

```
logger:
  file: /usr/local/omc/log/crontask.log
  level: trace
  duration: 24
  count: 90
```

OMC `sshsvc`程序的配置文件在`sshsvc.yaml`，日志配置项如下所示：

```
logger:
  file: /usr/local/omc/log/sshsvc.log
  level: debug
  duration: 24
  count: 30
```

OMC `captrace`程序的配置文件在`capconf.yaml`，日志配置项如下所示：

```
logger:
  file: /usr/local/omc/log/captrace.log
  level: trace
```

duration: 24
count: 10

5 管理核心网系统

通过浏览器输入: <http://192.168.0.229:8888> (nginx代理模式)

或: <http://192.168.0.229:8080> (非nginx代理模式)