

解决方案实践

快速部署高可用 MHA-MySQL 集群

文档版本	1.0.1
发布日期	2023-02-28



版权所有 © 华为技术有限公司 2024。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

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。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

安全声明

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：

<https://www.huawei.com/cn/psirt/vul-response-process>

如企业客户须获取漏洞信息，请参见如下网址：

<https://securitybulletin.huawei.com/enterprise/cn/security-advisory>

目 录

1 方案概述.....	1
2 资源和成本规划.....	3
3 实施步骤.....	5
3.1 准备工作.....	5
3.2 快速部署.....	9
3.3 开始使用.....	15
3.4 快速卸载.....	26
4 附录.....	27
5 修订记录.....	28

1 方案概述

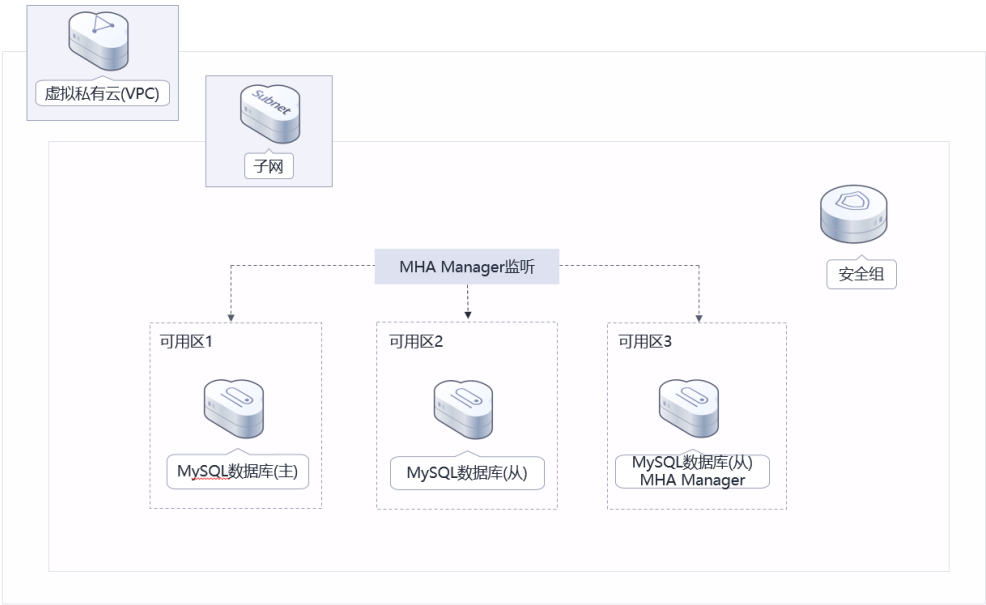
应用场景

该解决方案基于开源软件MHA构建，快速帮助用户在华为云服务器上完成高可用的MySQL集群部署。**MHA**（Master High Availability）是一款成熟且开源的MySQL高可用程序，主要提供心跳检测、主从复制、故障转移，并发送告警邮件。适用于需要高可用性、数据完整性以及近乎不间断的主服务器维护等场景。

方案架构

该解决方案能够快速帮助您在华为云上搭建基于开源MHA的Mysql集群环境，部署架构如下图所示：

图 1-1 方案架构



该解决方案会部署如下资源：

- 创建3台Linux弹性云服务器，加入同一个云服务器组，配置反亲和策略，跨可用区部署，并分别安装MHA和MySQL软件。

- 创建虚拟IP(VIP)，用于MySQL数据库主从切换。
- 创建3条弹性公网IP，用于MHA和MySQL环境部署及后期发生故障后发送报警邮件。
- 创建安全组，可以保护弹性云服务器的网络安全，通过配置安全组规则，限定云服务器的访问端口。

方案优势

- 高可靠
云服务器配置反亲和性策略，跨可用区部署，提供多可用区容灾能力；安装MHA故障切换和主从复制的软件，能够快速自动完成故障切换以及最大程度上保证数据一致性。
- 节约成本
MHA 由 MHA Manager 和 MHA Node 组成，MHA 管理器部署在其中一台从数据库服务器上，因此不需要额外的服务器。
- 一键部署
一键轻松部署，即可完成资源的快速发放以及高可用MySQL集群环境的部署。

约束与限制

- 部署该解决方案之前，您需注册华为账号并开通华为云，完成实名认证，且账号不能处于欠费或冻结状态。如果计费模式选择“包年包月”，请确保账户余额充足以便一键部署资源的时候可以自动支付；或者在一键部署的过程进入[费用中心](#)，找到“待支付订单”并手动完成支付。

2 资源和成本规划

该解决方案主要部署如下资源，不同产品的花费仅供参考，具体请参考华为云[官网价格](#)，实际以收费账单为准：

表 2-1 资源和成本规划（ 按需计费 ）

华为云服务	配置示例	每月预估花费
弹性云服务器 ECS	<ul style="list-style-type: none">• 按需计费：0.58元/小时• 区域：华北-北京四• 计费模式：按需计费• 规格：X86计算 ECS c6.large.2 2vCPUs 4GiB• 镜像：CentOS 7.6 64bit• 系统盘：高IO 40GB• 数据盘：通用SSD 100GB• 购买量：3	$0.58 * 24 * 30 * 3 = 1252.8\text{元}$
弹性公网IP EIP	<ul style="list-style-type: none">• 按需计费：0.34元/MBit/s/小时• 区域：华北-北京四• 计费模式：按需计费• 线路：动态BGP• 公网带宽：按带宽计费• 带宽大小：5Mbit/s• 购买量：3(配置完成后可手动释放2个，将不再计费)	$0.34 * 24 * 30 * 3 = 734.4\text{元}$
合计	-	1987.2元

表 2-2 资源和成本规划（包年包月）

华为云服务	配置示例	每月预估花费
弹性云服务器 ECS	<ul style="list-style-type: none">区域：华北-北京四计费模式：包年包月规格：X86计算 ECS c6.large.2 2vCPUs 4GiB镜像：CentOS 7.6 64bit系统盘：高IO 40GB数据盘：通用SSD 100GB购买量：3	292.50 * 3 =877.5元
弹性公网IP EIP	<ul style="list-style-type: none">区域：华北-北京四计费模式：包年包月线路：动态BGP公网带宽：按带宽计费带宽大小：5Mbit/s购买量：3(配置完成后可手动释放2个，将不再计费)	115 *3 = 345元
合计	-	1222.5元

3 实施步骤

- 3.1 准备工作
- 3.2 快速部署
- 3.3 开始使用
- 3.4 快速卸载

3.1 准备工作

创建 rf_admin_trust 委托

步骤1 进入华为云官网，打开[控制台管理](#)界面，鼠标移动至个人账号处，打开“统一身份认证”菜单。

图 3-1 控制台管理界面



图 3-2 统一身份认证菜单



步骤2 进入“委托”菜单，搜索“rf_admin_trust”委托。

图 3-3 委托列表



- 如果委托存在，则不用执行接下来的创建委托的步骤
- 如果委托不存在时执行接下来的步骤创建委托

步骤3 单击步骤2界面中右上角的“创建委托”按钮，在委托名称中输入“rf_admin_trust”，选择“普通账号”，委托的账号，输入“op_svc_IAC”，单击“下一步”。

图 3-4 创建委托

委托 / 创建委托

★ 委托名称

rf_admin_trust

★ 委托类型

☒ 普通帐号
将帐号内资源的操作权限委托给其他华为云帐号。

☐ 云服务
将帐号内资源的操作权限委托给华为云服务。

★ 委托的帐号

op_svc_IAC

★ 持续时间

永久

描述

请输入委托信息。

0/255

下一步

取消

步骤4 在搜索框中输入“Tenant Administrator”权限，并勾选搜索结果。

图 3-5 选择策略

1 选择策略 2 设置委托授权范围 3 完成

委托“rf_admin_trust”将拥有所选策略

策略已选(1)

从其他区域项目复制权限

全部策略

所有云服务

Tenant Administrator

X

Q

名称	类型	系统角色
<input checked="" type="checkbox"/> Tenant Administrator	全部策略	系统角色

步骤5 选择“所有资源”，并单击下一步完成配置。

图 3-6 设置授权范围

1 选择策略 2 设置委托授权范围 3 完成

根据当前所选策略的策略，系统推荐以下授权范围方案，更便于您最小化授权，了解如何根据您的应用场景选择合理的授权范围方案

选择授权范围方案

☒ 所有资源

授权后，IAM用户可以对资源权限使用帐号中所有资源，包括企业项目、区域项目和企业级资源。

☐ 展开其他方案

步骤6 “委托”列表中出现“rf_admin_trust”委托则创建成功。

图 3-7 委托列表



----结束

获取邮箱授权码

当MHA进行failover或由于错误停止时，可以使用send_report脚本以邮件报警的方式来获得failover报告，从而使得用户可以及时了解现在的数据库状态。请参考[示例获取网易邮箱授权码](#)或[获取QQ邮箱授权码](#)获取您部署MHA时发送邮件账户的授权码，以便MHA发送故障转移或其他错误停止的报警邮件。

图 3-8 开启 POP3/SMTP 服务



图 3-9 授权码



3.2 快速部署

本章节主要帮助用户快速部署该解决方案。

表 3-1 参数填写说明

参数名称	类型	是否必填	参数解释	默认值
vpc_name	string	必填	虚拟私有云名称，该模板使用新建VPC，不支持重名。取值范围：1-54个字符，支持数字、字母、中文、_(下划线)、-（中划线）、.（点）。	highly-available-mha-mysql-cluster-demo
security_group_name	string	必填	安全组名称，该模板使用新建安全组。取值范围：1-64个字符，支持数字、字母、中文、_(下划线)、-（中划线）、.（点）。	highly-available-mha-mysql-cluster-demo
bandwidth_size	number	必填	带宽大小，该模板计费方式为按带宽计费。取值范围：1-2,000Mbit/s。	5

参数名称	类型	是否必填	参数解释	默认值
ecs_name	string	必填	弹性云服务器名称，不支持重名。取值范围：1-57个字符，支持小写字母、数字、_（下划线）、-（中划线）。	highly-available-mha-mysql-cluster-demo
ecs_flavor	string	必填	弹性云服务器规格，具体请参考官网 弹性云服务器规格清单 。	c6.large.2 (c6 2vCPUs 4GiB)
ecs_password	string	必填	弹性云服务器、MySQL复制账户及MHA管理账户初始化密码，创建完成后，请参考 在控制台重置弹性云服务器密码 及时重置密码。取值范围：长度为8-26个字符，密码至少包含大写字母、小写字母、数字和特殊字符（\$!@%-_+=[:./^,{}?）中的三种，密码不能包含用户名或用户名的逆序。弹性云服务器管理员账户默认root，MySQL主从复制账户默认repl，MHA管理账户默认mha。	空
system_disk_size	number	必填	弹性云服务器系统盘大小，默认高IO型，不支持缩盘。取值范围：40-1,024GB。	40
data_disk_size	number	必填	弹性云服务器数据盘大小，默认通用SSD型，取值范围：10-32,768GB。	100
sender_email_address	string	必填	发件人邮箱地址，用于MHA故障转移发送报警邮件。例如： mha@huawei.com。	空
recipient_email_address	string	必填	收件人邮箱地址，用于接收报警邮件。例如： recipient@huawei.com	空
email_authorization_code	string	必填	邮箱账户授权码，用于授权MHA使用发件人邮箱账户发送邮件，获取请参考 3.1准备工作 。	空

参数名称	类型	是否必填	参数解释	默认值
charging_mode	string	必填	计费模式，默认自动扣费，可选值为：postPaid（按需计费）、prePaid（包年包月）。	postPaid
charge_period_unit	string	必填	订购周期类型，仅当charge_mode为prePaid（包年/包月）生效。取值范围：month（月），year（年）。	month
charge_period	number	必填	订购周期，仅当charge_mode为prePaid（包年/包月）生效。取值范围： charge_period_unit=month（周期类型为月）时，取值为1-9； charge_period_unit=year（周期类型为年）时，取值为1-3。	1

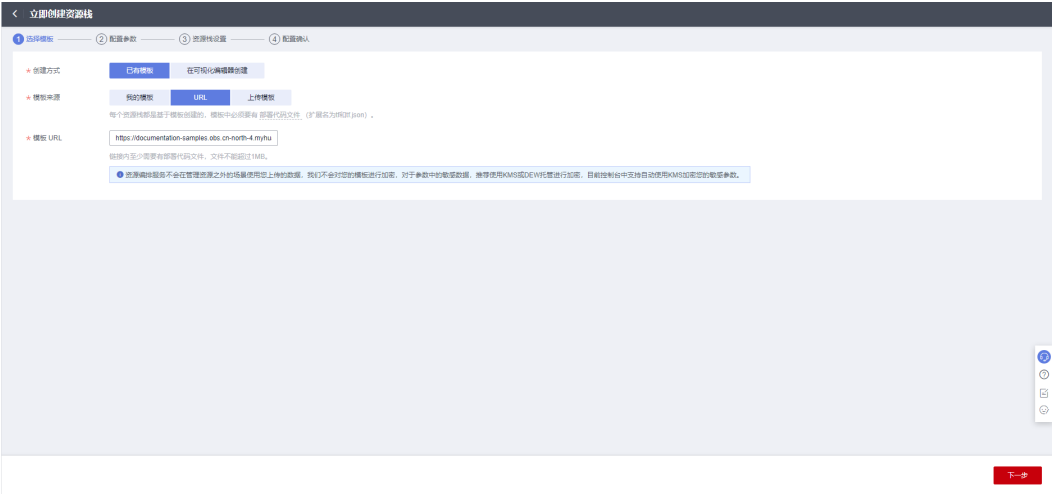
步骤1 登录华为云解决方案实践，选择“**快速部署高可用MHA-MySQL集群**”，跳转至该解决方案一键部署界面，数据中心下拉菜单可以选择需要部署的区域。

图 3-10 解决方案实施库



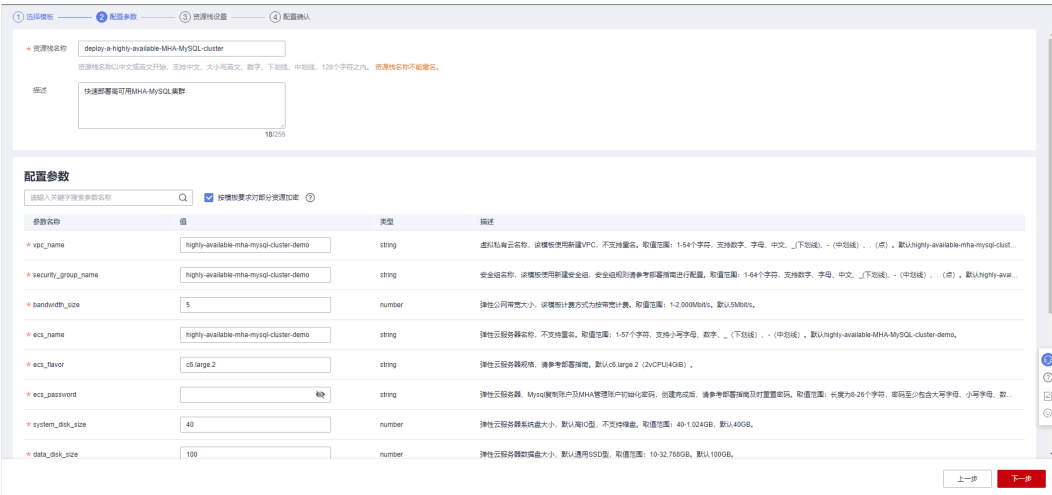
步骤2 单击“一键部署”，跳转至该解决方案创建资源栈部署界面。

图 3-11 创建资源栈



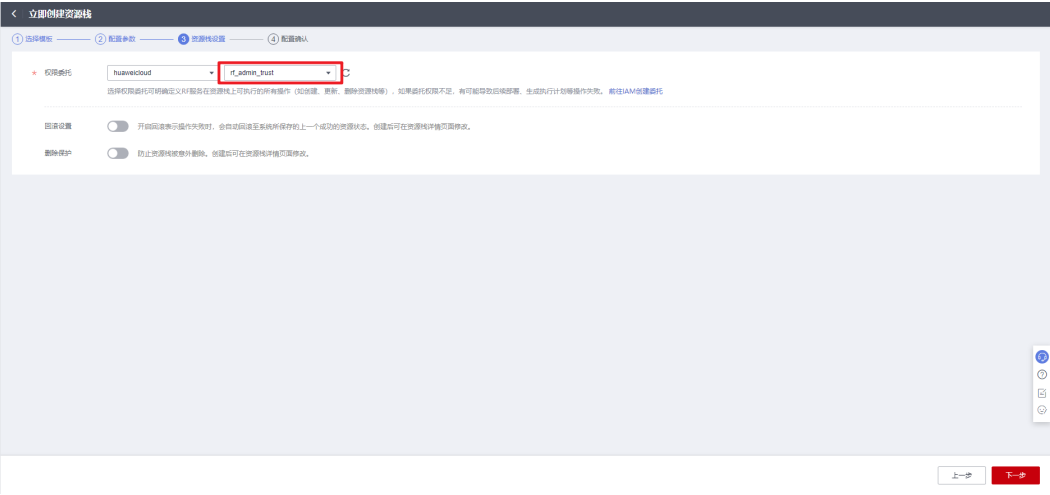
步骤3 单击“下一步”，参考表3-1完成自定义参数填写。

图 3-12 参数配置



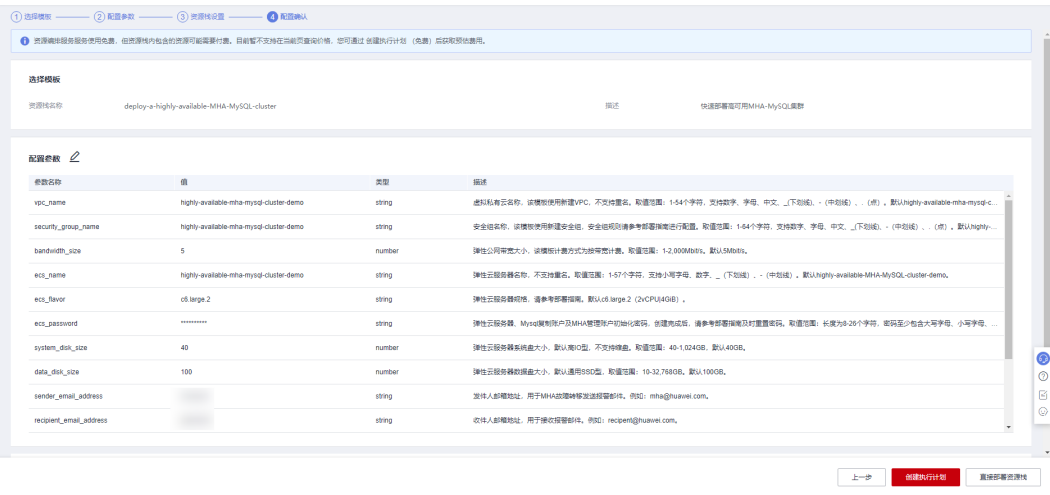
步骤4 在资源设置界面中，“权限委托”下拉框中选择“rf_admin_trust”委托，单击“下一步”。

图 3-13 资源栈设置



步骤5 在配置确认界面中，单击“创建执行计划”。

图 3-14 确认配置



步骤6 在弹出的创建执行计划框中，自定义填写执行计划名称，单击“确定”。

图 3-15 创建执行计划

创建执行计划

- 在部署资源栈之前，您可以通过创建执行计划提前浏览您准备部署的资源栈信息、检查各项配置，以此评估对正在运行资源的影响。
- 该创建免费，**但会占用资源栈配额**。创建后会生成一个未开通资源的资源栈，并在执行计划详情提示预计费用。

★ 执行计划名称

executionPlan_20230216_1745_pqtk

描述

请输入对执行计划的描述

0/255

确定

取消

步骤7 单击“部署”并在弹出的执行计划确认框中单击“执行”，等待资源部署。

图 3-16 部署执行计划

deploy-a-highly-available...

删除 更新模板或参数

基本信息 资源 输出 事件 模板 执行计划

部署

请输入关键字

Q C

执行计划名称ID	状态	占用额度①	创建时间	描述	操作
executionPlan_20230216_1745_pqtk 1548b9b6-2540-4468-9969-88050ee0b55	创建成功，待部署	部署占用0/10	2023/02/16 17:48:07 GMT+08:00	...	<div>删除 部署</div>

步骤8 （可选）如果计费模式选择“包年包月”，在余额不充足的情况下（所需总费用请参考表2-2）请及时登录费用中心，手动完成待支付订单的费用支付。

步骤9 待“事件”中出现“Apply required resource success”，表示该解决方案已经部署完成。

图 3-17 资源创建成功

deploy-a-highly-available...

删除 更新模板或参数

基本信息 资源 输出 事件 模板 执行计划

请输入关键字

Q C

事件时间 正	事件类型	事件描述	资源名称/类型	关联资源ID
2023/02/16 17:52:35 GMT+08:00	LOG	Apply required resource success.
2023/02/16 17:52:28 GMT+08:00	-	Apply completed Resources: 14 added, 0 changed, 0 destroyed.

步骤10 单击“输出”，查看虚拟IP及弹性云服务器相关信息。

图 3-18 输出信息

基本消息 资源 事件 输出 模板 执行计划		
名称	类型	值
温馨提示	string	环境初始化的需要15分钟，请耐心等待初始化完成。主数据库私有IP地址：192.168.100.111，从...

----结束

3.3 开始使用

安全组规则修改（可选）

须知

- MySQL服务端口号默认3306，默认对该方案创建的VPC子网网段放开，请参考[修改安全组规则](#)，配置IP地址白名单，以便能正常访问服务。
- 该解决方案使用22端口用来远程登录弹性云服务器，默认对该方案创建的VPC子网网段放开，请参考[修改安全组规则](#)，配置IP地址白名单，以便能正常访问服务。

安全组实际是网络流量访问策略，包括网络流量入方向规则和出方向规则，通过这些规则为安全组内具有相同保护需求并且相互信任的云服务器、云容器、云数据库等实例提供安全保护。

如果您的实例关联的安全组策略无法满足使用需求，比如需要添加、修改、删除某个TCP端口，请参考以下内容进行修改。

- 添加安全组规则：根据业务使用需求需要开放某个TCP端口，请参考[添加安全组规则](#)添加入方向规则，打开指定的TCP端口。
- 修改安全组规则：安全组规则设置不当会造成严重的安全隐患。您可以参考[修改安全组规则](#)，来修改安全组中不合理的规则，保证云服务器等实例的网络安全。
- 删除安全组规则：当安全组规则入方向、出方向源地址/目的地址有变化时，或者不需要开放某个端口时，您可以参考[删除安全组规则](#)进行安全组规则删除。

步骤1 修改初始化密码。登录[华为云服务器控制台](#)，参考[在控制台重置弹性云服务器密码](#)，进行密码重置。

图 3-19 重置密码



步骤2 登录[华为云服务器控制台](#)，查看一键部署创建的弹性云服务及其绑定的弹性公网IP。

图 3-20 弹性云服务器



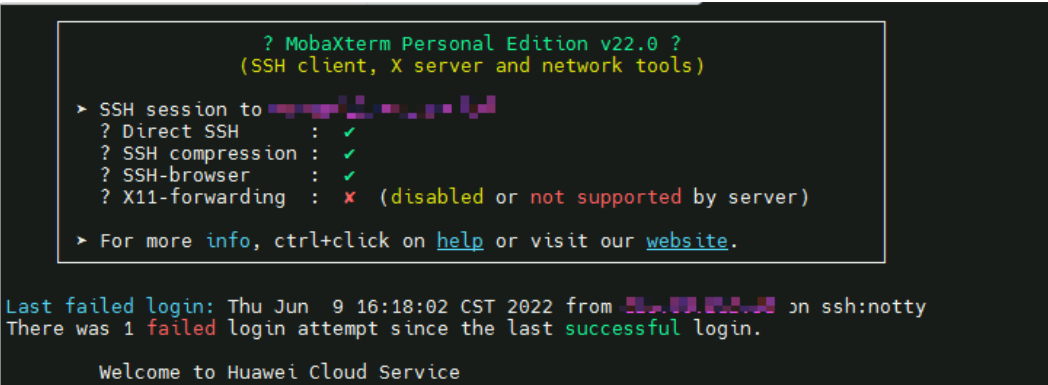
步骤3 打开业务虚拟机所属的[子网](#)，单击“IP地址管理”查看虚拟IP。

图 3-21 VIP



步骤4 使用远程连接工具，登录三台弹性云服务器。管理员账户为root。

图 3-22 使用远程连接软件登录



步骤5 或者通过[华为云服务器控制台](#)，选择以弹性云服务器名称为前缀的弹性云服务器，选择“远程登录”方式登录服务器。

图 3-23 VNC 远程登录

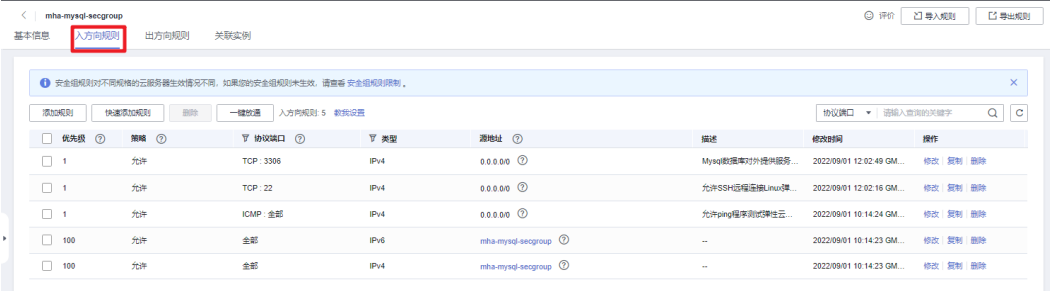


图 3-24 VNC 登录界面



步骤6 查看安全组。打开[安全组控制台](#)，找到以虚拟私有云VPC名称为前缀的安全组，单击进入查看该方案创建的安全组规则，入方向规则默认22端口、3306端口全放开，请参考说明进行修改。

图 3-25 安全组



步骤7 释放“主数据库”及“从数据库”绑定的弹性公网IP。登录[华为云服务器控制台](#)，选择以“master”、“slave1”为后缀的两台主从数据库服务器，分别单击其绑定的弹性公网IP，单击“解绑>是”进行解绑（注意请勿释放从数据库slave2绑定的公网IP，其上安装了MHA服务软件，故障切换时需要发送报警邮件，如果释放会导致报警邮件发送失败）。

图 3-26 解绑 EIP

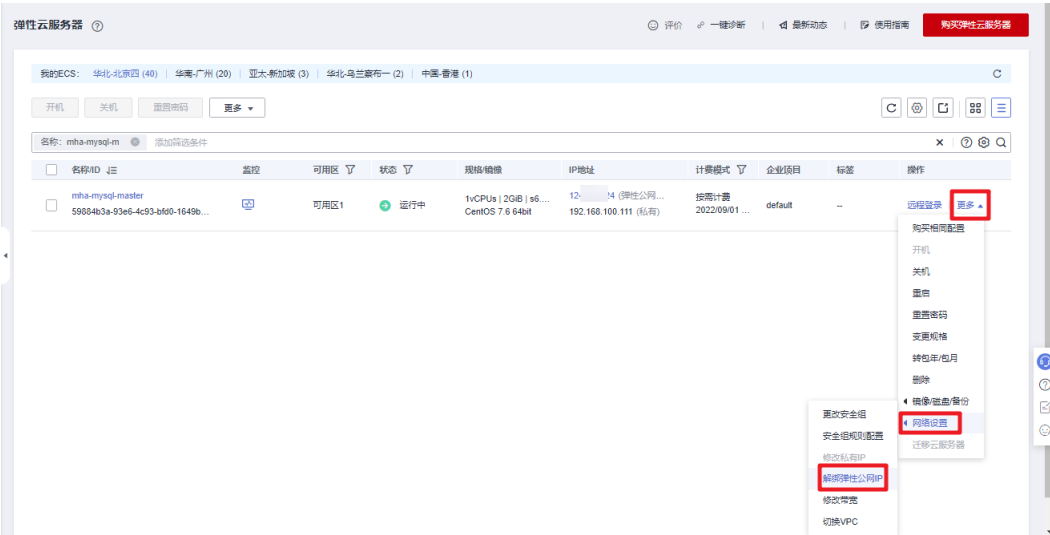


图 3-27 确认解绑



步骤8 释放弹性公网IP。分别单击两台弹性云服务弹性公网IP右侧“更多>释放>是”，释放弹性公网IP。

图 3-28 释放 EIP



图 3-29 确认释放



----结束

环境初始化

服务器重启后，有可能导致集群服务停止，需要用户重新手动启动服务，下面是手动开启服务的介绍。

步骤1 重新配置虚拟IP。登录主数据库服务器，执行ifconfig eth0:1 VIP/24命令，VIP为**3.3步骤3**或**3.2步骤9**中获取的VIP地址。

图 3-30 重新配置 VIP

```
[root@mha-mysql-master ~]# ifconfig eth0:1 192.168.100.95/24
[root@mha-mysql-master ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.111 netmask 255.255.255.0 broadcast 192.168.100.255
    inet6 fe80::f816:3eff:fec9:8590 prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:c9:85:90 txqueuelen 1000 (Ethernet)
    RX packets 431 bytes 72480 (70.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 347 bytes 57795 (56.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0:1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.95 netmask 255.255.255.0 broadcast 192.168.100.255
    ether fa:16:3e:c9:85:90 txqueuelen 1000 (Ethernet)

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@mha-mysql-master ~]#
```

步骤2 开启MHA服务。登录MHA Manager服务器名称后缀为slave2，执行mha_app1_start命令启动MHA监控服务。检查mha状态为“running”，则集群服务启动成功。

图 3-31 开启 MHA 服务

```
[root@mha-mysql-slave2 ~]# mha_app1_start
[1] 1553
[root@mha-mysql-slave2 ~]# mha_app1_status
app1 (pid:1553) is running(0:PING_OK), master:192.168.100.111
[root@mha-mysql-slave2 ~]#
```

----结束

MySQL 使用

该解决方案默认创建三台弹性云服务器，分别安装Mysql-5.7.34版本的数据库，一主（后缀名为master）两从（后缀名为slave1、slave2）。MySQL管理用户用户组默认创建mysql，服务端默认开启3306。主数据库上创建主从复制账户repl，密码同ECS初始化密码，允许登录地址为192.168.100.0/24网段。MHA Manger默认安装在从数据库slave2上，管理账户mha，密码同ECS初始化密码。

查看主从数据库复制状态

```
SHOW SLAVE STATUS\G;
```


图 3-32 从数据库复制状态

```
> show slave status\G;
***** 1. row *****
Slave_IO_State: Waiting for master to send event
Master_Host: 192.168.100.111
Master_User: repl
Master_Port: 3306
Connect_Retry: 10
Master_Log_File: mysql_bin.000002
Read_Master_Log_Pos: 740
Relay_Log_File: mha-mysql-slave2-relay-bin.000002
Relay_Log_Pos: 953
Relay_Master_Log_File: mysql_bin.000002
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Replicate_Do_DB:
Replicate_Ignore_DB:
Replicate_Do_Table:
Replicate_Ignore_Table:
Replicate_Wild_Do_Table:
Replicate_Wild_Ignore_Table:
Last_Errno: 0
Last_Error:
Skip_Counter: 0
Exec_Master_Log_Pos: 740
Relay_Log_Space: 1171
Until_Condition: None
Until_Log_File:
Until_Log_Pos: 0
Master_SSL_Allowed: No
Master_SSL_CA_File:
Master_SSL_CA_Path:
Master_SSL_Cert:
Master_SSL_Cipher:
Master_SSL_Key:
Seconds_Behind_Master: 0
Master_SSL_Verify_Server_Cert: No
Last_IO_Errno: 0
Last_IO_Error:
Last_SQL_Errno: 0
Last_SQL_Error:
Replicate_Ignore_Server_Ids:
Master_Server_Id: 111
```

在主数据库Master上创建复制账户命令（默认已创建repl账户，密码为ECS初始化密码）：

```
mysql -uroot -S /tmp/mysql.sock -e "grant replication slave on *.* to 账户@'%' identified by '密码';"
例如
mysql -uroot -S /tmp/mysql.sock -e "grant replication slave on *.* to repl@'192.168.100.%' identified by '123';"
```

在主数据库Master上创建MHA管理账户（默认已创建mha账户，密码为ECS初始化密码）：

```
mysql -uroot -e "GRANT ALL PRIVILEGES ON *.* TO 账户@'允许登录地址' IDENTIFIED BY '密码';"
例如:
mysql -uroot -e "GRANT ALL PRIVILEGES ON *.* TO mha@'192.168.100.%' IDENTIFIED BY '123';"
```

修改MySQL账户密码

```
update mysql.user set authentication_string=password('密码') where user='账户' and Host = 'localhost';
例如:
update mysql.user set authentication_string=password('123456') where user='repl' and Host = '192.168.100.%';
```

说明

- 修改MySQL数据库中账户密码，如修改了主从复制账户密码，须同时在从数据库重新配置主数据库信息：

```
> CHANGE MASTER TO
MASTER_HOST='192.168.100.111',
MASTER_USER='repl',
MASTER_PASSWORD='密码',
MASTER_PORT=3306,
MASTER_CONNECT_RETRY=10,
MASTER_AUTO_POSITION=1;
> START SLAVE;
```

- 修改主从复制账户及MHA管理账户密码后，需要在MHA Manager（即从数据库slave2）app1配置文件中同步修改密码：
vim /datadisk/mha/conf/app1.cnf修改password字段值。

MHA 使用

MHA Manager安装在从数据库slave2中，一个MHA可以管理多套主从，只需要创建不同的配置文件即可，方案初始化时默认创建一套主从配置文件，MHA管理的用户默认mha，密码为ECS初始化密码，配置文件路径：/datadisk/mha/conf/app1.cnf。

MHA管理命令：

主从数据库互信检查

```
masterha_check_ssh --conf=/datadisk/mha/conf/app1.cnf
```

见图15，出现“All SSH connection tests passed successfully”则说明三台服务器可以互相免密登录

图 3-33 互信检查

```
[root@mha-mysql-slave2 ~]# masterha_check_ssh --conf=/mha/conf/app1.cnf
Thu Sep 1 10:50:48 2022 - [warning] Global configuration file /etc/masterha_default.cnf not found. Skipping.
Thu Sep 1 10:50:48 2022 - [info] Reading application default configuration from /mha/conf/app1.cnf..
Thu Sep 1 10:50:48 2022 - [info] Starting SSH connection tests..
Thu Sep 1 10:50:49 2022 - [debug]
Thu Sep 1 10:50:48 2022 - [debug] Connecting via SSH from root@192.168.100.111(192.168.100.111:22) to root@192.168.100.112(192.168.100.112:22)..
Thu Sep 1 10:50:49 2022 - [debug] ok.
Thu Sep 1 10:50:49 2022 - [debug] Connecting via SSH from root@192.168.100.111(192.168.100.111:22) to root@192.168.100.113(192.168.100.113:22)..
Thu Sep 1 10:50:49 2022 - [debug] ok.
Thu Sep 1 10:50:50 2022 - [debug]
Thu Sep 1 10:50:49 2022 - [debug] Connecting via SSH from root@192.168.100.112(192.168.100.112:22) to root@192.168.100.111(192.168.100.111:22)..
Thu Sep 1 10:50:49 2022 - [debug] ok.
Thu Sep 1 10:50:49 2022 - [debug] Connecting via SSH from root@192.168.100.112(192.168.100.112:22) to root@192.168.100.113(192.168.100.113:22)..
Thu Sep 1 10:50:50 2022 - [debug] ok.
Thu Sep 1 10:50:49 2022 - [debug] Connecting via SSH from root@192.168.100.113(192.168.100.113:22) to root@192.168.100.111(192.168.100.111:22)..
Thu Sep 1 10:50:50 2022 - [debug] ok.
Thu Sep 1 10:50:50 2022 - [debug] Connecting via SSH from root@192.168.100.113(192.168.100.113:22) to root@192.168.100.112(192.168.100.112:22)..
Thu Sep 1 10:50:50 2022 - [info] All SSH connection tests passed successfully.
[root@mha-mysql-slave2 ~]#
```

主从数据库状态检查

```
masterha_check_repl --conf=/datadisk/mha/conf/app1.cnf
```

见图16，出现“MySQL Replication Health is OK”则说明主从状态良好

图 3-34 主从数据库状态

```
Thu Sep 1 10:52:10 2022 - [info] Alive Servers:
Thu Sep 1 10:52:10 2022 - [info] 192.168.100.111(192.168.100.111:3306)
Thu Sep 1 10:52:10 2022 - [info] 192.168.100.112(192.168.100.112:3306)
Thu Sep 1 10:52:10 2022 - [info] 192.168.100.113(192.168.100.113:3306)
Thu Sep 1 10:52:10 2022 - [info] Alive Slaves:
Thu Sep 1 10:52:10 2022 - [info] 192.168.100.112(192.168.100.112:3306) Version=5.7.34-log (oldest major version be
Thu Sep 1 10:52:10 2022 - [info] GTID ON
Thu Sep 1 10:52:10 2022 - [info] Replicating from 192.168.100.111(192.168.100.111:3306)
Thu Sep 1 10:52:10 2022 - [info] Primary candidate for the new Master (candidate_master is set)
Thu Sep 1 10:52:10 2022 - [info] 192.168.100.113(192.168.100.113:3306) Version=5.7.34-log (oldest major version be
Thu Sep 1 10:52:10 2022 - [info] GTID ON
Thu Sep 1 10:52:10 2022 - [info] Replicating from 192.168.100.111(192.168.100.111:3306)
Thu Sep 1 10:52:10 2022 - [info] Current Alive Master: 192.168.100.111(192.168.100.111:3306)
Thu Sep 1 10:52:10 2022 - [info] Checking slave configurations..
Thu Sep 1 10:52:10 2022 - [info] read_only=1 is not set on slave 192.168.100.112(192.168.100.112:3306).
Thu Sep 1 10:52:10 2022 - [info] read_only=1 is not set on slave 192.168.100.113(192.168.100.113:3306).
Thu Sep 1 10:52:10 2022 - [info] Checking replication filtering settings..
Thu Sep 1 10:52:10 2022 - [info] binlog_do_db= , binlog_ignore_db=
Thu Sep 1 10:52:10 2022 - [info] Replication filtering check ok.
Thu Sep 1 10:52:10 2022 - [info] GTID (with auto-pos) is supported. Skipping all SSH and Node package checking.
Thu Sep 1 10:52:10 2022 - [info] Checking SSH publickey authentication settings on the current master..
Thu Sep 1 10:52:10 2022 - [info] HealthCheck: SSH to 192.168.100.111 is reachable.
Thu Sep 1 10:52:10 2022 - [info]
192.168.100.111(192.168.100.111:3306) (current master)
+-192.168.100.112(192.168.100.112:3306)
+-192.168.100.113(192.168.100.113:3306)

Thu Sep 1 10:52:10 2022 - [info] Checking replication health on 192.168.100.112..
Thu Sep 1 10:52:10 2022 - [info] ok.
Thu Sep 1 10:52:10 2022 - [info] Checking replication health on 192.168.100.113..
Thu Sep 1 10:52:10 2022 - [info] ok.
Thu Sep 1 10:52:10 2022 - [info] Checking master_ip_failover_script status:
Thu Sep 1 10:52:10 2022 - [info] /mha/scripts/master_ip_failover --command=status --ssh_user=root --orig_master_hos
ig_master_port=3306

IN SCRIPT TEST====/sbin/ifconfig eth0:1 down====/sbin/ifconfig eth0:1 192.168.100.99/24===

Checking the Status of the script.. OK
Thu Sep 1 10:52:10 2022 - [info] OK.
Thu Sep 1 10:52:10 2022 - [warning] shutdown_script is not defined.
Thu Sep 1 10:52:10 2022 - [info] Got exit code 0 (Not master dead).

MySQL Replication Health is OK.
```

启动MHA监控
mha_app1_start

检查MHA运行状态
mha_app1_status

停止MHA监控
mha_app1_stop

查看MHA日志
tail -f /datadisk/mha/logs/manager

故障模拟

步骤1 在MHA Manager上查看日志变化：
tail -f /datadisk/mha/logs/manager

图 3-35 日志

```
[root@mha-mysql-slave2 ~]# mha_app1_status
app1 (pid:2426) is running(0:PING_OK), master:192.168.100.111
[root@mha-mysql-slave2 ~]# tail -f /mha/logs/manager

IN SCRIPT TEST====/sbin/ifconfig eth0:1 down====/sbin/ifconfig eth0:1 192.168.100.99/24===

Checking the Status of the script.. OK
Thu Sep 1 11:30:37 2022 - [info] OK.
Thu Sep 1 11:30:37 2022 - [warning] shutdown_script is not defined.
Thu Sep 1 11:30:37 2022 - [info] Set master ping interval 3 seconds.
Thu Sep 1 11:30:37 2022 - [warning] secondary_check_script is not defined. It is highly recommended setting it to check master reachability from two or more routes.
Thu Sep 1 11:30:37 2022 - [info] Starting ping health check on 192.168.100.111(192.168.100.111:3306)..
Thu Sep 1 11:30:37 2022 - [info] Ping(SELECT) succeeded, waiting until MySQL doesn't respond..
```

步骤2 关闭主数据服务：
systemctl stop mysqld

步骤3 回到MHA Manager服务器，查看日志，出现 “Master failover tocompleted successfully.”

图 3-36 故障转移

```
Enabling the VIP - 192.168.100.99/24 on the new master - 192.168.100.112
Thu Sep 1 11:34:07 2022 - [info] OK.
Thu Sep 1 11:34:07 2022 - [info] ** Finished master recovery successfully.
Thu Sep 1 11:34:07 2022 - [info] * Phase 3: Master Recovery Phase completed.
Thu Sep 1 11:34:07 2022 - [info] * Phase 4: Slaves Recovery Phase..
Thu Sep 1 11:34:07 2022 - [info]
Thu Sep 1 11:34:07 2022 - [info] * Phase 4.1: Starting Slaves in parallel..
Thu Sep 1 11:34:07 2022 - [info]
Thu Sep 1 11:34:07 2022 - [info] -- Slave recovery on host 192.168.100.113(192.168.100.113:3306) started, pid: 14227. Check tmp log /mha/logs/a
Thu Sep 1 11:34:09 2022 - [info]
Thu Sep 1 11:34:09 2022 - [info] Log messages from 192.168.100.113 ...
Thu Sep 1 11:34:09 2022 - [info]
Thu Sep 1 11:34:07 2022 - [info] Resetting slave 192.168.100.113(192.168.100.113:3306) and starting replication from the new master 192.168.10
Thu Sep 1 11:34:07 2022 - [info] Executed CHANGE MASTER.
Thu Sep 1 11:34:08 2022 - [info] Slave started.
Thu Sep 1 11:34:08 2022 - [info] gtid_wait(2ed2aeb1-299c-11ed-8501-fa163e4d0914:1-2) completed on 192.168.100.113(192.168.100.113:3306). Execu
Thu Sep 1 11:34:09 2022 - [info] End of log messages from 192.168.100.113.
Thu Sep 1 11:34:09 2022 - [info] -- Slave on host 192.168.100.113(192.168.100.113:3306) started.
Thu Sep 1 11:34:09 2022 - [info] All new slave servers recovered successfully.
Thu Sep 1 11:34:09 2022 - [info]
Thu Sep 1 11:34:09 2022 - [info] * Phase 5: New master cleanup phase..
Thu Sep 1 11:34:09 2022 - [info]
Thu Sep 1 11:34:09 2022 - [info] Resetting slave info on the new master..
Thu Sep 1 11:34:09 2022 - [info] 192.168.100.112: Resetting slave info succeeded.
Thu Sep 1 11:34:09 2022 - [info] Master failover to 192.168.100.112(192.168.100.112:3306) completed successfully.
Thu Sep 1 11:34:09 2022 - [info] Deleted server1 entry from /mha/conf/app1.cnf .
Thu Sep 1 11:34:09 2022 - [info]

----- Failover Report -----

app1: MySQL Master failover 192.168.100.111(192.168.100.111:3306) to 192.168.100.112(192.168.100.112:3306) succeeded

Master 192.168.100.111(192.168.100.111:3306) is down!

Check MHA Manager logs at mha-mysql-slave2:/mha/logs/manager for details.

Started automated(non-interactive) failover.
Invalidated master IP address on 192.168.100.111(192.168.100.111:3306)
Selected 192.168.100.112(192.168.100.112:3306) as a new master.
192.168.100.112(192.168.100.112:3306): OK: Applying all logs succeeded.
192.168.100.112(192.168.100.112:3306): OK: Activated master IP address.
192.168.100.113(192.168.100.113:3306): OK: Slave started, replicating from 192.168.100.112(192.168.100.112:3306)
Master failover to 192.168.100.112(192.168.100.112:3306) completed successfully.
```

步骤4 登录从数据库slave1，查看虚拟IP漂移成功。

图 3-37 VIP 漂移成功

```
[root@mha-mysql-slave1 ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.112 netmask 255.255.255.0 broadcast 192.168.100.255
    inet6 fe80::f816:3eff:fe4d:915 prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:4d:09:15 txqueuelen 1000 (Ethernet)
    RX packets 500720 bytes 739812916 (705.5 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 32344 bytes 2830128 (2.6 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0:1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.99 netmask 255.255.255.0 broadcast 192.168.100.255
    ether fa:16:3e:4d:09:15 txqueuelen 1000 (Ethernet)

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 96 bytes 18912 (18.4 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 96 bytes 18912 (18.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

步骤5 查看MHA Manager运行状态，此时MHA Manager已经完成工作，处于停止状态。

mha_app1_status

图 3-38 MHA 完成故障转移

```
[root@mha-mysql-slave2 ~]# mha_app1_status
app1 is stopped(2:NOT_RUNNING).
[root@mha-mysql-slave2 ~]#
```

步骤6 登录收件人邮箱查看报警邮件。

图 3-39 报警邮件

app1: MySQL Master failover 192.168.100.111(192.168.100.111:3306) to 192.168.100.112(192.168.100.112:3306) succeeded

收件人

Master 192.168.100.111(192.168.100.111:3306) is down!

Check MHA Manager logs at mha-mysql-xyh-slave2:/mha/logs/manager for details.

Started automated(non-interactive) failover.
Invalidated master IP address on 192.168.100.111(192.168.100.111:3306)
Selected 192.168.100.112(192.168.100.112:3306) as a new master.
192.168.100.112(192.168.100.112:3306): OK: Applying all logs succeeded.
192.168.100.112(192.168.100.112:3306): OK: Activated master IP address.
192.168.100.113(192.168.100.113:3306): OK: Slave started, replicating from 192.168.100.112(192.168.100.112:3306)
192.168.100.112(192.168.100.112:3306): Resetting slave info succeeded.
Master failover to 192.168.100.112(192.168.100.112:3306) completed successfully.

----结束

故障恢复

步骤1 在主数据库上重新启动MySQL服务，将其作为从数据库加入集群（注意主数据库IP已更新）。

```
systemctl start mysqld.service
mysql -uroot
> CHANGE MASTER TO
  MASTER_HOST='192.168.100.112',
  MASTER_USER='repl',
  MASTER_PASSWORD='密码',
  MASTER_PORT=3306,
  MASTER_CONNECT_RETRY=10,
  MASTER_AUTO_POSITION=1;
> START SLAVE;
> SHOW SLAVE STATUS\G;
```

步骤2 修改MHA Manager配置文件，将旧主数据库加入集群。

```
vim /datadisk/mha/conf/app1.cnf
[server1]
candidate_master=1
check_repl_delay=0
hostname=192.168.100.111
port=3306
```

步骤3 重新启动MHA服务。

```
mha_app1_start
mha_app1_status
```

----结束

手动切换

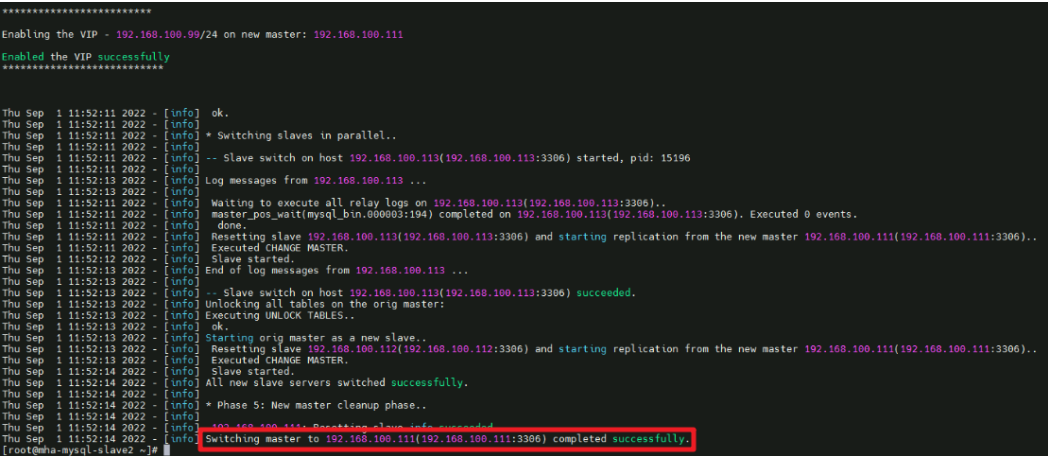
步骤1 手动切换主数据库，必须先停止MHA服务：

```
mha_app1_stop
```

步骤2 然后在MHA Manager上运行以下命令，进行手动在线切换主从数据库：

```
masterha_master_switch --conf=/datadisk/mha/conf/app1.cnf --master_state=alive --
new_master_host=192.168.100.111 --orig_master_is_new_slave --running_updates_limit=10000 --
interactive=0
出现-- Switching master to 192.168.0.111(192.168.0.111:3306) completed successfully. 代表切换成功
```

图 3-40 手动在线切换



步骤3 在新的主数据库（192.168.100.111）查看VIP漂移成功。
ifconfig

步骤4 在MHA Manager服务器上开启MHA服务即可。
mha_app1_start
mha_app1_status

----结束

3.4 快速卸载

一键卸载

步骤1 登录[应用编排服务AOS](#)资源栈，找到该解决方案创建的资源栈，单击资源栈名称最右侧“删除”按钮，在弹出的“删除资源栈”提示框输入“Delete”，单击“确定”进行解决方案卸载。

图 3-41 删除资源栈



----结束

4 附录

名词解释

基本概念、云服务简介、专有名词解释

- 弹性云服务器ECS：是一种可随时自助获取、可弹性伸缩的云服务器，可帮助您打造可靠、安全、灵活、高效的应用环境，确保服务持久稳定运行，提升运维效率。
- 云服务器组：通过[云服务器组](#)功能，弹性云服务器在创建时，将尽量分散地创建在不同的主机上，提高业务的可靠性。支持反亲和性策略，即同一云服务器组中的弹性云服务器分散地创建在不同的主机上，提高业务的可靠性。
- 弹性公网EIP：提供独立的公网IP资源，包括公网IP地址与公网出口带宽服务。可以与弹性云服务器、裸金属服务器、虚拟VIP、弹性负载均衡、NAT网关等资源灵活地绑定及解绑。
- MHA：MHA（Master High Availability）目前在 MySQL 高可用方面是一个比较成熟的解决方案，它是由日本 DeNA 公司 Yoshinori Matsunobu发的，是一套优秀的动作集 MySQL Failover 和高可用环境下的主从提升的高可用软件。在 MySQL 故障转移期间，MHA 可以在 0~30 秒内自动完成数据库故障转移操作，也在故障转移过程中，MHA 最大程度地保证数据的一致性，实现真正的高可用。MHA 中有两个字符之一是 MHA Node（数据节点），另一个是 MHA Manager（管理节点）。MHA Manager 可以单独部署在单机上管理多个主从集群，也可以部署在一个从节点上。MHA Node 运行在每个 MySQL Server 上，MHA Manager 会检测到 master 节点，当 master 失败的时候，它会自动更新 slave 的数据升级到一个新的 master，然后把所有其他 slave 点放回新的 master。整个故障转移过程对于应用程序来说是完全透明的。

5 修订记录

发布日期	修订记录
2022-08-30	第一次正式发布。
2023-02-28	修订实施步骤。