

# ServHA使用手册

## 前提

### 环境要求

- 两台服务器操作系统一致，每台服务器至少两个网口，在同一个网段下，至少用网线直连一对网口，另一个网口都接交换机用于对外服务。

注意：服务器型号、硬件条件可以不同。

网卡必须设置固定IP。

直连网卡设置与应用对外服务网卡处于不同的网段，只配置IP与子网掩码。

- 需要在服务网络中多预留一个IP地址（用于双机热备虚拟IP），该IP地址需要和两台服务器的实际IP在同一个网段下且不能绑定MAC，当双机热备部署完成后，用户通过此虚拟IP地址访问业务系统。
- 尽量不使用杀毒软件，切勿使用个人版的杀毒软件。
- 操作系统的root用户必须可用。
- 操作系统的以下命令必须可用，若不可用，请提前安装依赖包：

命令	依赖包
fuser	psmisc
ifconfig	net-tools

- 安装ServHA需要操作系统中有gcc、kernel-devel、kernel-header、flex组件，可在安装操作系统的时候安装上，若前期没有安装，则可以手动从操作系统的安装介质中安装，或者保留好安装介质，方便后面交给热备安装脚本自动安装。
- 两台服务器的hostname不能一样。
- 两台服务器均根据业务需求划分出一个或若干个独立的非操作系统分区（支持LVM）用于数据存储（比如/dev/sdb1），两台服务器用于数据存储的分区大小须一致（若无法做到一致，则备用服务器的数据分区可以略大），双机热备软件负责实时同步两台服务器该分区内所有的IO。注意：若该分区内有有用的数据需要保留，可在两台服务器上额外再划分出一个或若干个适当大小的分区（支持LVM）用于存放镜像元数据。

元数据分区大小 (MB) > 数据分区大小 (MB) / 32768 + 1

一般地，500MB的元数据分区支持15.5TB的数据分区

- ServHA在运行过程中需要开放以下端口：

14338、17968、17969、25792、45782（可自定义的镜像同步端口）、[7700]

其中17969、25792、45782（可自定义的镜像同步端口）为TCP通信端口，17968、14338为UDP通信端口，7700为监控端远程访问端口。

除需开放防火墙的端口外, 还需关闭SELinux等安全类组件。

## 应用安装要求

- 两台服务器业务系统安装必须一致, 包括但不限于安装路径、数据库实例路径、数据库实例名、数据库权限配置、业务系统附件存放路径等。
- 业务系统需要可以在两台服务器上分别独立正常运行, 若有绑定硬件的加密授权机制, 需要两台服务器均有正版授权。
- 业务系统不能绑定服务器MAC、IP地址、计算机名。
- 需要明确业务系统的变动数据, 比如数据库实例、文件附件之类的存放路径等需要同步的路径。

## 热备软件安装

热备软件和应用软件的安装顺序无要求

1. 将热备软件安装包存放到服务器系统的目录下 (例如/home) 。
2. 解压ServHA安装包

```
cd /home
tar -zxvf ServHA-Mirror.tar.gz
```

3. 进入解压的安装包目录并执行安装脚本

```
cd ServHA-Mirror_Setup/
./install.sh
```

选择安装卷镜像组件

```
感谢您选用ServHA Mirror双机镜像软件!
本脚本将自动安装ServHA Mirror程序及其相关组件,
[输入数字, 以选择要安装的镜像组件 Choose the mirroring component to install]
1) 全部安装 All
2) 仅卷镜像 Only volume mirroring component
3) 仅目录镜像 Only directory mirroring component
#?2
```

**\*如果提示缺少依赖组件, 则按y进行组件安装, 没有提示则直接跳过本步骤**

```
感谢您选用MicroColor ServHA Mirror双机镜像软件!
本脚本将自动安装MicroColor ServHA Mirror 程序及其相关组件,
正在检查您的系统环境...
请确保系统已经安装了 gcc, kernel-header, flex。您可以在linux的安装光盘中找到相关的安装包
检测到您没有安装gcc, 请先安装gcc。
检测到您没有安装kernel-devel, 请先安装kernel-devel。
检测到您没有安装flex, 请先安装flex。
安装前检查没有通过。
是否要执行insenv.sh来自动安装相关依赖组件 (y/n)?
```

此时将操作系统安装光盘放入光驱, 或者将操作系统安装iso文件拷贝到服务器上并且手动挂载到系统

中, 并按任意键继续:

```
#手动挂载iso文件示例:
```

```
mkdir /mnt/cdrom
```

```
mount -o loop ISO文件绝对路径 /mnt/cdrom
```

本脚本将自动安装 gcc, kernel-header, flex 组件, 请确保插入了当前操作系统对应的操作系统光盘。

COPYRIGHT 北京微彩华创科技有限公司

按任意键继续...

如果出现下图的提示:

```
按任意键继续... 尝试挂载光盘..
```

```
rmdir: 删除 "/mchatmpdisk"失败: 没有那个文件或目录
```

```
mount: 在 /dev/sr0 上找不到媒体
```

```
无法自动挂载/dev/cdrom, 请手动挂载并指定系统盘根目录
```

```
请输入系统盘根目录:
```

请检查Linux安装光盘是否正确的插入到了服务器上;

- 如果没有插入, 则在插入Linux安装光盘后按CTRL+C退出脚本并重新运行./install.sh脚本;
- 如果确认插入了光盘, 而脚本无法自动识别, 或者无法将光盘插入服务器, 则需手动输入Linux安装盘的根路径, 并按回车继续, 否则, 安装过程将无法继续。安装盘路径以实际情况为准;
- 如果无法自动安装依赖组件, 请按CTRL+C退出脚本, 并自行手动安装以下组件:

gcc	kernel-header	kernel-devel	flex
-----	---------------	--------------	------

安装方式及版本请以实际情况为准; 当所有组件安装完成后, 再次运行./install.sh脚本继续操作;

\*如果提示检测到系统开启了SELINUX, 如下图, 必须按y禁用SELINUX才能继续操作, 否则将无法继续安装, 没有提示则直接跳过本步骤。

```
检测到您的系统开启了SELINUX, 需要禁用SELINUX后才能继续
```

```
是否要禁用 (y/n)? y
```

```
已成功禁用SELINUX。
```

\*当脚本提示系统已满足安装条件时, 按任意键继续操作, 如下图

```
您的系统已满足安装条件, 按任意键开始安装...
```

\*当出现如下图提示时, 说明热备软件已经安装成功

```
=====
```

```
输入 'servha' 命令启动配置监控端开始配置热备系统。
```

```
恭喜你, ServHA安装完毕, 请尽情使用吧!
```

```
=====
```

## 应用软件安装

**\*重要! 两台服务器的应用软件安装与配置须完全一致。**

\*有安装包的应用软件使用安装包默认安装即可。

\*无安装包的绿色应用软件可按需注册成系统服务或者编写供root用户调用的启停脚本。

\*应用软件若需要生成单独的用户或用户组（例如Oracle、mysql、postgresql等），须保证该用户和用户组在两台服务器上的ID一致，ID可在/etc/passwd和/etc/group文件中查看并更改。

\*由于Linux系统提供了灵活的磁盘挂载和软链接功能，绝大多数的应用软件的数据目录都可以通过这些功能来移动到数据盘中，部分小型的应用甚至可以整个安装到数据盘中。

## ORACLE

可否全部安装到数据盘中	不可
独立用户和组	oracle:oinstall
需同步目录	/安装路径/app/oracle/oradata /安装路径/app/oracle/flash_recovery_area 或 /安装路径/app/oracle/fast_recovery_area
同步目录可否软链接	可
同步目录可否挂载盘	不推荐
同步目录可否配置	不可
启动方式	脚本
停止方式	脚本
检测方式	脚本

\*Linux下的Oracle可编辑第三方脚本用来启动、停止和检测。

\*用户可按需安装非RAC模式的oracle程序及实例，按需配置监听器，只需确保所有服务器与oracle相关的参数完全一致即可。

**\*建议将所有的表空间都建到/安装路径/app/oracle/oradata目录下，以免出现不必要的问题！**

## 其他、自研应用

将一个应用配置到热备中的主要思路是：

1. 告诉热备怎么启动应用（由于热备软件是运行在root用户下的程序，所以需要提供root用户能调用的启动方式）
2. 告诉热备怎么停止应用（由于热备软件是运行在root用户下的程序，所以需要提供root用户能调用的停止方式）
3. 告诉热备怎样判断应用是否正常运行，这个会在下面的章节章节详细描述（《本手册》【通用应用配置】）

热备软件每隔一段时间会去判断应用是否运行，并根据策略自动调用该应用的启动/停止方式，并不会关心这个应用具体是做什么业务的。



# 热备配置

## 配置前准备

### 网络

#### 配置服务网络（公网）IP

将两台服务器连入服务网络的网卡分别配上固定（静态）IP，此IP又称为“实际IP”，可按需配置双网卡绑定来提高网络链路的可用性。

注意：

实际IP不能和网络中其他设备的IP冲突

实际IP虽然也能对外提供业务访问能力，但在切换后，客户端需要通过备机的IP来访问业务服务，所以请在网络条件允许的情况下配置虚拟IP来对外提供服务

#### 配置直连网络（私网）IP

将两台服务器上相互直连的网卡分别配上固定（静态）IP，此IP又称为“私网IP”，可按需配置多组直连网络链路，也可按需配置双网卡绑定来提高网络链路的可用性。

注意：

私网IP与公网的实际IP需要做网段上的隔离

多组直连网络链路的情况下，组与组之间的私网IP也需要做网段上的隔离

私网IP仅需配置固定（静态）IP和子网掩码，无需配置网关和DNS

### 明确参考IP

参考IP为当前服务网络中一个长期稳定存在的IP地址，一般为网关的IP地址，**不能是做热备的两台服务器上配置的任何IP，也不能是配置的虚拟IP。**

参考IP的作用是当两台服务器之间失联的时候，哪台能连通（ping）这个参考IP，谁就启动业务资源树。

### 预留虚拟IP

虚拟IP又称为“漂移IP”，顾名思义，这个IP会随着业务资源树的切换而在两台服务器的公网网卡上漂移，客户端无需关心当前业务服务是运行在哪台服务器上，因为通过虚拟IP总能访问到当前对外提供服务的服务器上。

虚拟IP需跟对应的公网实际IP同网段，不同网段虽然可以配置上，但不一定能生效

虚拟IP不能是两台服务器的上配置的任何IP，也不能和网络中其他设备的IP冲突

虚拟IP的实现方式并不是通过热备软件中转网络请求，而是给公网网卡上多设置了一个IP地址（操作系统支持），所以虚拟IP在其中一台

服务器上生效的情况下，即使退出了热备的主程序，也不会影响客户端通过虚拟IP访问业务。

## 防火墙

操作系统防火墙必须开启前文【前提-环境要求】中所提端口对应协议的入站规则，此外，7700端口为配置监控端的管理端口，若需要进行远程配置，请开放此端口的TCP协议或HTTP协议的入站规则。

在安全条件允许的情况下也可以关闭并禁用防火墙服务。

Linux下常见的防火墙有以下几种：

- **firewalld**：常见于CentOS（7及以上）、RedHat（7及以上）、麒麟服务器操作系统等RedHat（7及以上）内核的linux操作系统

```
#关闭/禁用命令
systemctl stop firewalld
systemctl disable firewalld

#单独添加端口
firewall-cmd --add-port=端口号/[tcp|udp] --permanent
firewall-cmd --reload
```

- **ufw**：常见于Ubuntu、麒麟桌面操作系统等Debian内核的linux操作系统

```
#关闭/禁用命令
ufw disable

#单独添加端口
ufw allow 端口号/[tcp|udp]
```

- **iptables**：常见于CentOS（6及以下）、RedHat（6及以下）等RedHat（6及以下）内核的linux操作系统

```
#关闭/禁用命令
service iptables stop
chkconfig iptables off

#单独添加端口
iptables -A INPUT -p [tcp|udp] --dport 端口号 -j ACCEPT
cp /etc/sysconfig/iptables /etc/sysconfig/iptables.bak
iptables-save > /etc/sysconfig/iptables
```

## 其他常见安全组件

## SELinux

由于SELinux组件会阻止热备程序调用其他程序的脚本，造成热备管理应用程序出现异常，所以必须将SELinux组件禁用。

相关命令：

```
#查看SELinux状态
/usr/sbin/sestatus -v

#临时关闭
setenforce 0

#彻底关闭
#修改/etc/selinux/config文件
#将SELINUX=enforcing改为SELINUX=disabled
#保存
```

## Apparmor

\*Apparmor组件一般是Ubuntu系统默认自带的安全组件

Apparmor组件可能会导致某些应用程序无法使用非安装目录下的文件，常见于更改了数据位置的mysql数据库，且停止该组件的运行也无法阻止其配置的生效，应对策略就是在更改完应用的某些文件位置后去修改Apparmor的配置文件，让其对指定位置的文件访问放行。

以mysql为例：

修改/etc/apparmor.d/usr.sbin.mysqld文件

找到下列几行：

```
# Allow data dir access
/var/lib/mysql/ r,
/var/lib/mysql/** rwk,
```

在这几行后面加入：

```
/新路径/ r,
/新路径/** rwk,
```

保存后重启Apparmor服务

```
service apparmor restart
```

## KYSEC

KYSEC组件常见于麒麟操作系统，该组件常常会阻止很多操作，所以需要将KYSEC设置为软模式：

```
sudo setstatus softmode -p
```

## 镜像卷

### 镜像盘初始化

建立镜像前，作为主端的镜像盘大小必须小于等于从端镜像盘的大小；

两台服务器上用作镜像盘的分区均不能有挂载点和默认挂载点，可以使用umount命令取消挂载，和修改/etc/fstab文件来取消自动挂载。

**\*镜像组件需要一定的磁盘空间来存储镜像运行元数据，如果选择将元数据放在镜像盘中将会清空镜像盘内现有的数据，若想保留这些数据，可以选择下面两种方案的其中一种：**

- 将元数据存放在其他分区（推荐），关于元数据分区的说明参见前文
- 先将镜像盘中现有数据考出，待镜像建立完毕后再将考出的数据考回

## 基础双机环境配置

### 登录配置监控端

ServHA提供两种管理方式：

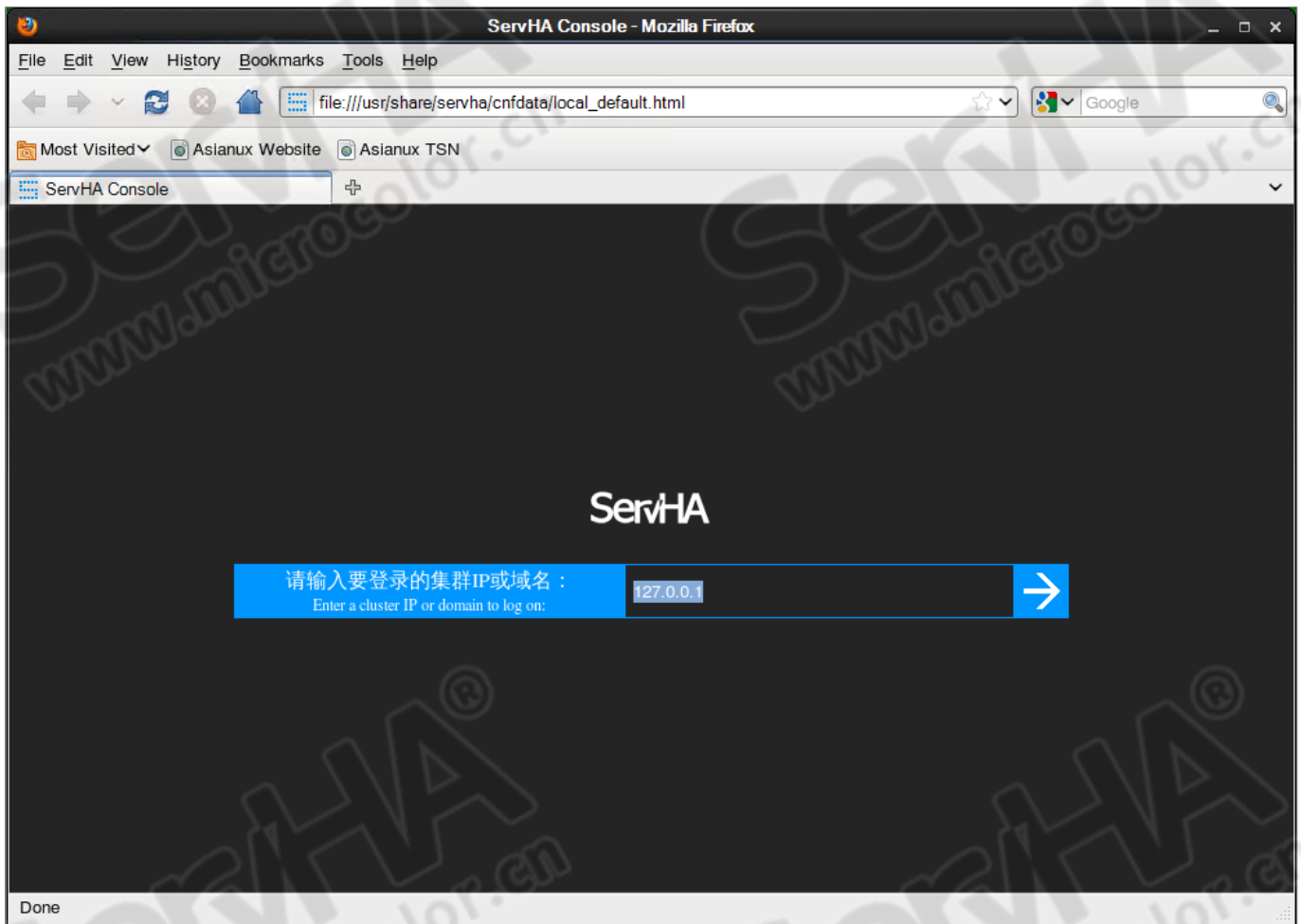
- 本地管理（Windows可以使用ServHA自带的配置监控端，Linux需要有GUI界面和支持HTML5的浏览器）
- 远程管理（服务器防火墙需要开放7700端口的访问，远程计算机上需要有支持HTML5的浏览器）

这两种方式是并存且可同时使用的。

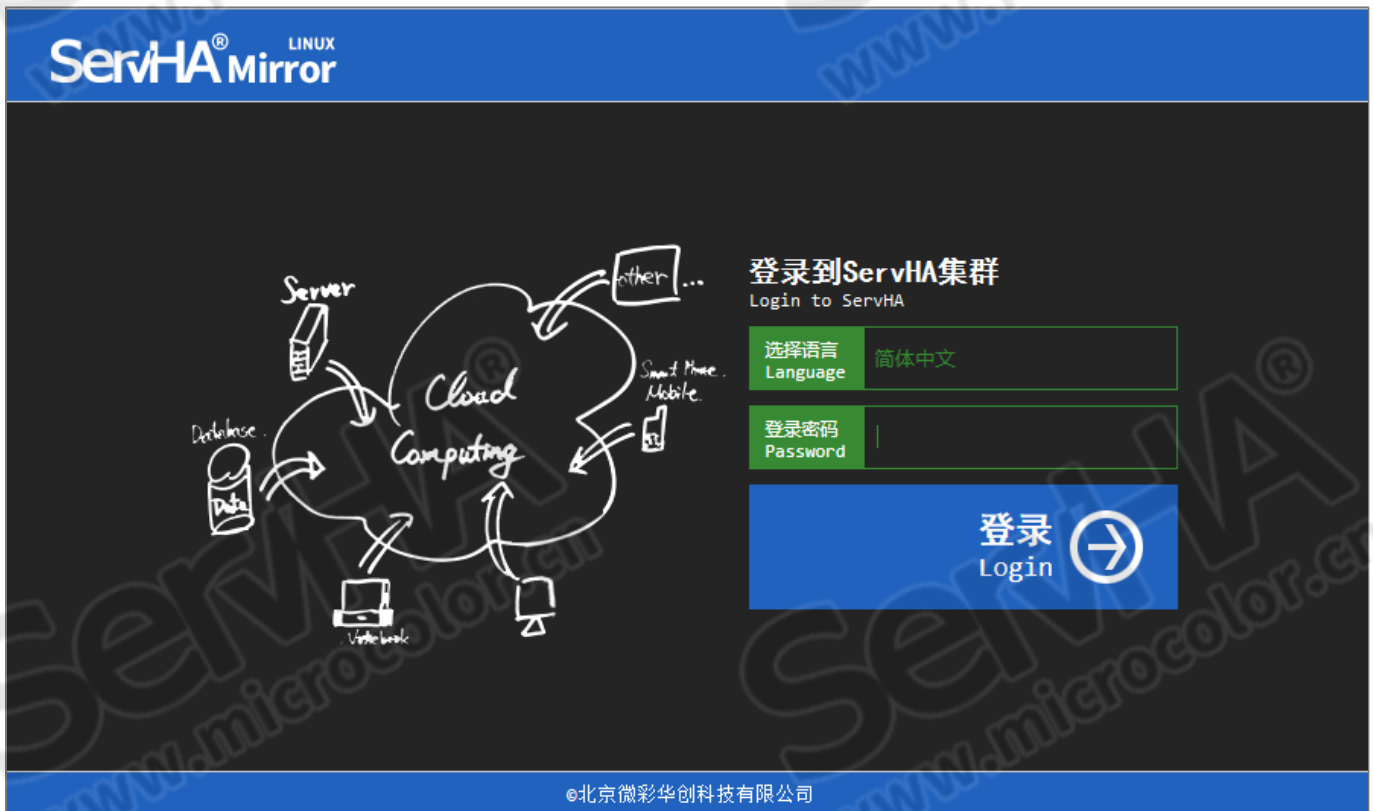
### 本地连接

在安装有ServHA的服务器上运行命令“servha”，系统将会自动使用默认浏览器打开ServHA配置工具。

打开配置工具后，输入需要连入的双机热备中其中一个节点的IP地址，并点击箭头按钮；如下图：



如果输入的IP地址正确，则会到达登录界面，输入正确的密码（默认密码为“1”）后点击“登录”按钮即可连入ServHA，如下图：





## 远程连接

打开远程计算机的浏览器，在地址栏中输入双机热备中其中一个节点的IP地址+7700端口访问，例如：http://服务器IP:7700。

## 集群初始化

### 第一步：确认基本信息

主机名称为当前计算机名，可以修改（并不会改变计算机名）。如下图



初始化集群向导

第一步：确认本机信息

第二步：初始化集群信息

第三步：心跳配置

第四步：确认

第一步：确认本机信息

\*主机名称：

主机描述：

### 第二步：初始化集群信息

为双机集群填写一个名称，设置登录密码（不填写则为默认密码“1”）。

参考IP是指双机集群所处公网环境中的一个或多个一直在线节点的IP地址（例如路由器，交换机，网关的IP）。

参考IP可以添加多个，中间用半角逗号隔开。

参考IP不要填写两台主机的IP。

若网络中没有合适的参考IP时，参考IP可填写127.0.0.1。

在镜像双机热备方案中，选择不使用仲裁。

初始化集群向导

ServHA

第一步：确认本机信息

第二步：初始化集群信息

第三步：心跳配置

第四步：确认

\*集群名称：

HA

登录密码：（不填写为不修改，为默认）

确认密码：（不填写为不修改，为默认）

\*心跳UDP端口：

14338

\*心跳间隔（ms）：

2200

\*心跳丢失告警次数：

2

\*首次启动接管时间（ms）：

15000

\*心跳超时时长（ms）：

3000

\*心跳丢失宕机判定次数：

3

\*默认语言：

简体中文

\*参考IP：

192.168.1.1

仲裁策略：

不使用仲裁

仲裁盘：

不配置

上一步

下一步

取消

第三步：心跳配置

选择要用于搭建心跳的网卡，并为其填写一个名称，点击“添加心跳”。

已添加的多个心跳可以通过鼠标上下拖拽进行排序，排序越靠上心跳连接的优先级越高，建议私网链路心跳优先级高于公网链路心跳。

初始化集群向导

ServHA

第一步：确认本机信息

第二步：初始化集群信息

第三步：心跳配置

第四步：确认

第三步：心跳配置

设置心跳信息，并添加到下面的集合中

\*心跳名称：

本机对应网卡：

添加心跳

已添加心跳（拖动顺序，排位越上优先级越高）：

心跳名称：public  
IP：192.168.1.11  
网卡名称：eth1

上一步

下一步

取消

第四步：信息确认

此处为前三步信息的汇总，确认无误之后点击“完成”按钮完成集群初始化。

初始化集群向导

ServHA

第一步：确认本机信息

第二步：初始化集群信息

第三步：心跳配置

第四步：确认

第四步：信息确认

请确认以下信息均配置正确无误后点击完成按钮。

当前主机名：

Node01

当前主机描述：

集群名称：

HA

是否修改密码：

否

心跳UDP端口：

14338

心跳间隔：

2200 毫秒

心跳超时长：

3000 毫秒

告警心跳丢失次数：

2

宕机心跳丢失后判定次数：

3

首次启动接管延时：

15000 毫秒

仲裁策略：

不使用仲裁

仲裁盘：

不配置

默认语言：

简体中文

参考IP：

192.168.1.1

心跳连接（1）：

private 192.168.3.11 eth2

心跳连接（2）：

public 192.168.1.11 eth1

上一步

完成

取消

## 将第二台主机加入现有集群

### 第一步：点击加入

连入第二台主机，点击初始化操作中的右侧“+”图标，加入现有集群。如下图：



### 第二步：输入第一台主机的IP

在新页面的左侧输入框里输入第一台主机的IP（不是输入本机IP），点击确定，将跳转至目标集群。

### 将节点加入现有集群

请输入现有集群其中一个节点的IP:

确定

- 在左侧文本框中输入现有集群IP
- 点击确定按钮前往目标集群
- 登录目标集群
- 添加新节点向导会自动运行
- 设置节点信息
- 保存提交表单

第三步：跳转至目标第一台主机

点击确定之后，输入密码登录集群，然后将跳转至“添加新节点向导”，新节点的IP默认为之前填写的节点IP，如下图：

添加新节点向导

ServHA

第一步：连接新节点

第二步：新节点信息

第三步：新节点心跳配置

第四步：确认

第一步：连接新节点

新节点的IP :

192.168.1.12

下一步

取消



第四步：查看第二台主机信息

此处可以查看第二台主机的信息。

添加新节点向导

ServHA

第一步：连接新节点

第二步：新节点信息

第三步：新节点心跳配置

第四步：确认

第二步：新节点信息

新节点信息

\*主机名称：

Node02

主机描述：

集群信息

集群名称：

HA

心跳UDP端口：

14338

心跳间隔：

2200

心跳超时长：

3000

告警心跳丢失次数：

2

宕机心跳丢失后判定次数：

3

上一步

下一步

取消

第五步：配对心跳

在心跳配置界面，左侧列出了当前配置的心跳连接，右侧列出的是第二台主机的网卡及IP列表（如下图），心跳连接需要由一对对应的网卡组成配对，先点击左侧某项心跳连接，再点击右侧对应的新节点的网卡，即可配为一对心跳连接（如下图）。

添加新节点向导

ServHA

第一步：连接新节点

第二步：新节点信息

第三步：新节点心跳配置

第四步：确认

第三步：新节点心跳配置

请为左侧的每一个心跳连接在右侧配置相应的新节点网卡

集群心跳

请在这里依次选择每个心跳，并在右侧选择该心跳使用新节点的哪个网卡

心跳名称：private [192.168.3.11]

对应网卡：[请选择对应网卡]

心跳名称：public [192.168.1.11]

对应网卡：[请选择对应网卡]

新节点网卡列表

如果您更改了网卡设置，请点击[这里刷新信息](#)

eth2 [ 192.168.3.12 ]

eth1 [ 192.168.1.12 ]

添加新节点向导

ServHA

第一步：连接新节点

第二步：新节点信息

第三步：新节点心跳配置

第四步：确认

第三步：新节点心跳配置

请为左侧的每一个心跳连接在右侧配置相应的新节点网卡

集群心跳

请在这里依次选择每个心跳，并在右侧选择该心跳使用新节点的哪个网卡

心跳名称：private [192.168.3.11]

eth2 [ 192.168.3.12 ]

心跳名称：public [192.168.1.11]

对应网卡：[请选择对应网卡]

新节点网卡列表

如果您更改了网卡设置，请点击[这里刷新信息](#)

eth1 [ 192.168.1.12 ]

第六步：确认信息

此步骤列出第二台主机的信息，确认无误之后点击“完成”，完成节点加入的步骤。如下图：

添加新节点向导

ServHA

第一步：连接新节点

第二步：新节点信息

第三步：新节点心跳配置

第四步：确认

第四步：确认

您将要吧新节点加入到集群中

新节点IP：

192.168.1.12

集群名称：

HA

新节点主机名称：

Node02

新节点主机描述：

心跳连接：

心跳名称：private [192.168.3.11]

eth2 [ 192.168.3.12 ]

心跳名称：public [192.168.1.11]

eth1 [ 192.168.1.12 ]

集群信息

集群名称：

HA

心跳UDP端口：

14338

心跳间隔：

2200

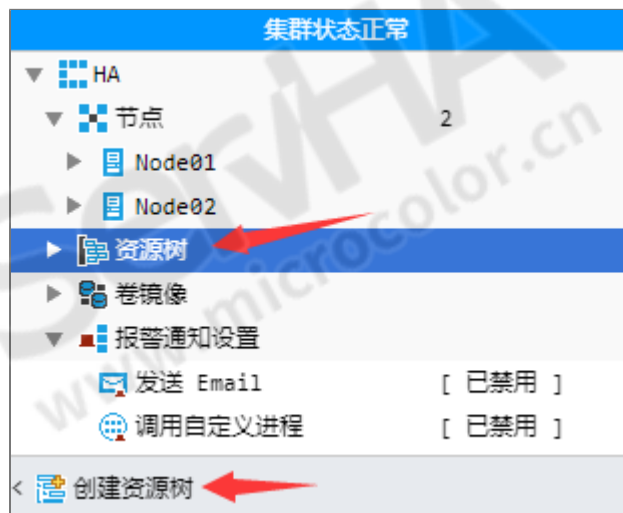
上一步

完成

取消

创建资源树

单击左侧资源树图标，点击下面的创建资源树按钮。



填写资源树名称，把两台服务器从未选主机拖拽到已选主机上，默认故障转移模式其他选项不用变更。配置完毕后如下图，点击保存，资源树就创建完毕了。

创建资源树 - MicroColor ServHA高可用软件综合管理平台

资源树属性

资源树名称:	ResTree1		
启动模式:	自动	资源树类型:	故障转移模式
切换策略:	按主机优先级顺序自动接管	切回策略:	由高优先级主机接管资源
停止失败处理方案:	等待管理员处理	启动失败处理方案:	转移到下一台主机

运行主机 (可拖动主机顺序调整优先级)  
注: 拖动下面的主机列表中的主机以选择运行主机及优先运行主机

已选主机, 位置越靠上的优先级越高

Node01

Node02

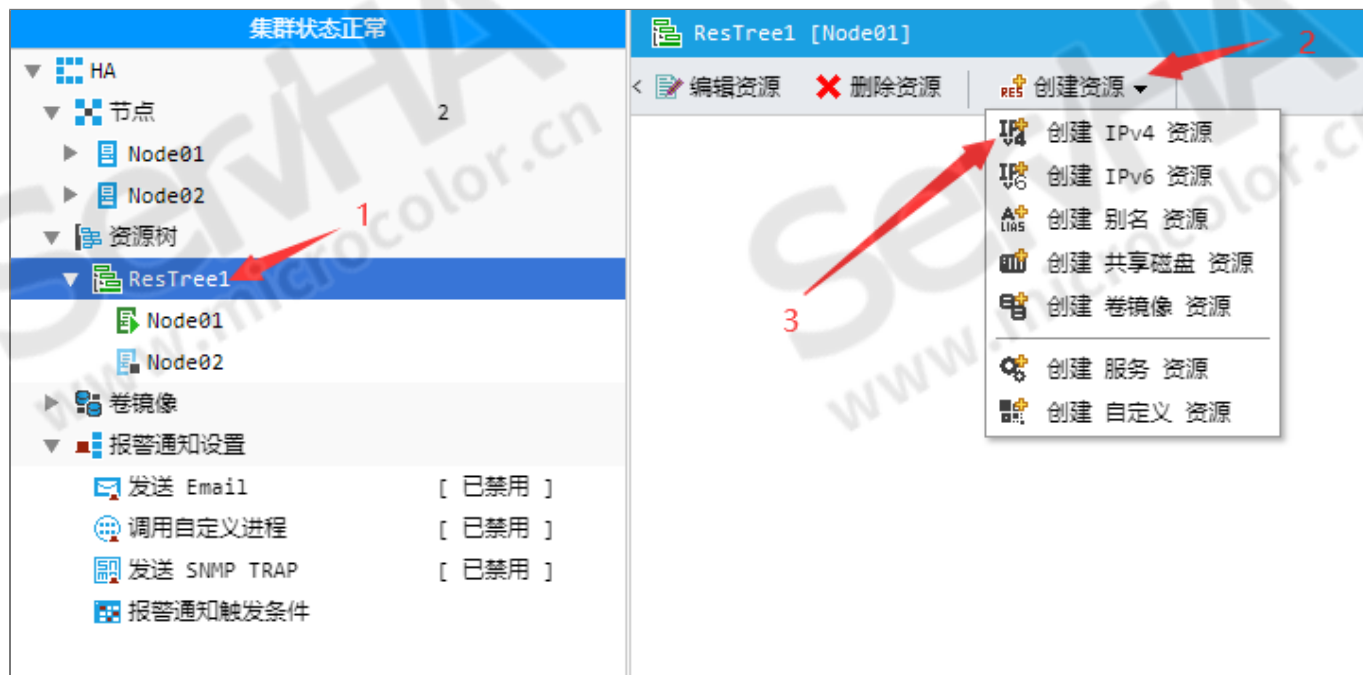
未选主机

保存

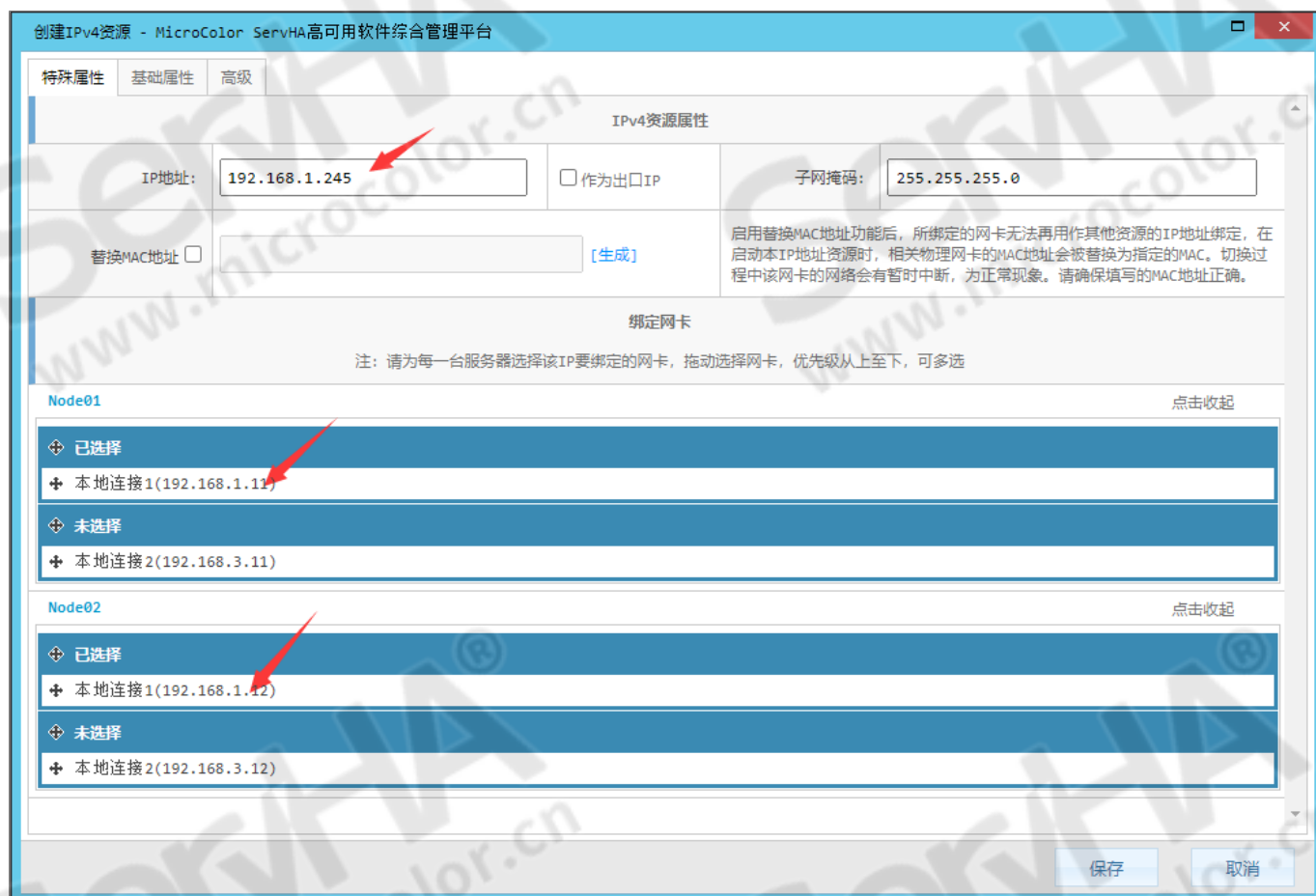
取消

## 创建IPv4资源 (虚拟IP)

先从左边树列表中选择要创建IP资源的资源树，切换右侧视窗后，点击创建资源，选择创建IPv4资源。



填写规划好的应用对外服务的IP地址（即虚拟IP），并分别将两机用作公网连接的网卡拖拽至已选择列表，如下图，配置完毕后点击保存。



## 创建卷镜像资源

首先需要创建一个卷镜像，卷镜像是一对服务器间磁盘卷镜像关系的管理单元。

点击选择左侧树列表的卷镜像条目，点击工具栏上的创建卷镜像按钮：



填写一个镜像名称，点击选择两台服务器，点击下一步：



选择作为输出（主）端的主机，选择两台主机中作为镜像盘的磁盘卷，点击下一步：





按需求选择脑裂处理策略，点击下一步。

脑裂是指，一对配置了镜像的服务器双方均在对方不在线（无法同步数据）的情况下分别接管过镜像资源，导致双方镜像盘内数据不一致并且无法自动合并数据，此时镜像双方均认为自身的数据是最新的，在这种情况下会触发镜像的脑裂，在镜像脑裂之后，镜像数据将暂停同步，直至脑裂恢复。脑裂恢复处理需要以两台服务器中的其中某一台的数据为准进行重新同步。

脑裂处理策略有以下几种方式：

- 手工处理：发生脑裂后，系统只发出错误报告，并等待人工处理，管理员需手动选择需要保留数据的一方。在脑裂期间，管理员仍然可以在服务器间转移资源树，方便确认哪台服务器数据为最新的版本，进行数据备份或手动合并数据。
- 自动丢弃较旧的一方：发现脑裂后，较早接管镜像资源的主机变更的数据将被丢弃，以较晚接管镜像资源主机的数据为准进行恢复同步
- 自动丢弃较新的一方：发现脑裂后，较晚接管镜像资源的主机变更的数据将被丢弃，以较早接管镜像资源主机的数据为准进行恢复同步
- 自动丢弃修改较少的一方：发现脑裂后，镜像盘中数据变动（修改）较少的主机变更的数据将被丢弃，以数据变动（修改）较多的主机的数据为准进行恢复同步
- 自动丢弃修改较多的一方：发现脑裂后，镜像盘中数据变动（修改）较多的主机变更的数据将被丢弃，以数据变动（修改）较少的主机的数据为准进行恢复同步

一般情况下，建议选择手工处理，避免因自动处理造成差异数据丢失。

第三步：协议类型 - MicroColor ServHA高可用软件综合管理平台

创建镜像向导

ServHA

第一步：基础信息

第二步：镜像盘信息

第三步：协议类型

第四步：元数据位置

第五步：同步链路

第三步：选择镜像协议

数据同步协议：

同步

脑裂处理策略：

手工处理

上一步

下一步

取消

根据规划设置元数据存放方式及位置，并点击下一步：

第四步：元数据位置 - MicroColor ServHA高可用软件综合管理平台

创建镜像向导

ServHA

第一步：基础信息

第二步：镜像盘信息

第三步：协议类型

第四步：元数据位置

第五步：同步链路

第四步：设置元数据存放位置

☒ 存放在镜像盘中

☐ 存放在其他磁盘

将元数据存放在镜像盘中，若所选的所有磁盘卷不满足以下条件，则需清除所选磁盘卷下所有数据（请先自行备份好有效数据）：

• 所选的磁盘卷曾做过镜像盘并成功建立了镜像

• 所选的磁盘卷上一次作为镜像盘时也将元数据存放在了镜像盘中

• 从上一次使用所选磁盘卷成功建立镜像后，从未直接格式化过此磁盘卷

☒ 清除所选磁盘卷中的数据

Node01

/dev/sdb1

5.99 GB

ext4

无挂载点

☒ 清除所选磁盘卷中的数据

Node02

/dev/sdb1

5.99 GB

ext4

无挂载点

上一步

下一步

取消

设置此卷镜像使用的端口号（该端口号不能与此集群上其他卷镜像使用的端口号冲突），选择数据同步链路，点击添加，添加到列表中（建议选择私网链路），点击完成：

第五步：同步链路 - MicroColor ServHA高可用软件综合管理平台

创建镜像向导

ServHA

第一步：基础信息

第二步：镜像盘信息

第三步：协议类型

第四步：元数据位置

第五步：同步链路

### 第五步：配置同步链路

使用端口号： 不同镜像必须使用不同的端口

选取双机能互通的IP地址，并点击添加按钮将其添加到列表中

Node01	Node02	限速 (0为不限)
<input type="text" value="192.168.1.11"/>	<input type="text" value="192.168.1.12"/>	<input type="text" value="0"/> MB/s <input type="button" value="添加"/>

已选链路

192.168.3.11	192.168.3.12	不限	<input type="button" value="X"/>
--------------	--------------	----	----------------------------------

确认配置无误后即可开始创建卷镜像：

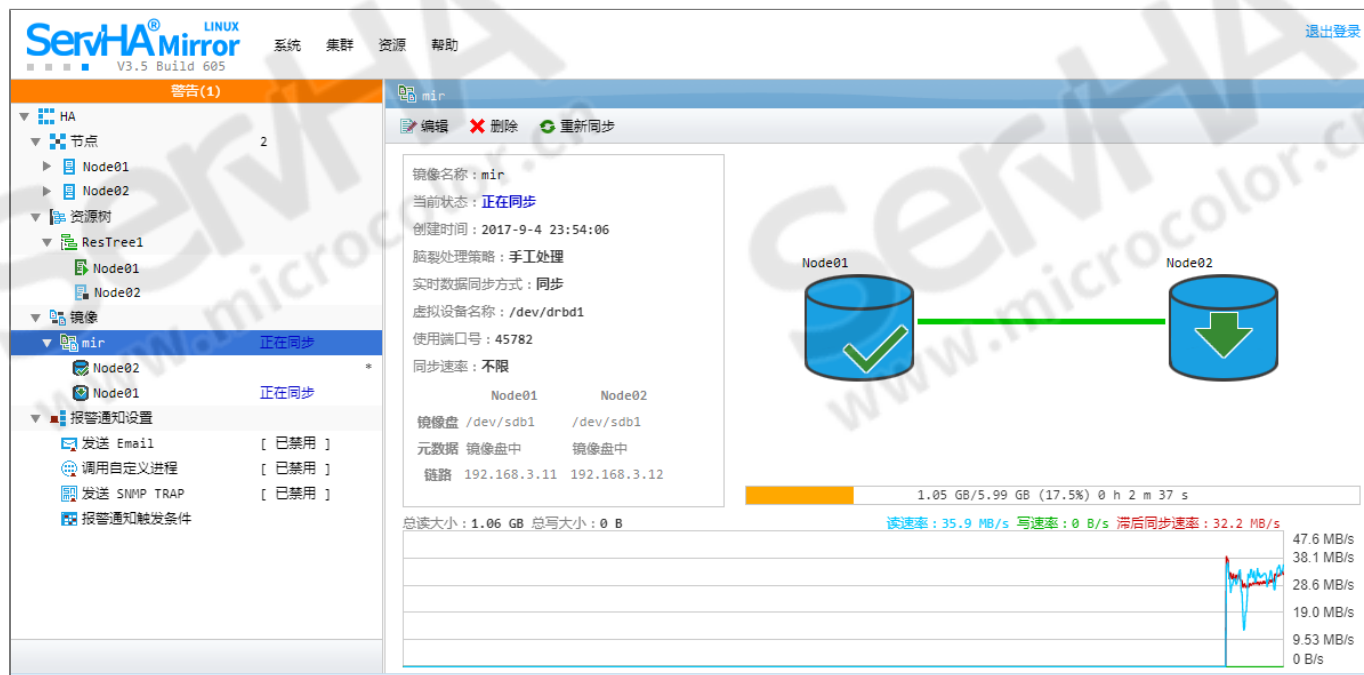
汇总信息

?

请校验以下信息：

镜像名称	mir
Node01	/dev/sdb1(输出端)
Node02	/dev/sdb1*将被输出端覆盖*
脑裂恢复策略	手工处理
同步协议	同步
使用端口号：	45782
元数据存放位置：	存放在镜像盘中
Node01	/dev/sdb1
Node02	/dev/sdb1

创建完毕后如下图：



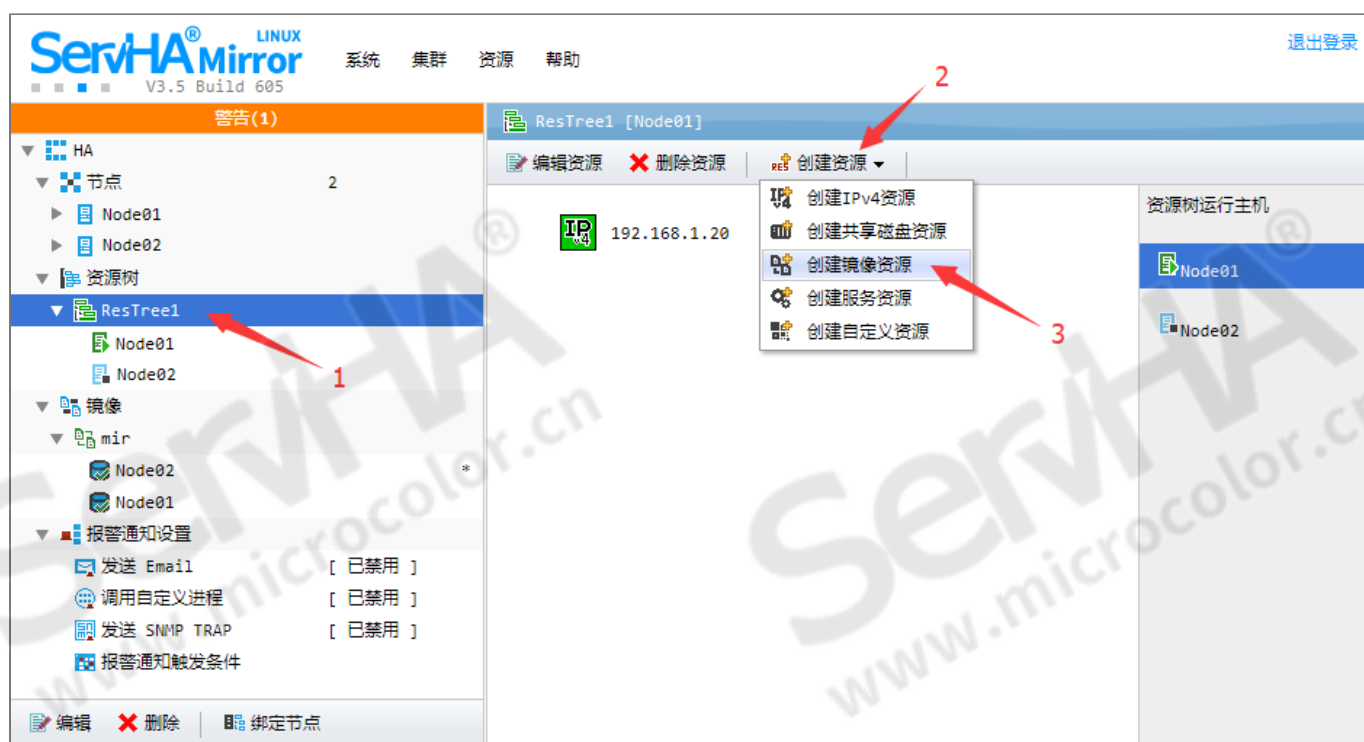
镜像创建完成后，在满足下列其中一个或多个条件的情况下需要进行格式化操作：

- 在建立镜像之前，主端镜像盘内没有文件系统
- 建立镜像的时候选择将元数据存放在镜像盘中并且选择了清除数据

格式化操作是针对镜像生成的虚拟设备进行的，在镜像界面可以看到虚拟设备名称，本例为 /dev/drbd1，可以在镜像主端使用命令格式化这个设备，例如：“mkfs.ext4 /dev/drbd1”。注：在格式化完成之前不能进行切换操作。

接下来，需要将这个镜像加入相应的资源树。

先从左边树列表中选择要创建镜像资源的资源树，切换右侧视窗后，点击创建资源，选择创建镜像资源

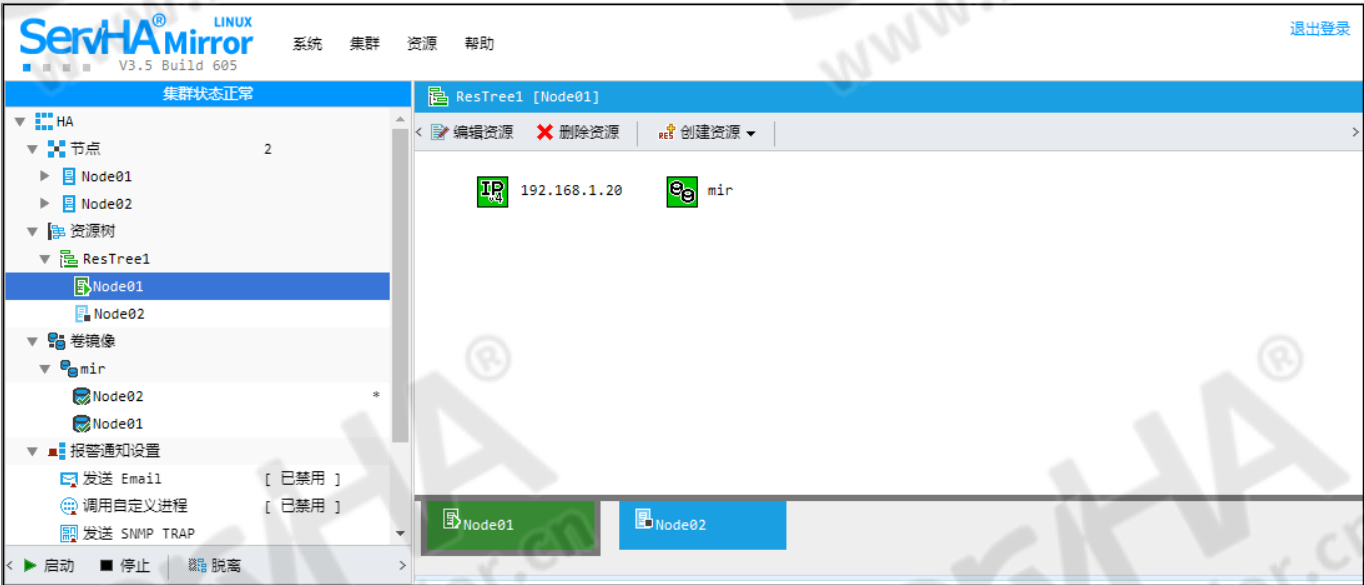




在创建镜像资源界面，选择刚才创建的镜像，并且填入挂载点（以“/”开头），点击保存。



创建完毕后如下图：

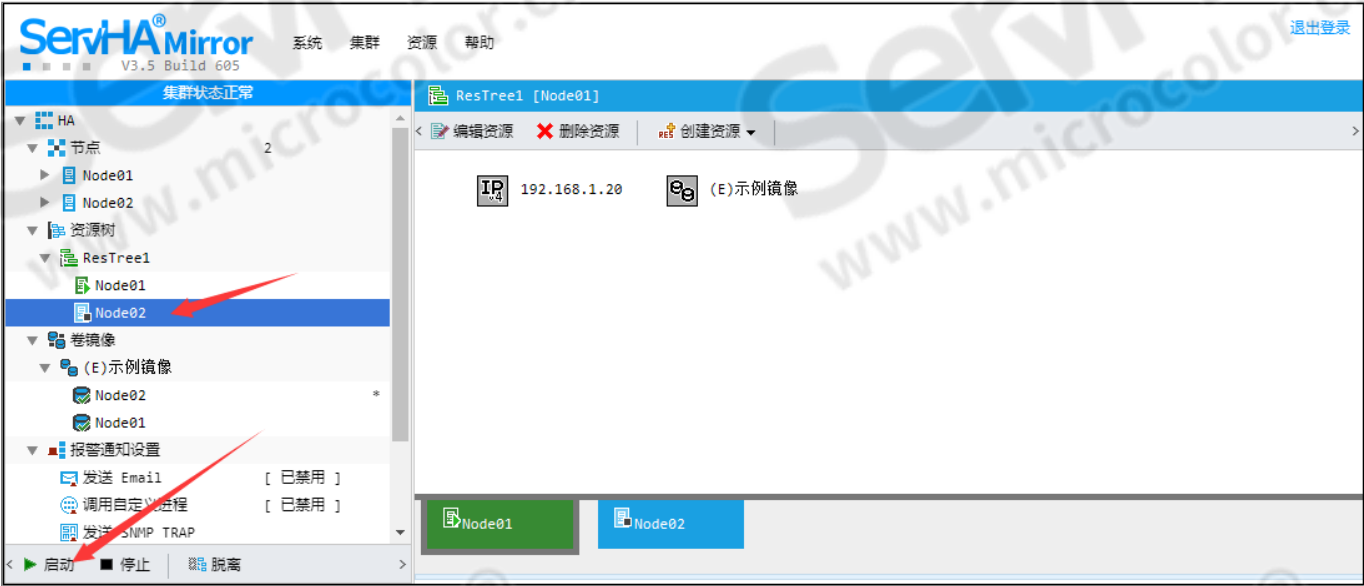


注意：当镜像卷还处于首次同步差异数据的过程中时，资源树备机节点会提示无法接管，待卷镜像状态从“正在同步”变为“正常”状态时才能正常切换。

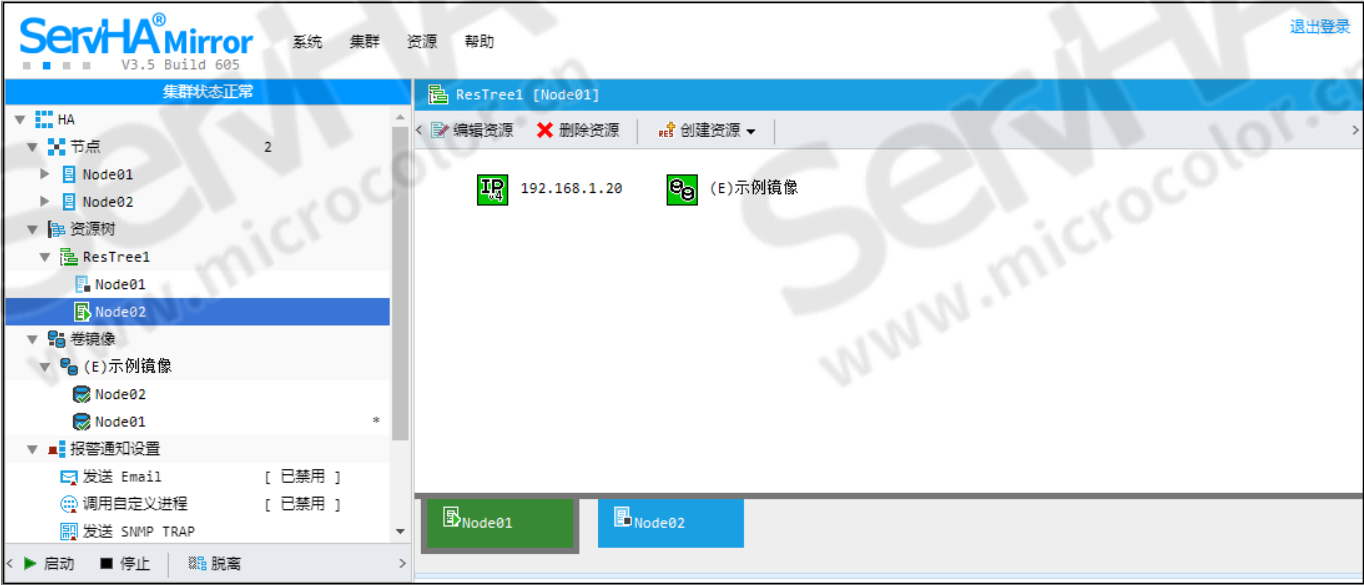
## 切换测试

首次建立镜像资源后，可以操作资源树在每台节点中进行一次切换，确认其切换正常。

单击左侧要切换到的节点，点击启动按钮



等待切换完毕。



## 应用配置

### ORACLE

#### 参数查看

将Oracle应用配置成高可用模式需要记录下列参数，您可以自行整理这些参数，也可以参考以下实例来查看这些参数：

## ORACLE\_BASE

示例:

```
su - oracle -c 'echo $ORACLE_BASE'
```

```
[root@linux ~]# su - oracle -c 'echo $ORACLE_BASE'  
/opt/app/oracle
```

## ORACLE\_HOME

示例:

```
su - oracle -c 'echo $ORACLE_HOME'
```

```
[root@linux ~]# su - oracle -c 'echo $ORACLE_HOME'  
/opt/app/oracle/product/11.2.0/db_1
```

## ORACLE\_SID

示例:

```
su - oracle -c 'echo $ORACLE_SID'
```

```
[root@linux ~]# su - oracle -c 'echo $ORACLE_SID'  
orcl
```

## 监听器配置文件位置

示例:

```
su - oracle -c 'echo $TNS_ADMIN'
```

```
[root@linux ~]# su - oracle -c 'echo $TNS_ADMIN'  
/opt/app/oracle/product/11.2.0/db_1/network/admin
```

监听器配置文件名称一般为“listener.ora”

## 监听器名称

示例:

```
su - oracle -c 'cat $TNS_ADMIN/listener.ora'
```

```
[root@linux ~]# su - oracle -c 'cat $TNS_ADMIN/listener.ora'  
# listener.ora Network Configuration File: /opt/app/oracle/product/11.2.0/db_1/network/admin/listener.ora
```

```
# Generated by Oracle configuration tools.

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
      (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
    )
  )

ADR_BASE_LISTENER = /opt/app/oracle
```

如上图中第一个等号“=”前的名称为监听器名称

## Oracle实例安装位置

示例:

```
su - oracle

sqlplus / as sysdba

select t1.name,t2.name from v$tablespace t1,v$datafile t2 where t1.ts# = t2.ts#;
```

```
[root@linux ~]# su - oracle
[oracle@linux ~]# sqlplus / as sysdba

SQL*Plus: Release 11.2.0.1.0 Production on Tue Nov 28 15:43:50 2017

Copyright (c) 1982, 2009, Oracle. Allrights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Oartitioning, Oracle Label Security, OLAP, Data Mining,
Oracle Database Vault and Real Application Testing options

SQL> select t1.name,t2.name from v$tablespace t1,v$datafile t2 where t1.ts# = t2.ts#;

NAME
-----
NAME
-----
SYSTEM
/opt/app/oracle/oradata/orcl/system01.dbf

SYSAUX
/opt/app/oracle/oradata/orcl/sysaux01.dbf

UNDOTBS1
/opt/app/oracle/oradata/orcl/undotbs01.dbf

NAME
-----
NAME
-----
USERS
/opt/app/oracle/oradata/orcl/users01.dbf

SQL>
```

如上图中实例名目录所处的路径为实例安装位置。

## 控制文件位置

示例:

```
cd /opt/app/oracle #进入ORACLE_BASE  
find -name "c*.ctl" #一般控制文件名为control**.ctl
```

```
[root@linux ~]# cd /opt/app/oracle  
[root@linux oracle]# find -name "c*.ctl"  
.flash_recovery_area/orcl/control02.ctl  
./oradata/orcl/control01.ctl
```

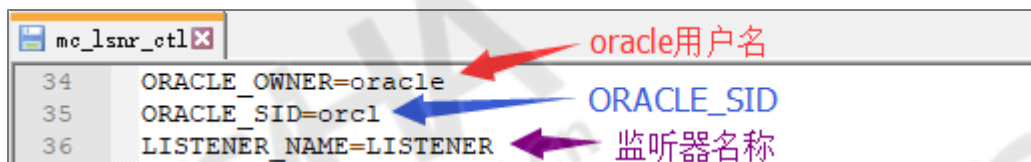
## 配置脚本

### 获取Oracle脚本

从安装介质或互联网上获取mc\_lsnr\_ctl脚本和mc\_orcl\_ctl脚本，备用。

### 修改mc\_lsnr\_ctl脚本

\*使用支持/n换行的文本编辑器（如vscode、vim等，切勿使用windows自带的记事本）编辑mc\_lsnr\_ctl脚本：



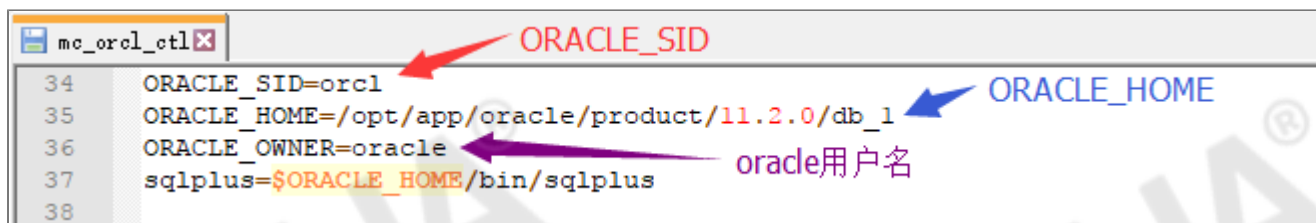
修改第34行ORACLE\_OWNER=后的值为oracle的用户名，一般为“oracle”；

修改第35行ORACLE\_SID=后的值为SID；

修改第36行LISTENER\_NAME=后的值为监听器的名称；

### 修改mc\_orcl\_ctl脚本

\*使用支持/n换行的文本编辑器（如vscode、vim等，切勿使用windows自带的记事本）编辑mc\_orcl\_ctl脚本：



修改第34行ORACLE\_SID=后的值为SID；

修改第35行ORACLE\_HOME=后的值为ORACLE\_HOME；

修改第36行ORACLE\_OWNER=后的值为oracle的用户名，一般为“oracle”；

## 部署脚本

将mc\_lsnr\_ctl脚本和mc\_orcl\_ctl脚本传输到所有服务器的非数据盘目录下（本例为“/home”），并赋予777权

限

示例:

```
chmod 777 /home/mc_*_ctl
```

```
[root@linux ~]# chmod 777 /home/mc_*_ctl
```

## 测试脚本mc\_lsnr\_ctl

示例:

```
/home/mc_lsnr_ctl status  
/home/mc_lsnr_ctl start  
/home/mc_lsnr_ctl status  
/home/mc_lsnr_ctl stop  
/home/mc_lsnr_ctl status
```

```
[root@linux ~]# /home/mc_lsnr_ctl status  
Listener is stopped  
[root@linux ~]# /home/mc_lsnr_ctl start  
-----  
The command completed successfully  
LSNRCTL> info Listener running:  
[root@linux ~]# /home/mc_lsnr_ctl status  
Listener is running  
[root@linux ~]# /home/mc_lsnr_ctl stop  
The command completed successfully  
[root@linux ~]# /home/mc_lsnr_ctl status  
Listener is stopped
```

mc\_lsnr\_ctl脚本支持三种参数: start、stop、status

status参数将返回该脚本中所指定oracle监听器的运行状态;

start参数可以将脚本中指定监听器启动;

stop参数可以将脚本中指定监听器停止;

测试需要将此三种参数均测试至少一遍, 确保其正常。

## 测试脚本mc\_orcl\_ctl

示例:

```
/home/mc_orcl_ctl status
```



```
/home/mc_orcl_ctl start
/home/mc_orcl_ctl status
/home/mc_orcl_ctl stop
/home/mc_orcl_ctl status
```

```
[root@linux ~]# /home/mc_orcl_ctl status
Oracle instance orcl is stopped
[root@linux ~]# /home/mc_orcl_ctl start
info Oracle instance orcl started:
[root@linux ~]# /home/mc_orcl_ctl status
Oracle instance orcl is running
[root@linux ~]# /home/mc_orcl_ctl stop
info Cleaning up for orcl
[root@linux ~]# /home/mc_orcl_ctl status
Oracle instance orcl is stopped
```

mc\_orcl\_ctl脚本支持三种参数：start、stop、status

status参数将返回该脚本中所指定oracle实例的运行状态；

start参数可以将脚本中指定实例启动；

stop参数可以将脚本中指定实例停止；

测试需要将此三种参数均测试至少一遍，确保其正常。

## 设置同步路径

**重要!!!** 此示例仅针对默认的实例路径，若在实例路径之外建立了表空间，则这些路径也要用相同办法建立软链接，但还是建议将所有表空间都建立到实例路径下

## 构建路径

**\*请先确保Oracle的所有实例和监听器在所有服务器上都是停止状态**

主机：将oracle实例目录和快速恢复区目录带权限拷贝到数据盘（这里假设数据盘挂载在/mir上）。

示例命令：

```
cp -rp /opt/app/oracle/oradata /mir
cp -rp /opt/app/oracle/flash_recovery_area /mir
```

主机与备机：将oracle安装目录下的实例目录和快速恢复区目录改名（后面加上\_old即可）：

示例命令：

```
mv /opt/app/oracle/oradata /opt/app/oracle/oradata_old
```

```
mv /opt/app/oracle/flash_recovery_area /opt/app/oracle/flash_recovery_area_old
```

## 建立软链接

主机与备机：将移入数据盘的目录软链接到oracle原目录，并给软链接赋权。

示例：

```
ln -s /mir/oradata /opt/app/oracle
ln -s /mir/flash_recovery_area /opt/app/oracle
```

```
[root@linux01 ~]# ln -s /mir/oradata /opt/app/oracle
[root@linux01 ~]# ln -s /mir/flash_recovery_area /opt/app/oracle
```

```
[root@linux02 ~]# ln -s /mir/oradata /opt/app/oracle
[root@linux02 ~]# ln -s /mir/flash_recovery_area /opt/app/oracle
```

\*即使/mir目录不存在也能建立软链接

\*记得给软链接赋权

```
chown -R oracle:oinstall /opt/app/oracle/oradata
chown -R oracle:oinstall /opt/app/oracle/flash_recovery_area
```

```
[root@linux ~]# chown -R oracle:oinstall /opt/app/oracle/oradata
[root@linux ~]# chown -R oracle:oinstall /opt/app/oracle/flash_recovery_area
```

## 修改监听器配置

该操作需在主备机上分别操作

使用vi、vim或gedit编辑监听器配置文件

示例：

```
vi /opt/app/oracle/product/11.2.0/db_1/network/admin/listener.ora
```

```
[root@linux ~]# vi /opt/app/oracle/product/11.2.0/db_1/network/admin/listener.ora
```

\*若无此文件，请建立，注意权限所有者为oracle:oinstall。

将内容按以下要求进行编辑

```
SID_LIST_LISTENER =
(
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = orcl) <ORACLE_SID>
      (ORACLE_HOME = /opt/app/oracle/product/11.2.0/db_1) <ORACLE_HOME>
      (SID_NAME = orcl) <ORACLE_SID>
    )
  )
)
LISTENER =
```

```
(DESCRIPTION_LIST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
    (ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.1.236)(PORT = 1521)) <虚拟IP
  )
)
ADR_BASE_LISTENER = /opt/app/oracle
```

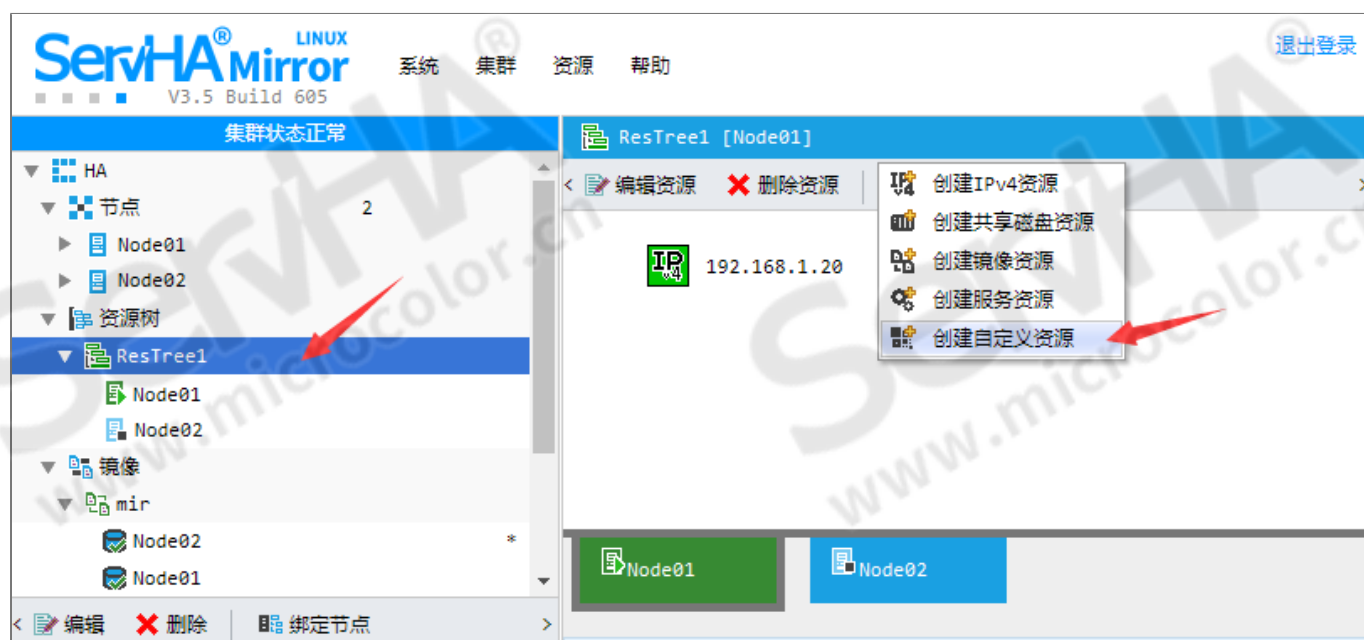
\*若监听器配置文件中没有“SID\_LIST\_LISTENER”、“LISTENER”等区块，请自行添加；

\*“SID\_LIST”区块下允许有多个“SID\_DESC”区块；

## 创建资源

### 创建监听器资源

选中资源树，创建自定义资源，如下图：



在弹出的“创建自定义资源”窗口中按下图所示填入参数，并点“保存”：

创建自定义资源 - MicroColor ServHA高可用软件综合管理平台

特殊属性 基础属性 高级

自定义资源属性

资源名称：	listener		
启动脚本：	/home/mc_lsnr_ctl	启动参数：	start
停止脚本：	/home/mc_lsnr_ctl	停止参数：	stop
测试脚本：	/home/mc_lsnr_ctl	测试参数：	status
是否尝试运行在桌面上：	<input type="checkbox"/>		

保存 取消

创建成功后资源树界面如下图：

ServHA® LINUX Mirror V3.5 Build 605 系统 集群 资源 帮助 退出登录

集群状态正常

HA

- 节点 2
  - Node01
  - Node02
- 资源树
  - ResTree1
    - Node01
    - Node02
  - 镜像
    - mir
      - Node02 \*
      - Node01

ResTree1 [Node01]

编辑资源 删除资源 创建资源

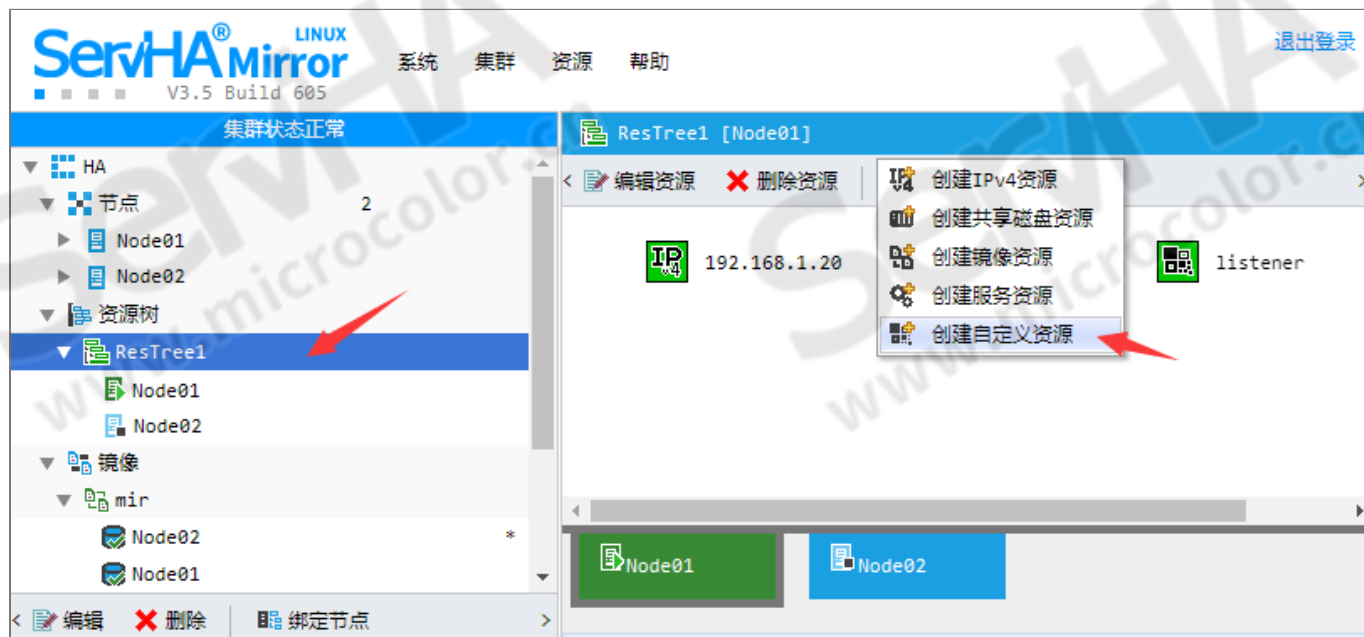
192.168.1.20 mir listener

Node01 Node02

编辑 删除 绑定节点

创建Oracle实例资源

选中资源树，创建自定义资源，如下图：



在弹出的“创建自定义资源”窗口中按下图所示填入参数，并点“保存”：

创建自定义资源 - MicroColor ServHA高可用软件综合管理平台

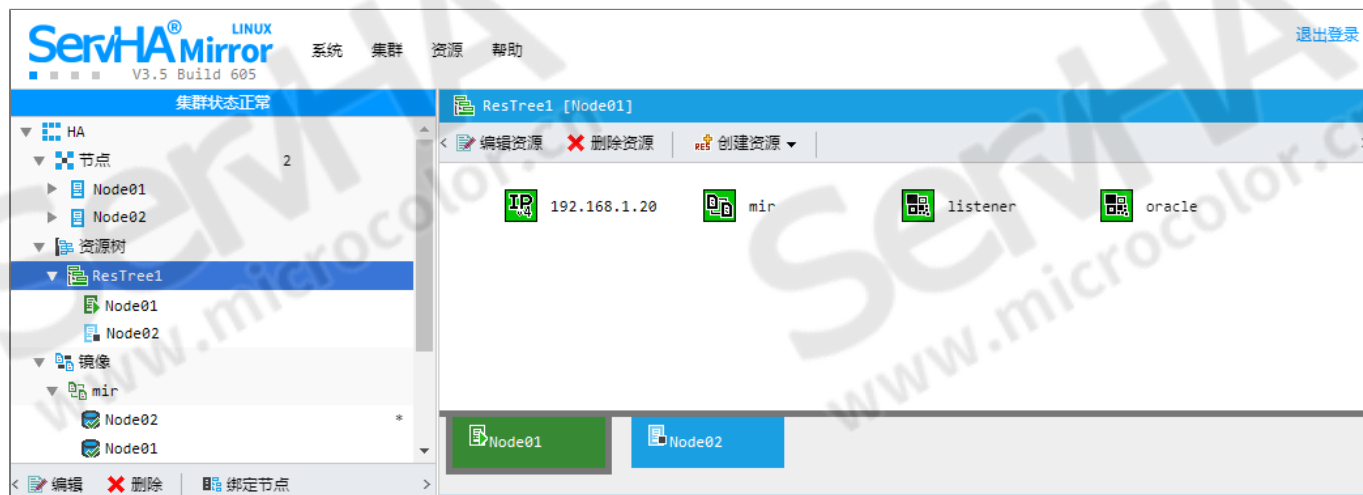
特殊属性 基础属性 高级

自定义资源属性

资源名称：	oracle		
启动脚本：	/home/mc_orcl_ctl	启动参数：	start
停止脚本：	/home/mc_orcl_ctl	停止参数：	stop
测试脚本：	/home/mc_orcl_ctl	测试参数：	status
是否尝试运行在桌面上：	<input type="checkbox"/>		

保存 取消

创建成功后资源树界面如下图：



## 推拽依赖关系

将监听器资源分别拖拽到IP资源和存储资源之下，并将Oracle实例资源拖拽到监听器资源之下，如下图：



## 通用应用配置

### 安装应用程序

ServHA对应用的安装只有一点要求：主备机上安装的应用要完全一致，即应用程序的版本、安装路径、配置等等参数要完全一致，也可将应用程序完全放入数据盘中。

请按照实际需求以合适的方式为两台服务器都安装适合的应用程序。

某些需要JDK或Mono等环境的应用程序，请先确保两台服务器上已配置好环境。

某些应用需要桌面环境，请自行按需安装。



## 处理应用同步路径

首先需要明确应用程序需要同步的路径，例如tomcat的默认路径为：

```
/usr/local/tomcat/webapps
```

- 如果应用自带配置文件可以修改此路径，则可以将此路径修改到数据盘下。
- 如果此应用是绿色软件，且可以在任意目录下都能正常运行，则可以将整个应用安置在数据盘中。
- 大部分的应用软件的同步路径都可以用软链接来将数据/文件转移到数据盘中。
- 若以上方式都不可用，则可以考虑将数据路径中的数据/文件移动到数据盘中，然后将数据盘直接挂载到同步路径上。

以tomcat使用软链接为例（数据盘挂载点为/mir）：

**\*请选确保所有服务器上的tomcat全部为停止状态**

```
#主机带权限移动同步目录
cp -rp /usr/local/tomcat/webapps /mir

#主机与备机给原同步目录改名
mv /usr/local/tomcat/webapps /usr/local/tomcat/webapps_old

#主机与备机建立软链接
ln -s /mir/webapps /usr/local/tomcat/webapps
```

\*tomcat只需保持软链接的root用户权限即可，其他应用需要保持软链接权限与原同步目录的权限一致，设置命令格式如下：

```
#设置权限用户
chown -R 用户名[:用户组] 绝对路径

#设置权限级别
chmod -R 权限级别 绝对路径
```

**\*关于权限级别**

#使用ls -l命令查看目录/文件列表的时候，第一列就是该目录/文件的权限级别

```
[root@linux ~]# ls -l
drwxr-xr-x 所属用户 所属用户组 (其他属性) 文件/目录名
```

权限级别部分要拆成4部分来看

- 第1个字符为文件类型，d代表是目录，l代表是软链接，-代表是文件
- 第2~4个字符为所属用户对该文件拥有的权限
- 第5~7个字符为所属用户组对该文件拥有的权限

- 第8~10个字符为其他用户对该文件拥有的权限

每组权限的3个字符的意义：

- 第1个字符代表是否有读权限，r代表有，-代表没有
- 第2个字符代表是否有写权限，w代表有，-代表没有
- 第3个字符代表是否有执行权限，x代表有，-代表没有

一般的赋权命令会使用权限级别的3位数字掩码

- 第1位为所属用户权限
- 第2位为所属用户组权限
- 第3位为其他用户权限

$r=4, w=2, x=1$

- 若要rwx属性，则 $4+2+1=7$
- 若要rw-属性，则 $4+2=6$
- 若要r-x属性，则 $4+1=5$

示例：

```
chmod 755 绝对路径
```

#代表该目录/文件的所属用户拥有全权限，所属用户组的用户和其他用户有读和运行权限

## 明确应用程序启停机制

接入ServHA管理的应用程序资源，需要使用启动脚本和停止脚本进行启停操作，启停脚本需要在完成操作后退出，不会伴随业务主进程占用终端，例如tomcat自带的startup.sh和shutdown.sh。

\*若此应用程序带有图形窗口，则需操作系统拥有桌面环境（gnome、KDE等），并且在操作系统启动后有用户登录过桌面才能正常启动该应用，在桌面登录之前，此应用程序的资源会标记为“决断错误”，资源树对应服务器节点会标记为“无法启动”，并且日志会输出相关信息。

若启动脚本会一直占用终端的话，需要再单独写一个脚本，脚本内容为：

```
nohup '应用的启动脚本' >/dev/null 2>&1 &
```

**\*记得给这个脚本赋777权限**

## 明确应用程序的检测机制

接入ServHA管理的应用程序资源，除了需要提供应用程序的启停方式之外，还需要提供检测其是否正常运行方式，ServHA会在每个检测周期内（大约几秒）通过调用检测器来判定本机的应用程序是否运行，并根据运行状态触发相应的动作（重启资源或切换）。

资源在启动和停止的过程中不会调用检测器，其他的时段，两台服务器的检测器会被周期性的分别调用，启动着资源树的服务器需要检测到资源为启动状态，未启动资源树的服务器需要检测到资源为停止状态。

ServHA提供了四种检测器，可根据实际情况和需要启用并配置相应的参数：

端口检测器（适用于网络服务程序）

检测本机指定的端口是否被程序占用，还可以联合判断占用该端口的程序是不是指定的程序，满足条件则认为应用程序在本机处于运行状态；

URL检测器（适用于WEB程序）

系统会尝试访问指定的URL，并检测返回内容，还可以联合判断返回内容中是否包含指定的字符串，满足条件则认为应用程序在本机处于运行状态；

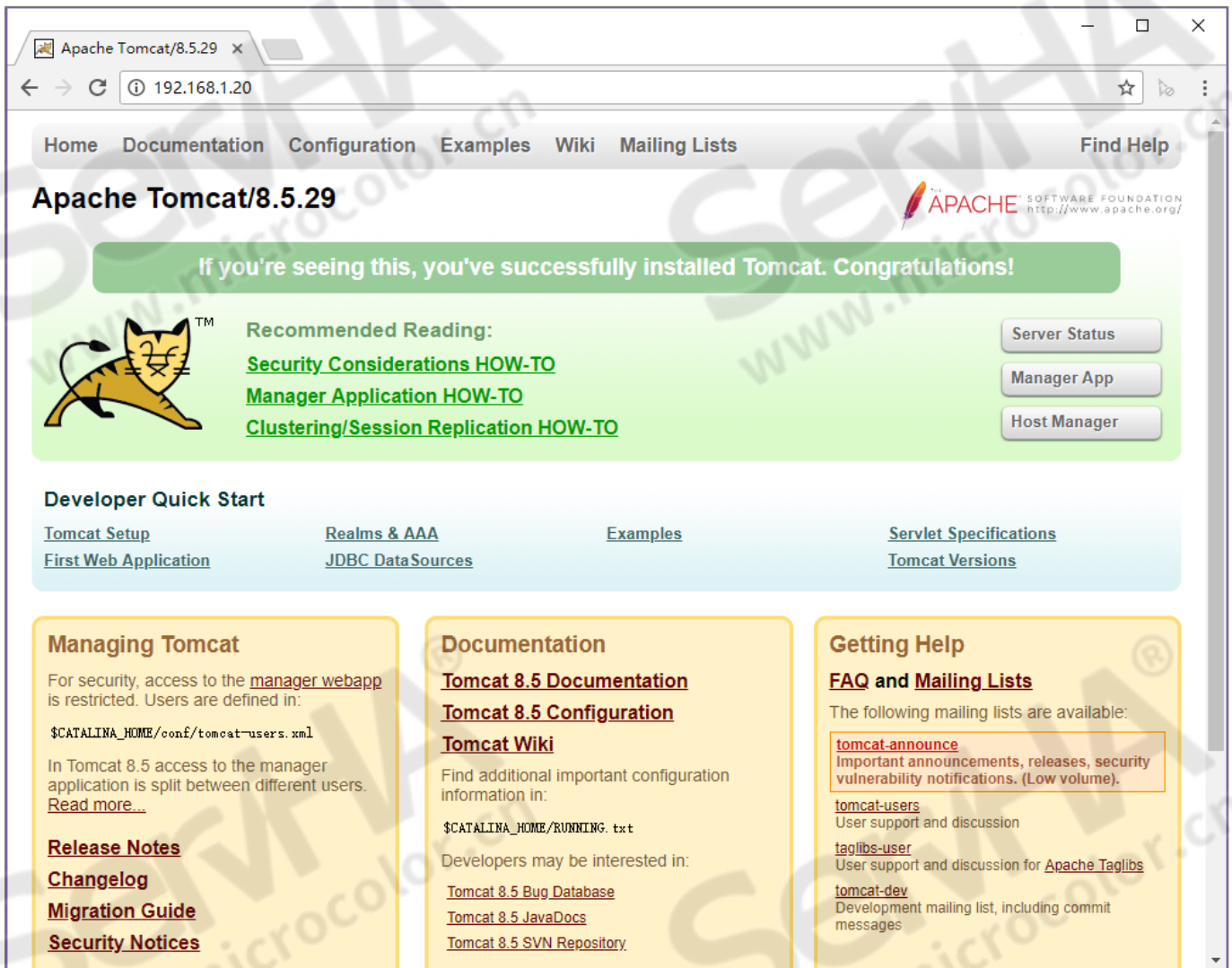
进程检测器（适用于大多数本地运行程序）

检测本机是否运行着指定进程或程序，指定的进程和程序皆运行，则认为应用程序在本机处于运行状态；

自定义检测器（适用于更复杂的情况）

如遇到更复杂的情况，可以自行编写检测脚本，该脚本需要能立即返回“0”或者“非0”的值，两台服务器的检测器会分别调用该脚本，返回“0”则认为应用程序在本机处于运行状态，返回其它值则认为本机应用程序处于停止状态。（注：由于两台服务器都会间隔性的调用该脚本，所以该脚本必须要满足两台服务器都能独立访问到的条件，即处于相同的绝对路径并且不处于数据盘下）

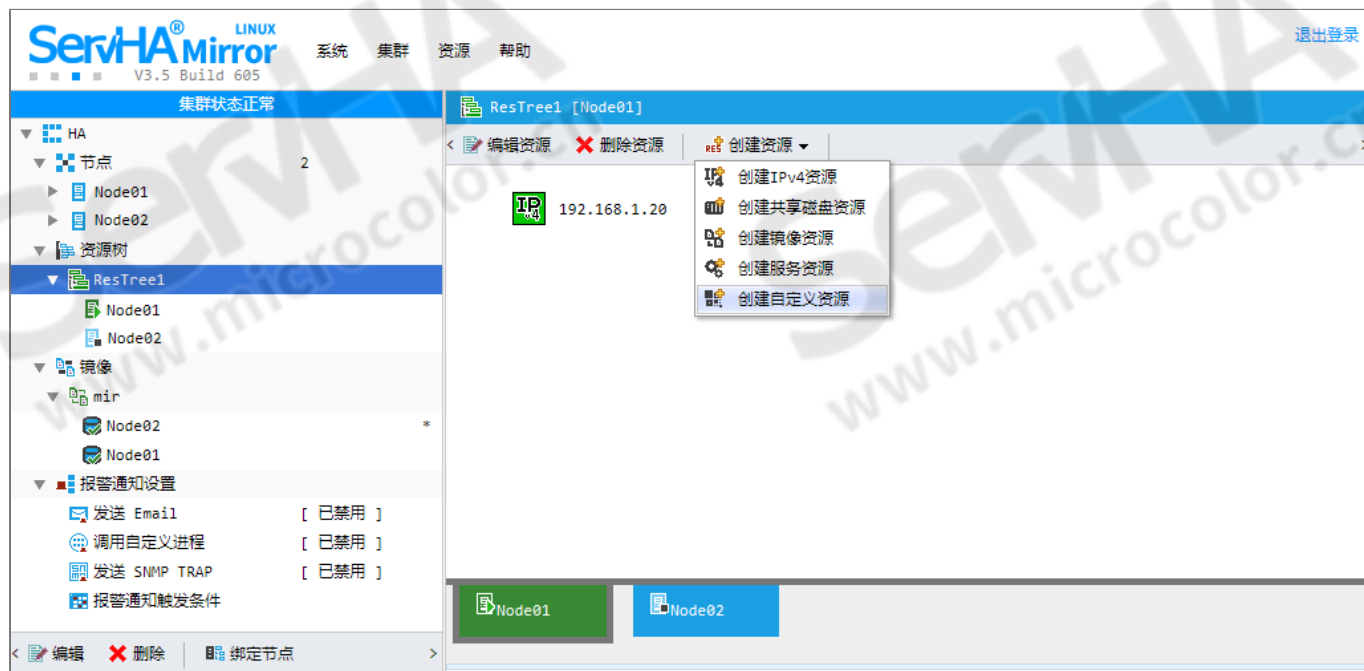
\*例如使用tomcat做WEB服务，启动之后会有java进程监听配置的端口（如80或者8080端口）；使用浏览器进行访问会显示一个页面，如下图：



可以使用URL检测器检测访页面中是否出现了“Tomcat”字符串，并且辅以端口检测器达到检测效果。

## 创建资源

选中资源树并创建自定义资源，如下图：



在“创建自定义资源”窗口中填写“资源名称”、“启动脚本”、“停止脚本”并配置合适的检测器，（若应用带有图形窗口，则需勾选“需要运行在桌面”选项），点击“保存”按钮，如下图为tomcat的示例：

创建自定义资源 - MicroColor ServHA高可用软件综合管理平台

特殊属性

基础属性

高级

自定义资源属性

资源名称:	tomcat		
启动脚本:	/mir/tomcat/bin/startup.sh	启动参数:	
停止脚本:	/mir/tomcat/bin/shutdown.sh	停止参数:	
需要运行在桌面:	<input type="checkbox"/>		

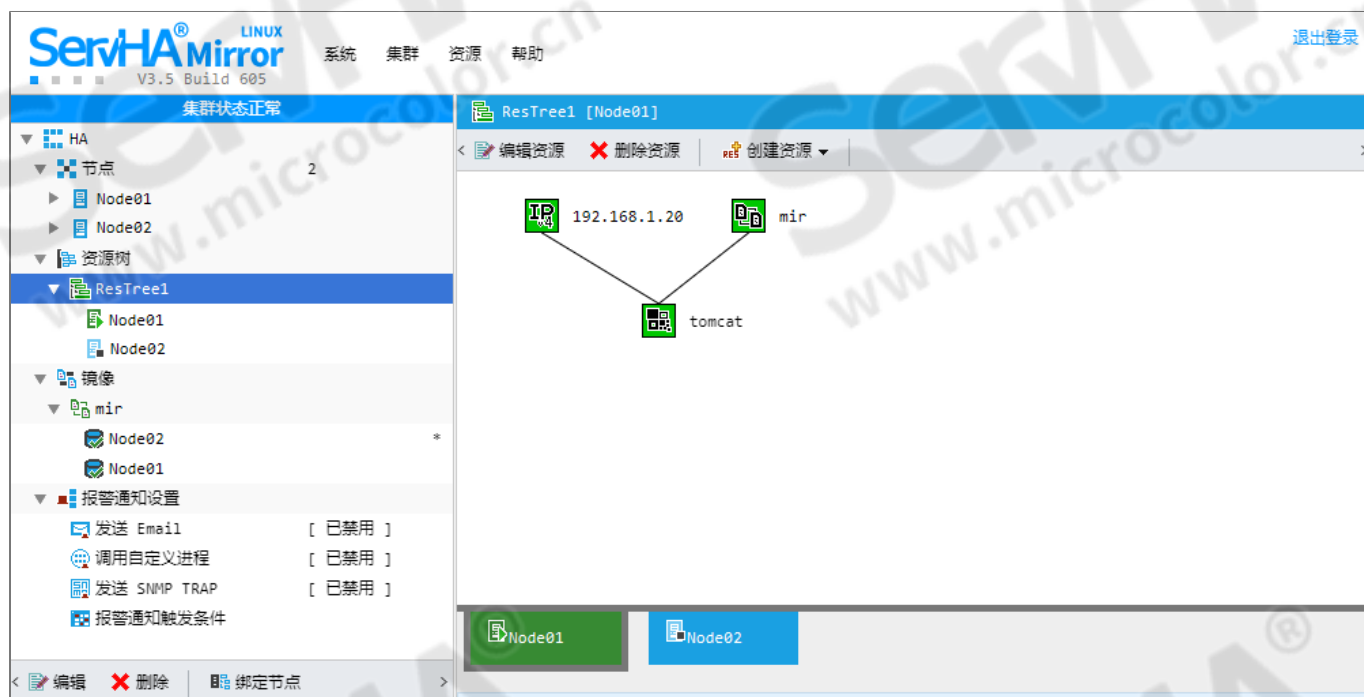
自定义资源检测

*启动后检测超时时间:	120000 ms		
*检测器:	检测端口是否被监听:	<input checked="" type="checkbox"/>	
	*端口号:	80	监听此端口的进程: java
	检测URL能否访问:	<input checked="" type="checkbox"/>	
	*URL:(以http或https开头)	http://192.168.1.20	
	URL返回关键字:	Tomcat	(仅支持英文字符)
	*重试次数:	3	*单次超时时间: 3000 ms
检测进程是否运行:	<input type="checkbox"/>		

保存

取消

资源创建完成后，将该资源的图标拖动到镜像/共享存储资源以及IP资源图标上，建立依赖关系，完成后效果如下图：



## 报警器配置

### 报警器

#### 发送Email

描述：触发该报警器，系统会将报警信息通过已设置的发件邮箱发送给一个或多个接收电子邮箱；

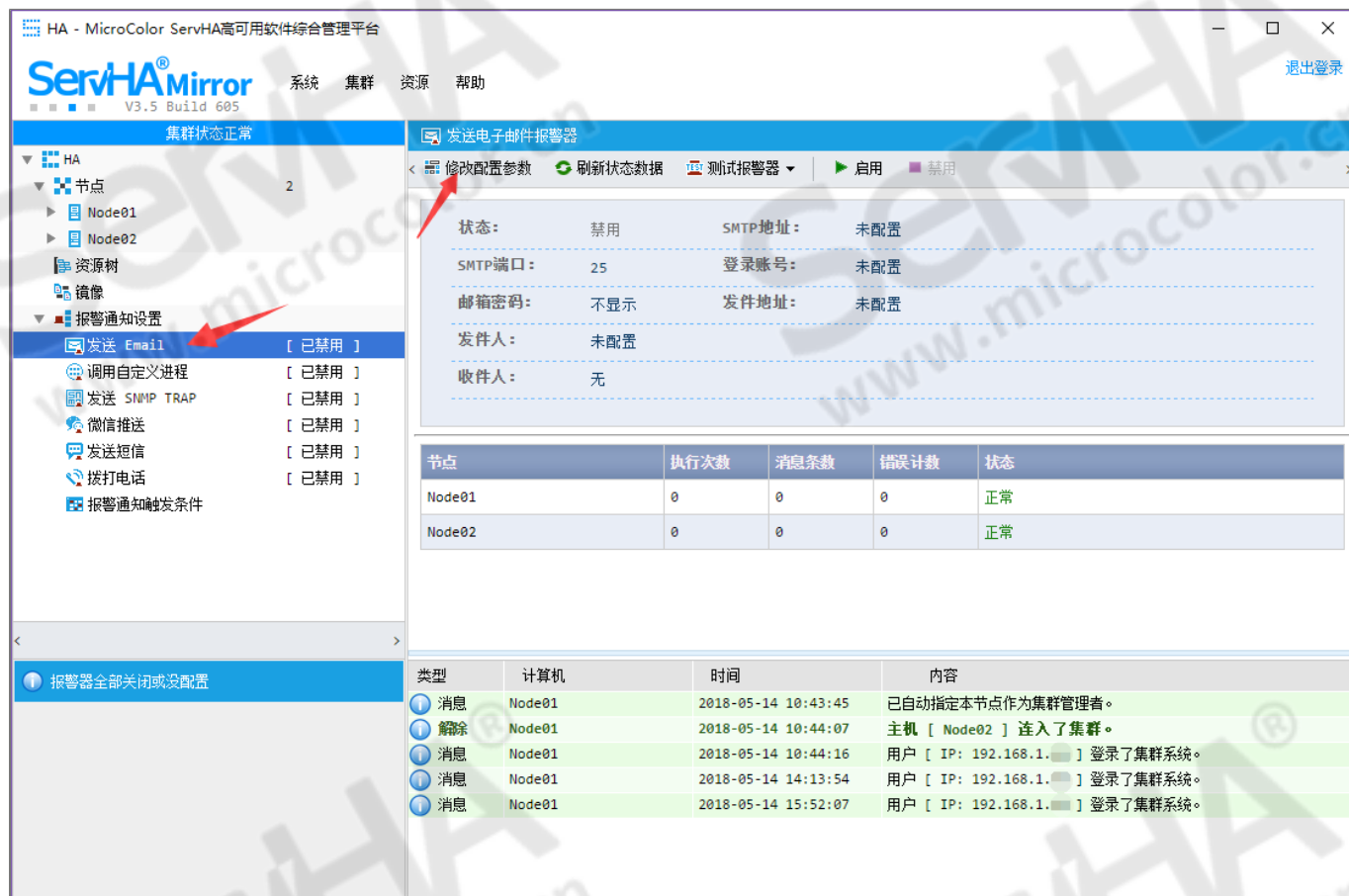
使用条件：服务器可以连接外网或者内网的一个SMTP服务；

所需信息：发件邮箱（SMTP地址、端口，登录账号、密码）、收件邮箱地址（至少一个）；

配置步骤：

1、在配置监控端主界面左侧选择“报警通知设置”下的“发送Email”节点，并在主界面右侧的“发送电子邮件报警器”工具栏中点击“修改配置参数”按钮，如下图；

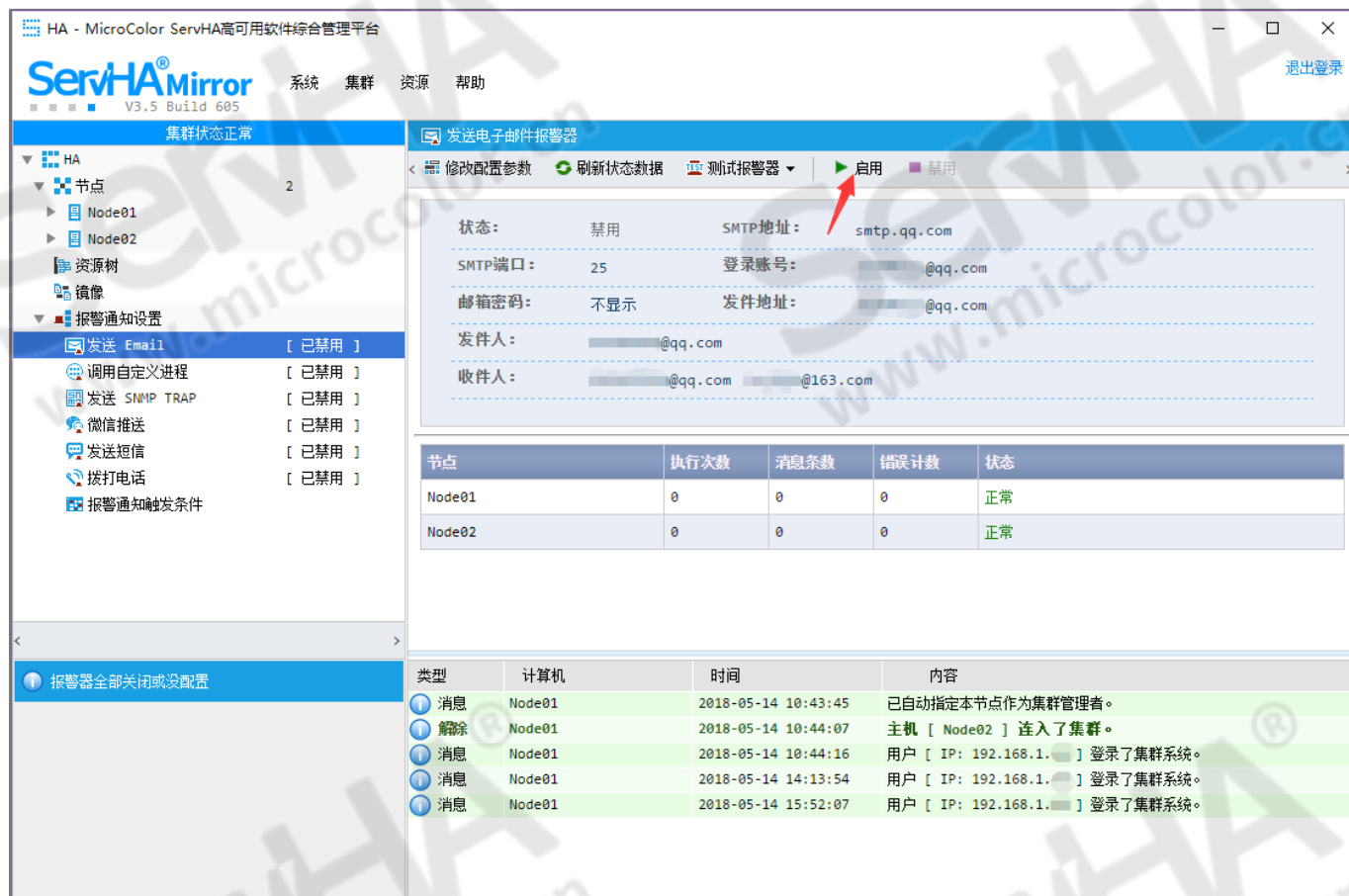




2、在“修改Email电子邮件报警器配置”窗口填写表单，填写完毕后点击左下方的“发送测试邮件”按钮，在确认收到了测试邮件后点击右下方的“保存”按钮，如下图；



3、回到主界面右侧的“发送电子邮件报警器”界面，在工具栏中点击“启用”按钮打开报警器，如下图；



## 自定义进程

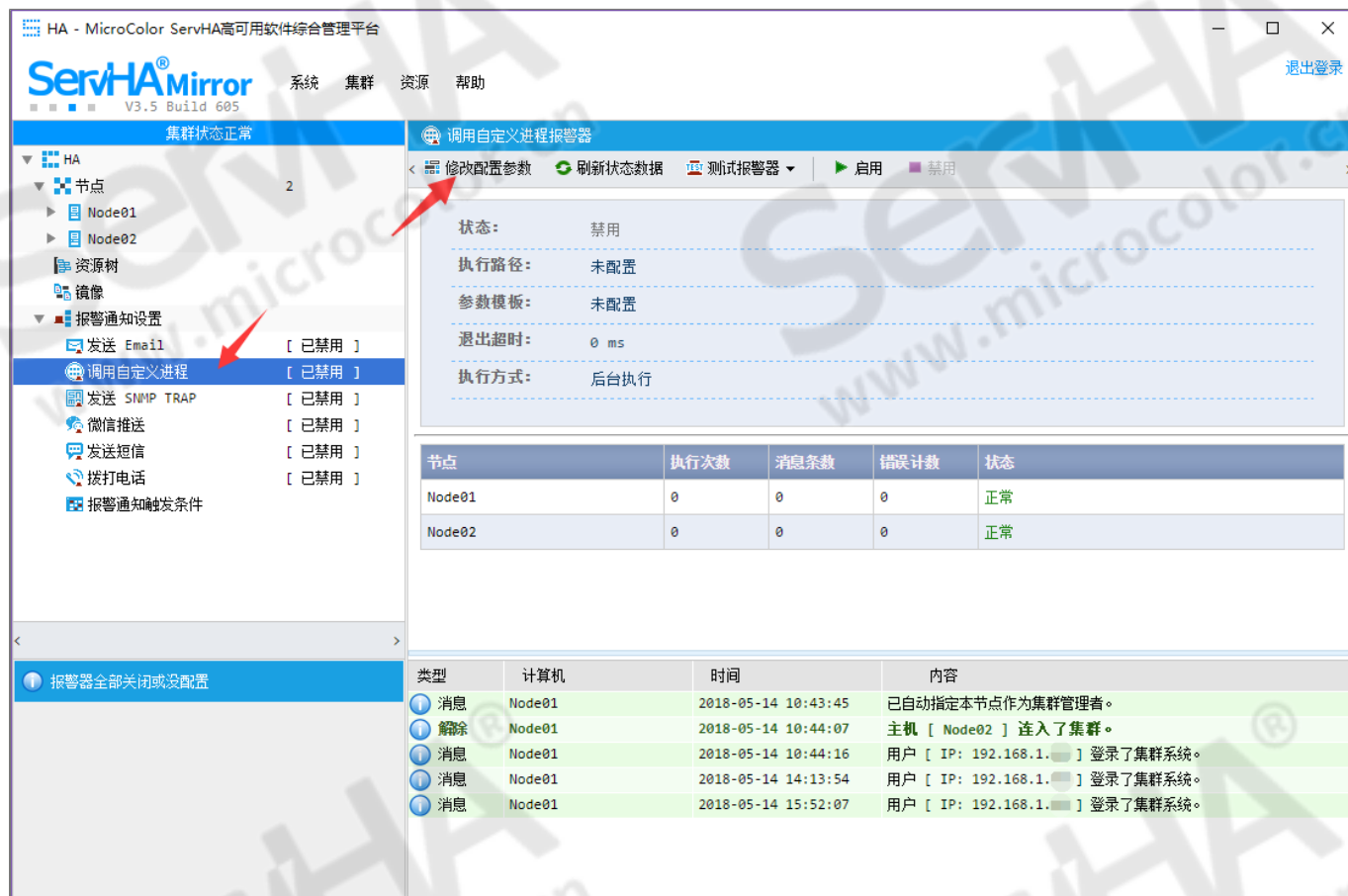
描述: 触发该报警器, 系统会调用服务器上指定程序, 并将报警信息通过参数方式传送给该程序;

使用条件: 有能力自行开发一个可以接收参数启动的报警程序;

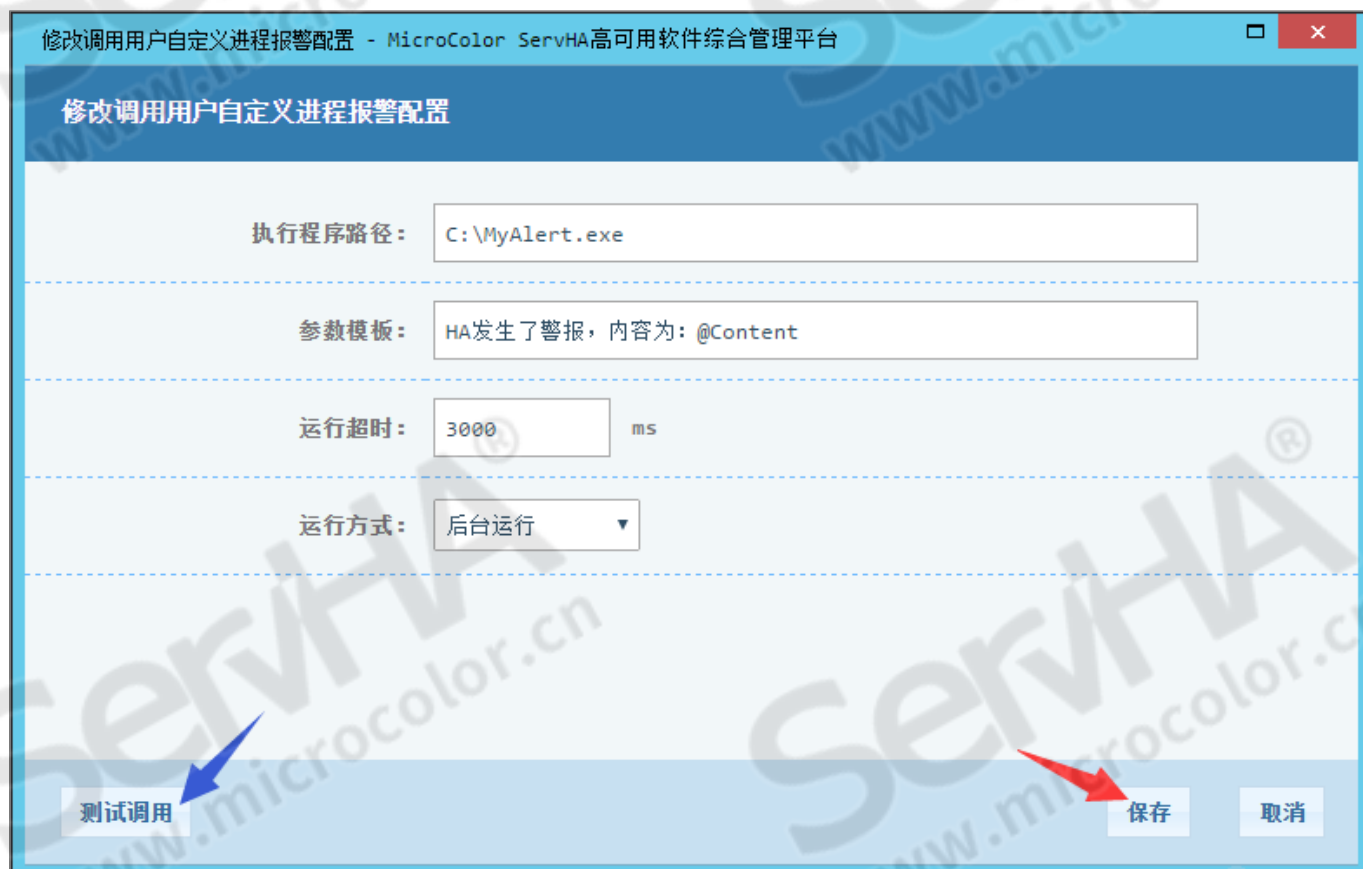
所需信息: 自定义报警程序的绝对路径 (每台服务器上需一致);

配置步骤:

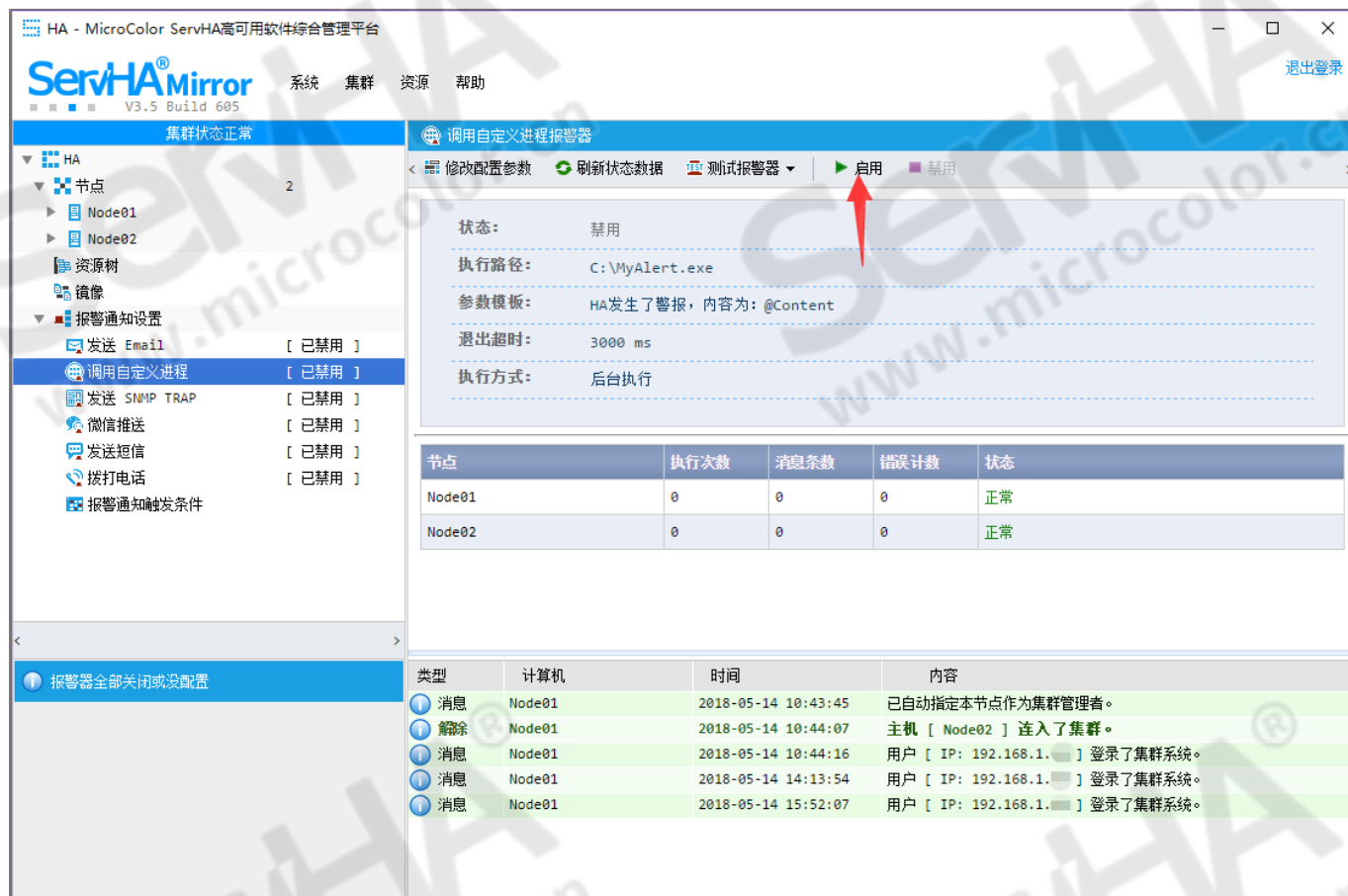
1、在配置监控端主界面左侧选择“报警通知设置”下的“调用自定义进程”节点, 并在主界面右侧的“调用自定义进程报警器”工具栏中点击“修改配置参数”按钮, 如下图;



2、在“修改调用用户自定义进程报警配置”窗口填写表单，填写完毕后点击左下方的“测试调用”按钮，在确认自定义程序收到了测试调用后点击右下方的“保存”按钮，如下图；



3、回到主界面右侧的“调用自定义进程报警器”界面，在工具栏中点击“启用”按钮打开报警器，如下图；



## SNMP Trap

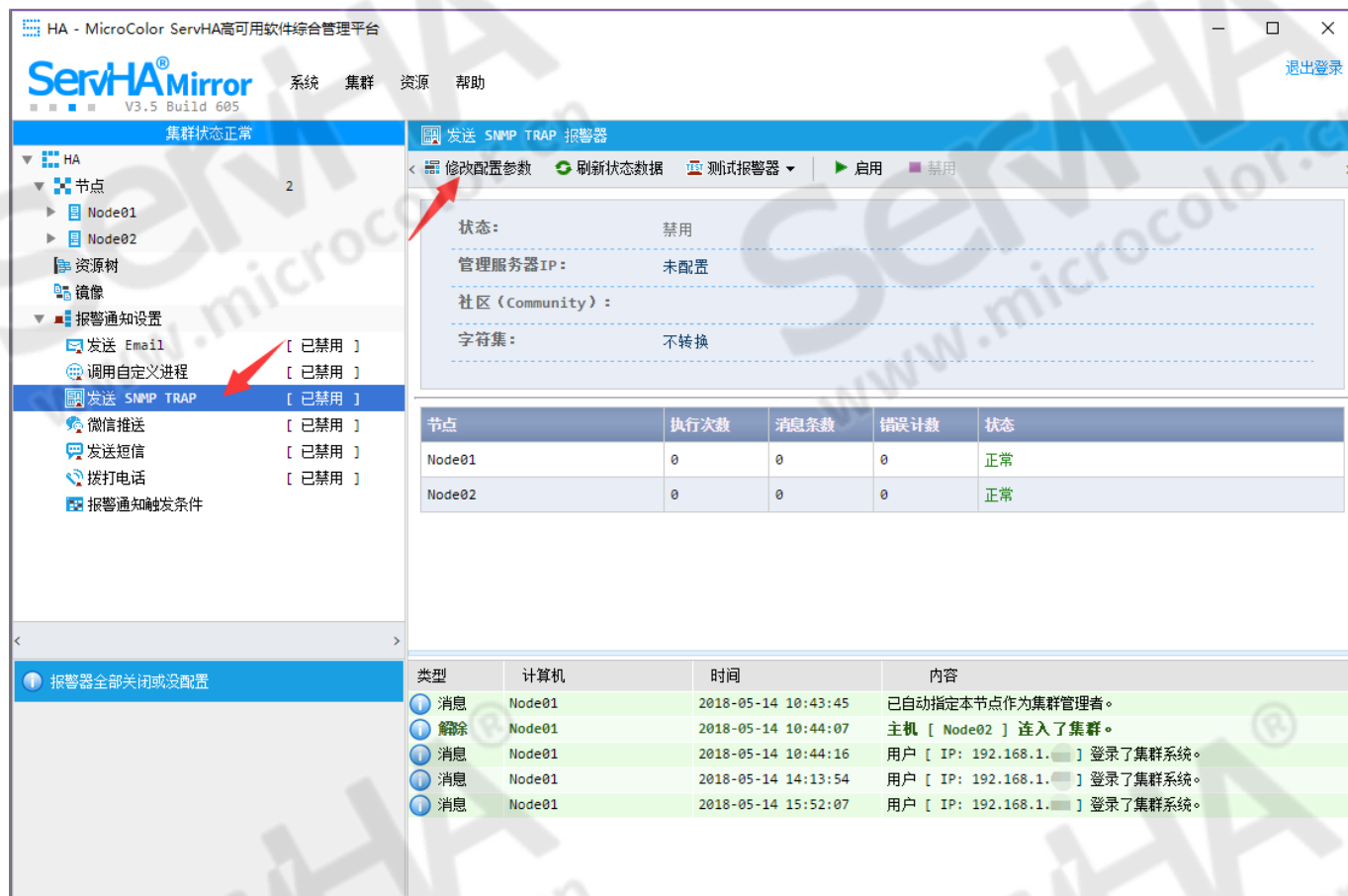
描述: 触发该报警器, 系统会使用SNMP协议发送含有报警信息的SNMP TRAP封包;

使用条件: 服务器所在网络中拥有可用的SNMP服务器;

所需信息: SNMP服务器IP、mib文件(安装目录内含有)

配置步骤:

1、在配置监控端主界面左侧选择“报警通知设置”下的“发送 SNMP TRAP”节点, 并在主界面右侧的“发送 SNMP TRAP报警器”工具栏中点击“修改配置参数”按钮, 如下图;

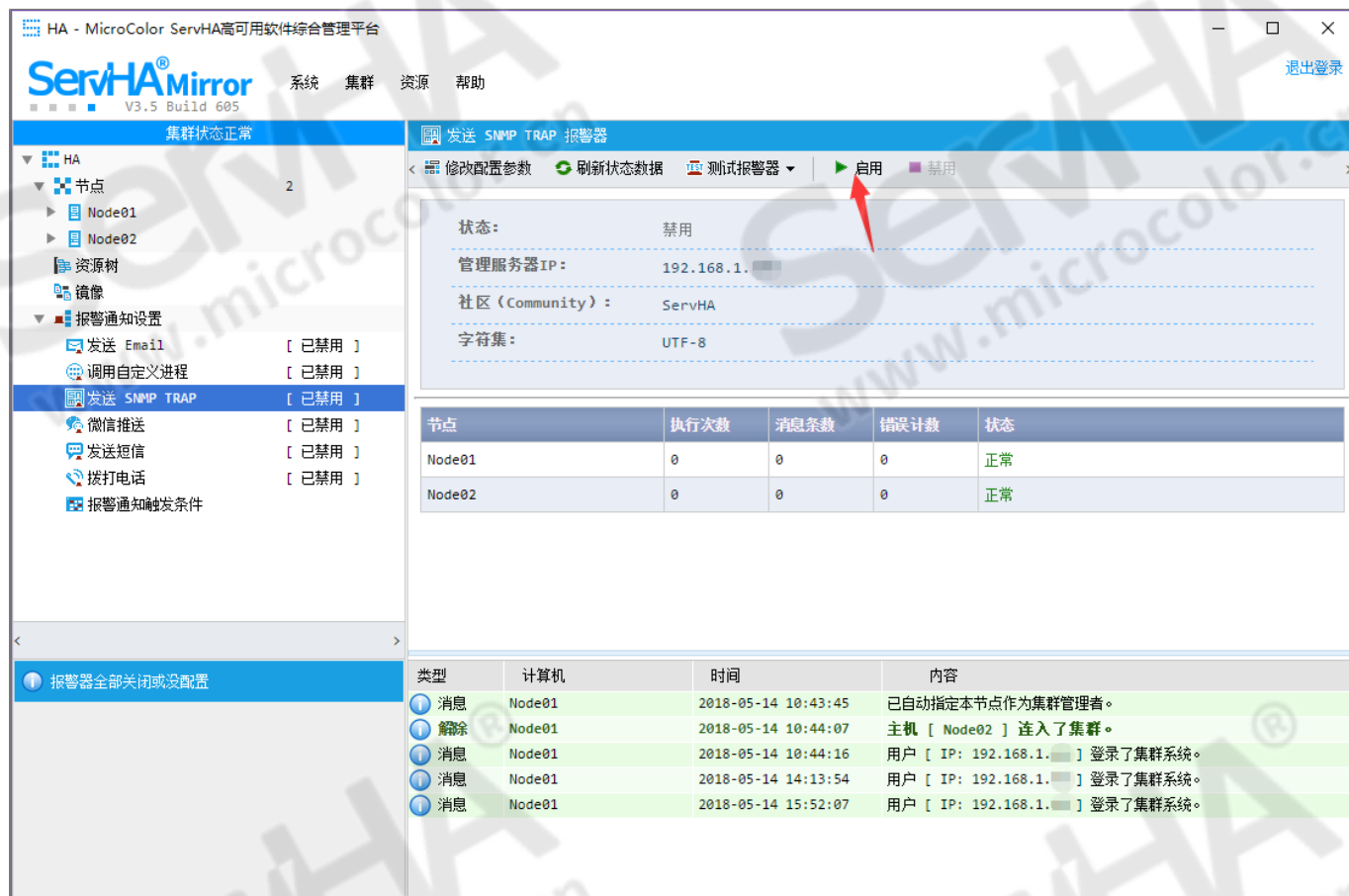


2、在“发送 SNMP TRAP 报警器”窗口填写表单，填写完毕后点击左下方的“测试调用”按钮，在确认SNMP服务器收到了测试SNMP TRAP封包后点击右下方的“保存”按钮，如下图；



3、回到主界面右侧的“发送SNMP TRAP报警器”界面，在工具栏中点击“启用”按钮打开报警器，如下图；





## 微信推送

描述：触发该报警器，系统会将报警信息通过微信消息的方式发送给一个或多个微信用户；

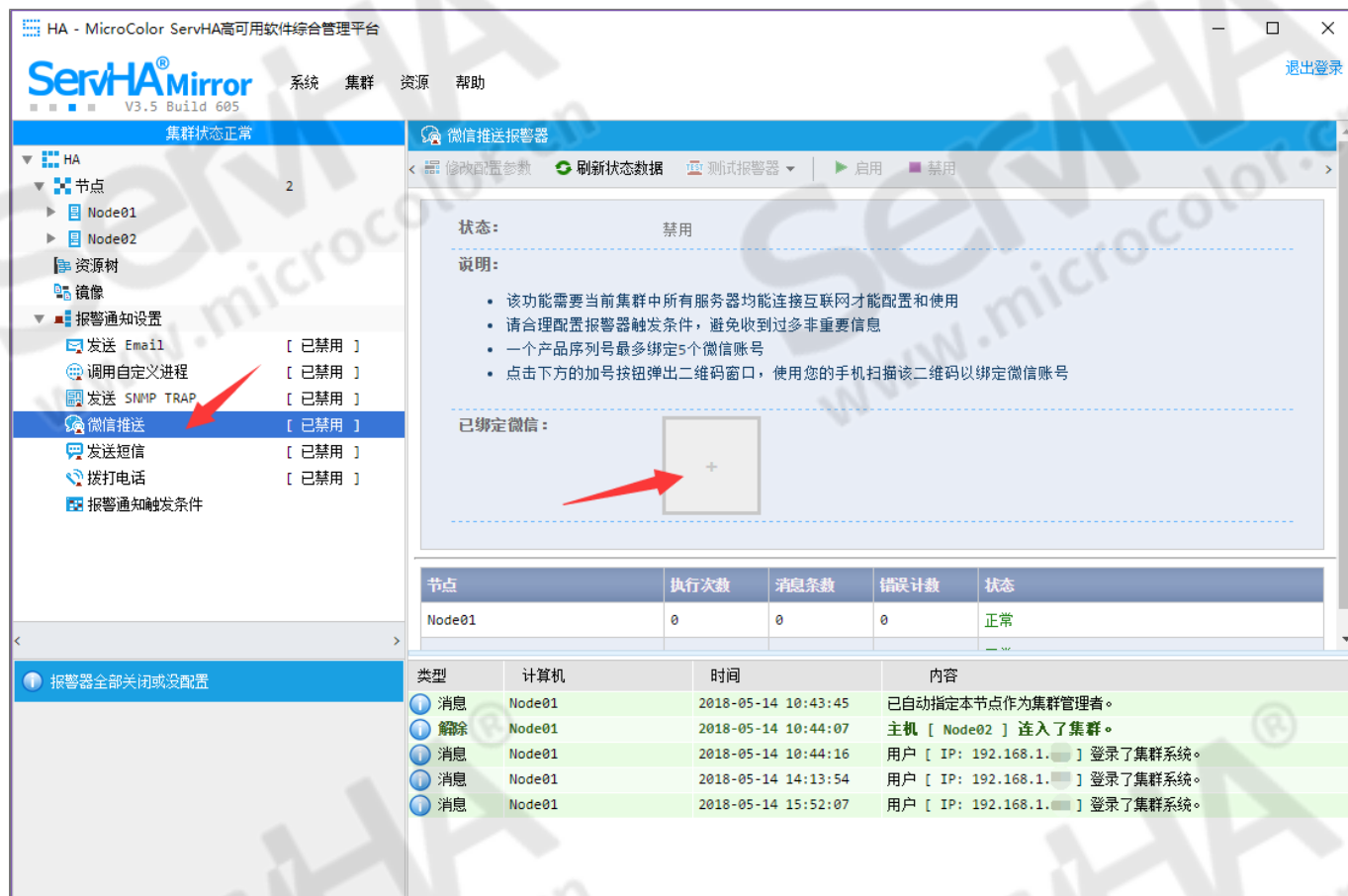
使用条件：安装ServHA的服务器能连外网、ServHA软件已使用序列号授权；

所需信息：接收报警信息的微信号（关注ServHA公众号）；

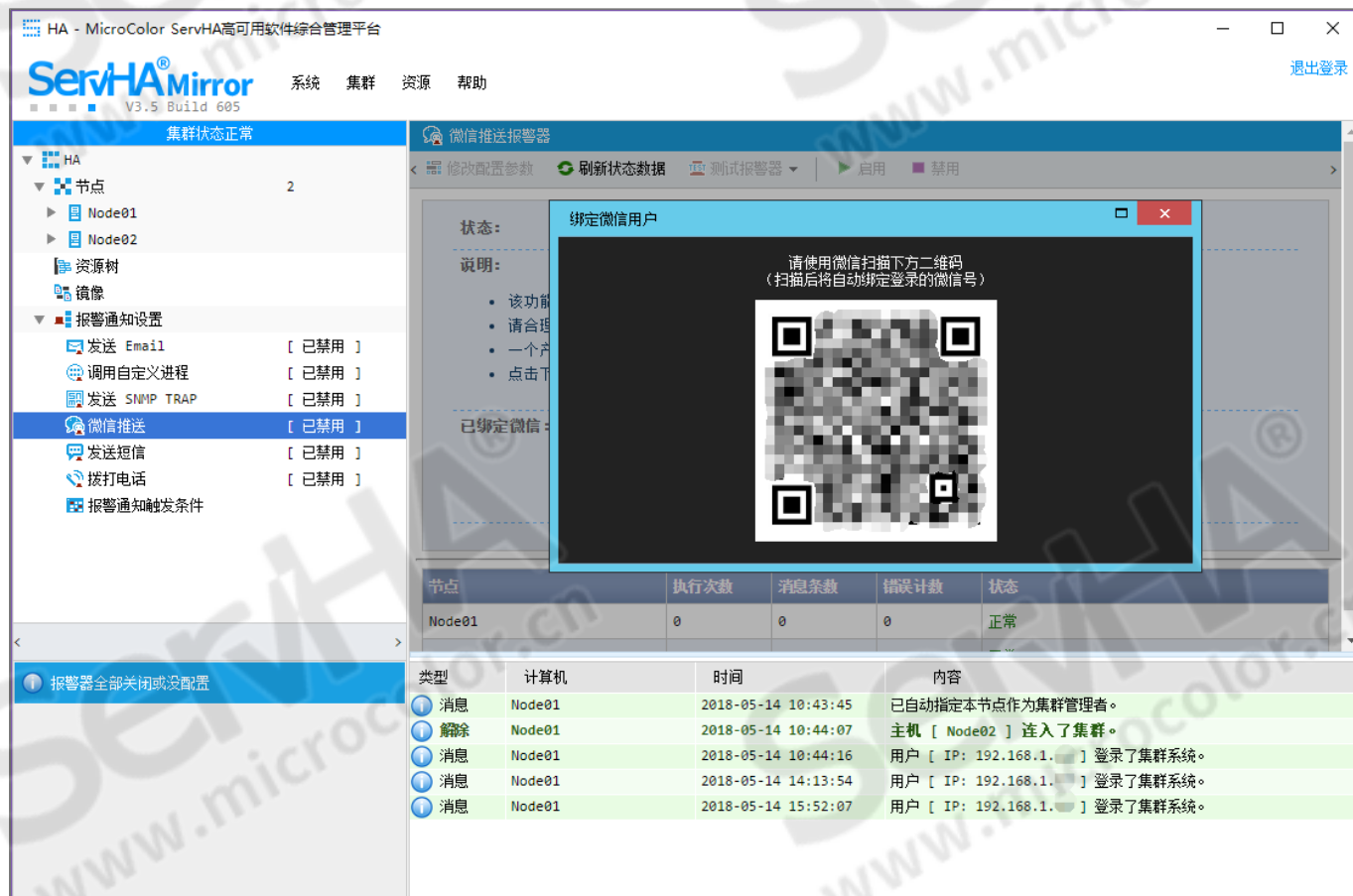
配置步骤：

1、在配置监控端主界面左侧选择“报警通知设置”下的“微信推送”节点，并在主界面右侧的“微信推送报警器”界面中点击“已绑定微信”栏后的加号(+)按钮，如下图；

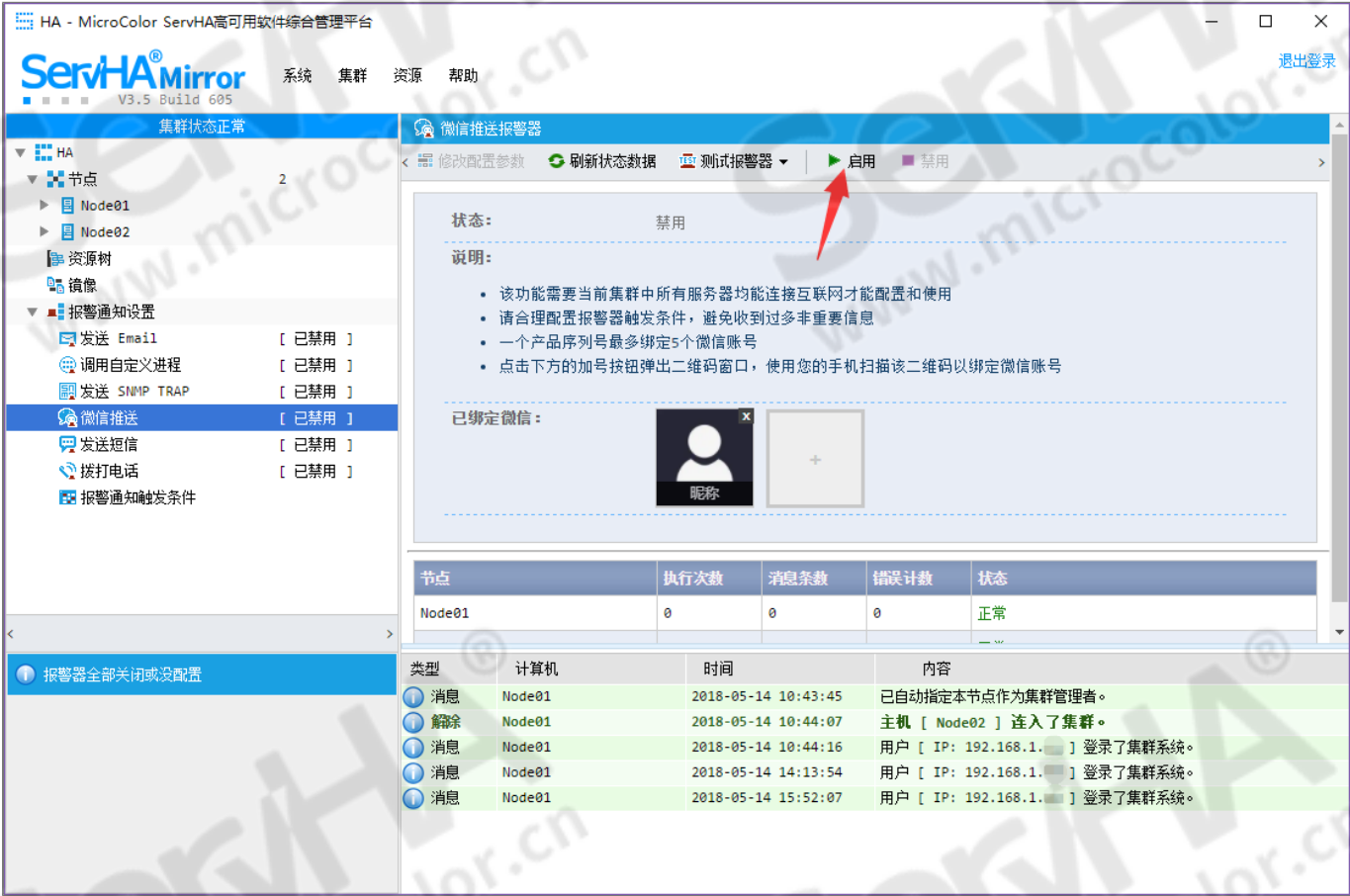




2、使用手机微信扫一扫功能扫描界面上生成的二维码（如该微信号未曾关注ServHA公众号，会提示需要先关注，点击关注后），等待系统自动添加绑定，如下图；



3、回到主界面右侧的“微信推送报警器”界面，在工具栏中点击“启用”按钮打开报警器，如下图；



## 发送短信

描述：触发该报警器，系统会将报警信息通过短信发送给一个或多个手机号；

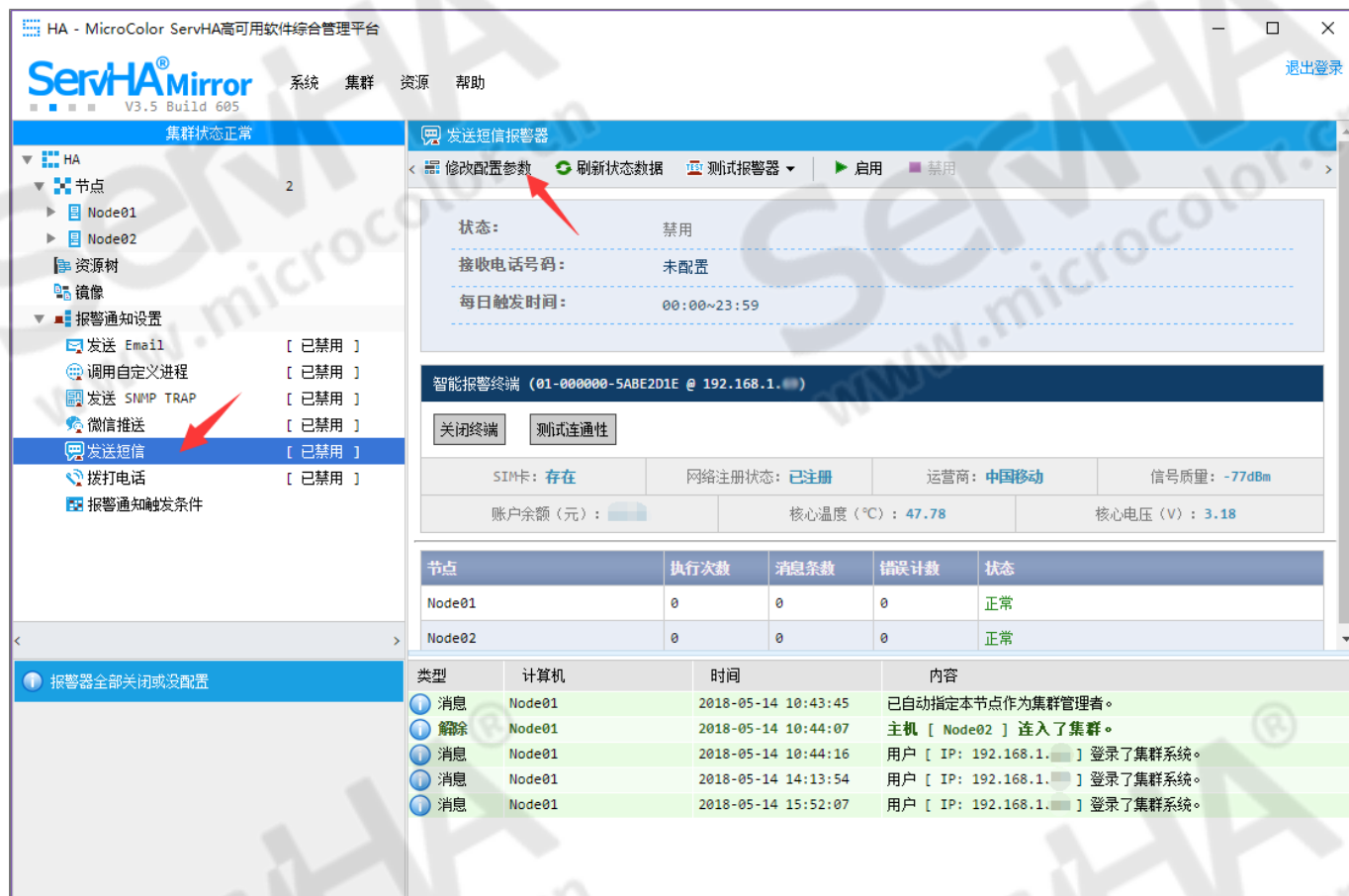
使用条件：拥有ServHA智能告警终端，终端内插有可以发送短信的SIM卡；

所需信息：用于接收短信的手机号；

配置步骤：

**\*若尚未初始化ServHA智能告警终端，请先按照附件中的步骤初始化；**

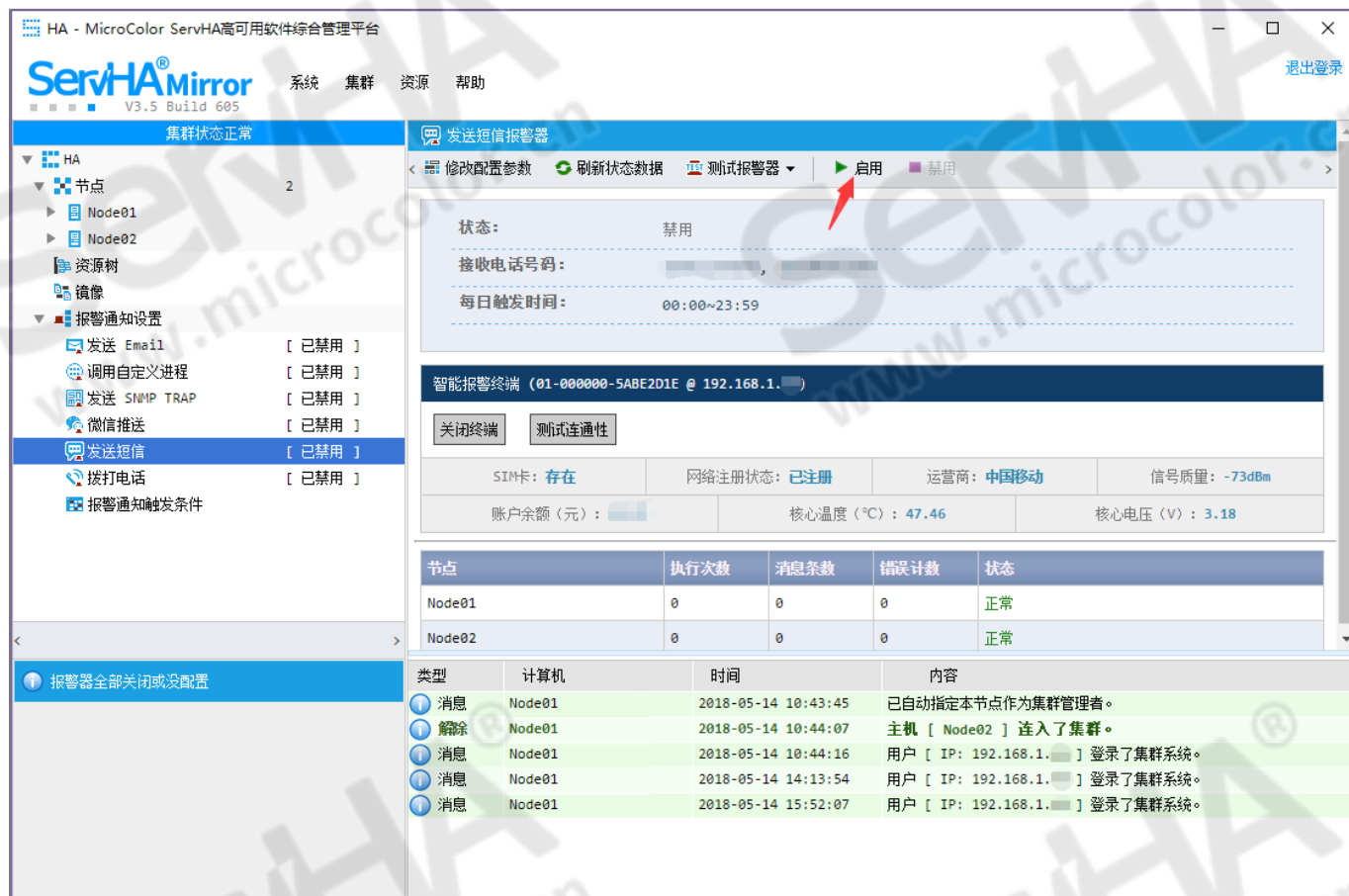
1、在配置监控端主界面左侧选择“报警通知设置”下的“发送短信”节点，并在主界面右侧的“发送短信报警器”工具栏中点击“修改配置参数”按钮，如下图；



2、在“发送短信 报警器”窗口填写表单，填写完毕后点击左下方的“测试调用”按钮，在确认收到了测试短信后点击右下方的“保存”按钮，如下图；



3、回到主界面右侧的“发送短信报警器”界面，在工具栏中点击“启用”按钮打开报警器，如下图；



## 拨打电话

描述：触发该报警器，系统会拨打设定的电话号码（手机或固话），并将报警信息使用语音播报出来；

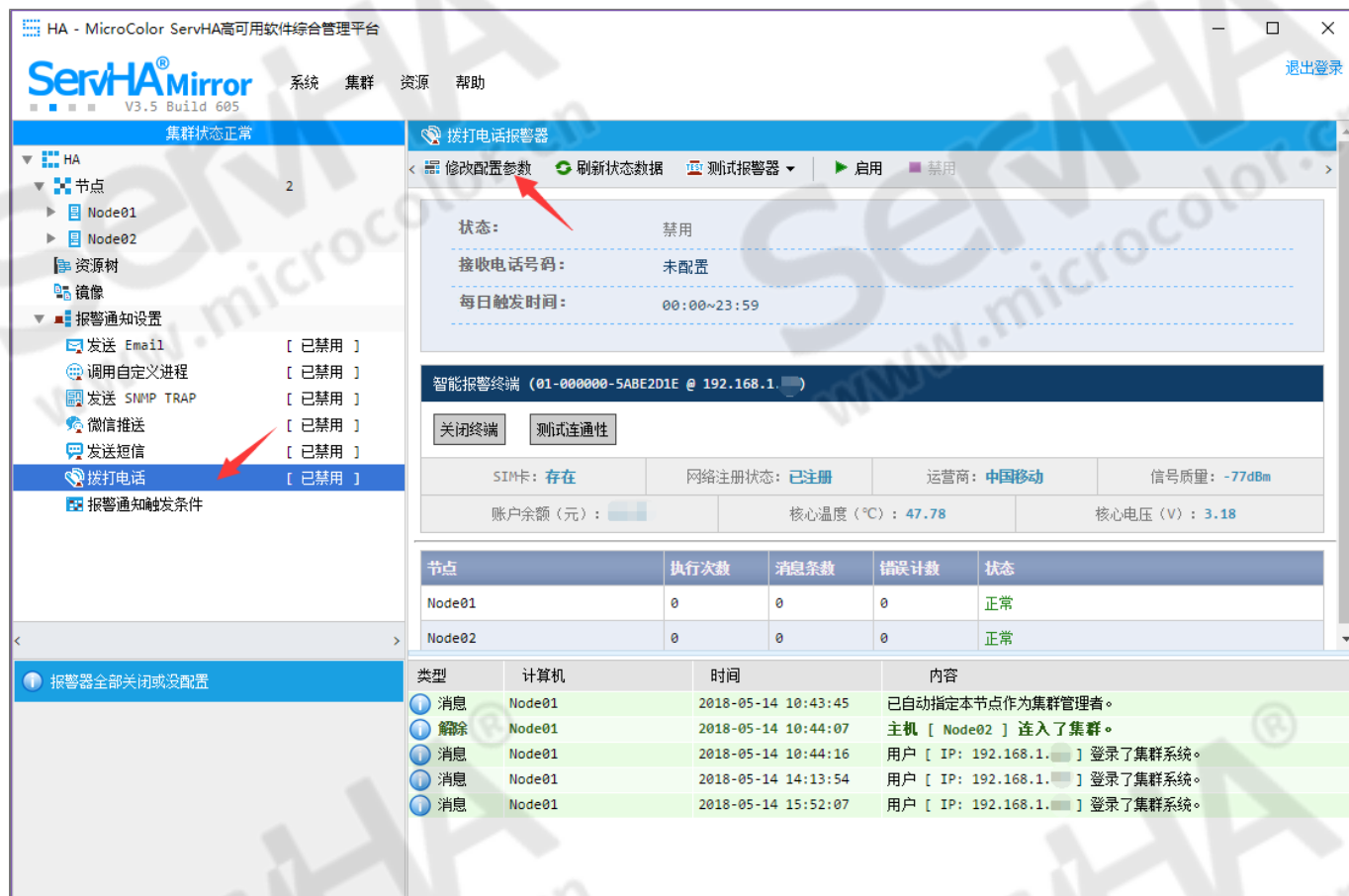
使用条件：拥有ServHA智能告警终端，终端内插有可以拨打电话的SIM卡；

所需信息：用于接收电话的手机或固话号码；

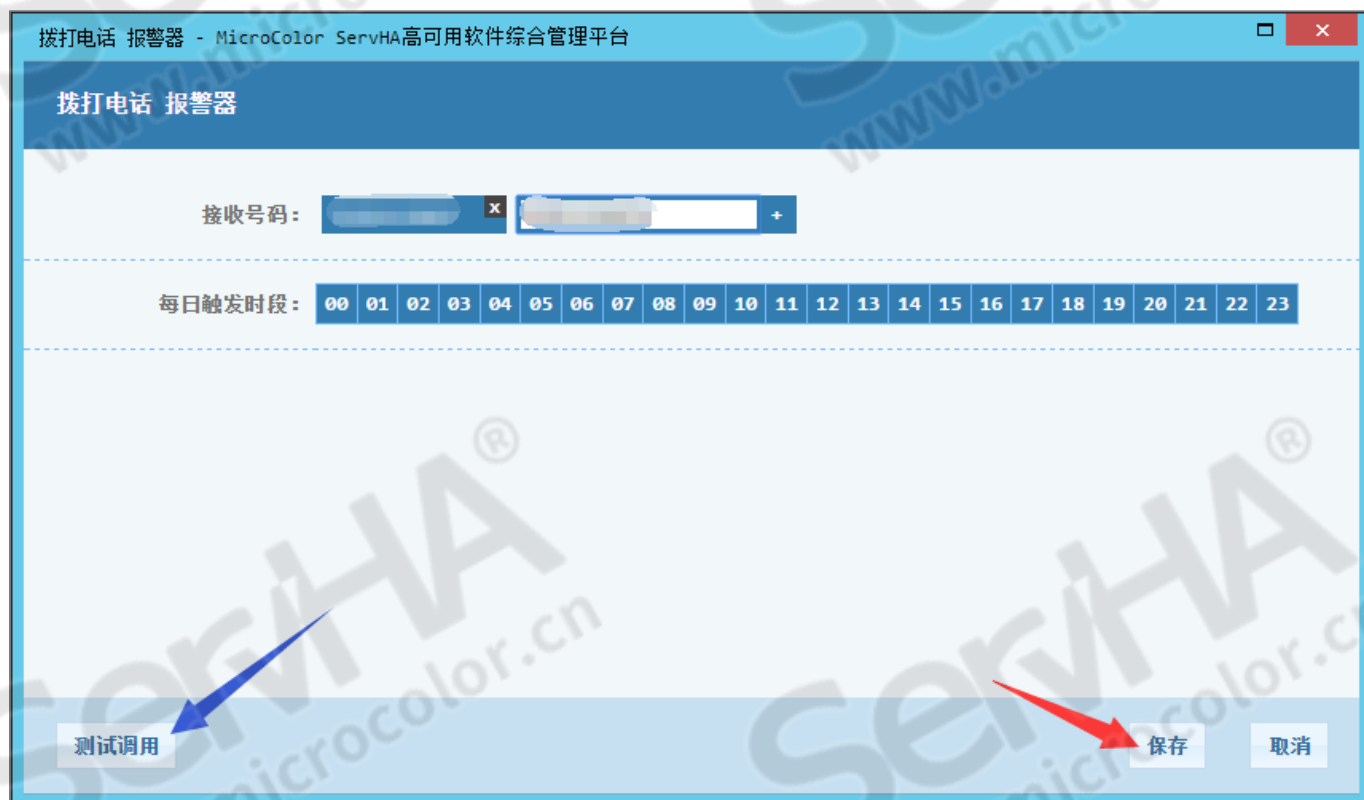
配置步骤：

**\*若尚未初始化ServHA智能告警终端，请先按照附件中的步骤初始化；**

1、在配置监控端主界面左侧选择“报警通知设置”下的“拨打电话”节点，并在主界面右侧的“拨打电话报警器”工具栏中点击“修改配置参数”按钮，如下图；

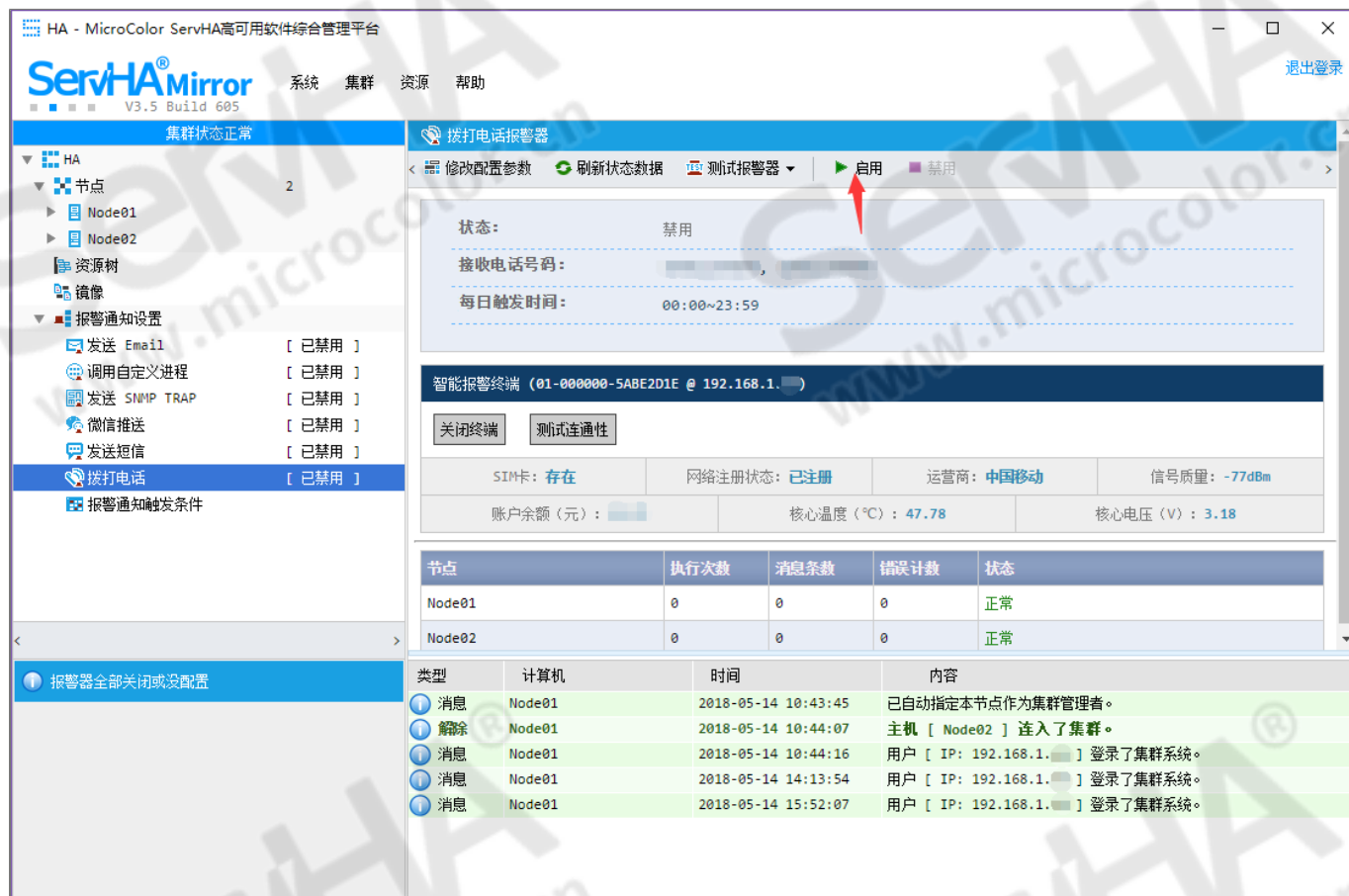


2、在“拨打电话 报警器”窗口填写表单，填写完毕后点击左下方的“测试调用”按钮，在确认收到了测试通话后点击右下方的“保存”按钮，如下图；



3、回到主界面右侧的“拨打电话报警器”界面，在工具栏中点击“启用”按钮打开报警器，如下图；





## 附件：初始化ServHA智能告警终端

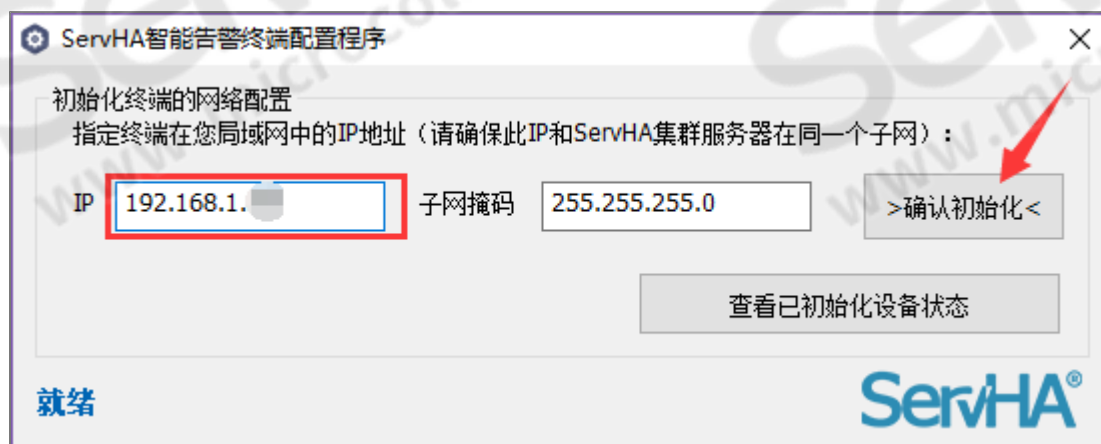
\*本操作适用于新获取的ServHA智能告警终端（以下简称“终端”）的初始化，或者是之前已经初始化过的终端的IP地址修改；

- 1、将终端通电，并使用网线连至任意计算机（windows操作系统）或者局域网中；
- 2、在能访问到终端的计算机（windows操作系统）上运行安装光盘（或者从ServHA官网下载）的“初始化ServHA智能报警终端.exe”，如下图；





3、在“ServHA智能告警终端配置程序”中填写希望设置的终端IP，并点击“确认初始化”按钮，如下图；



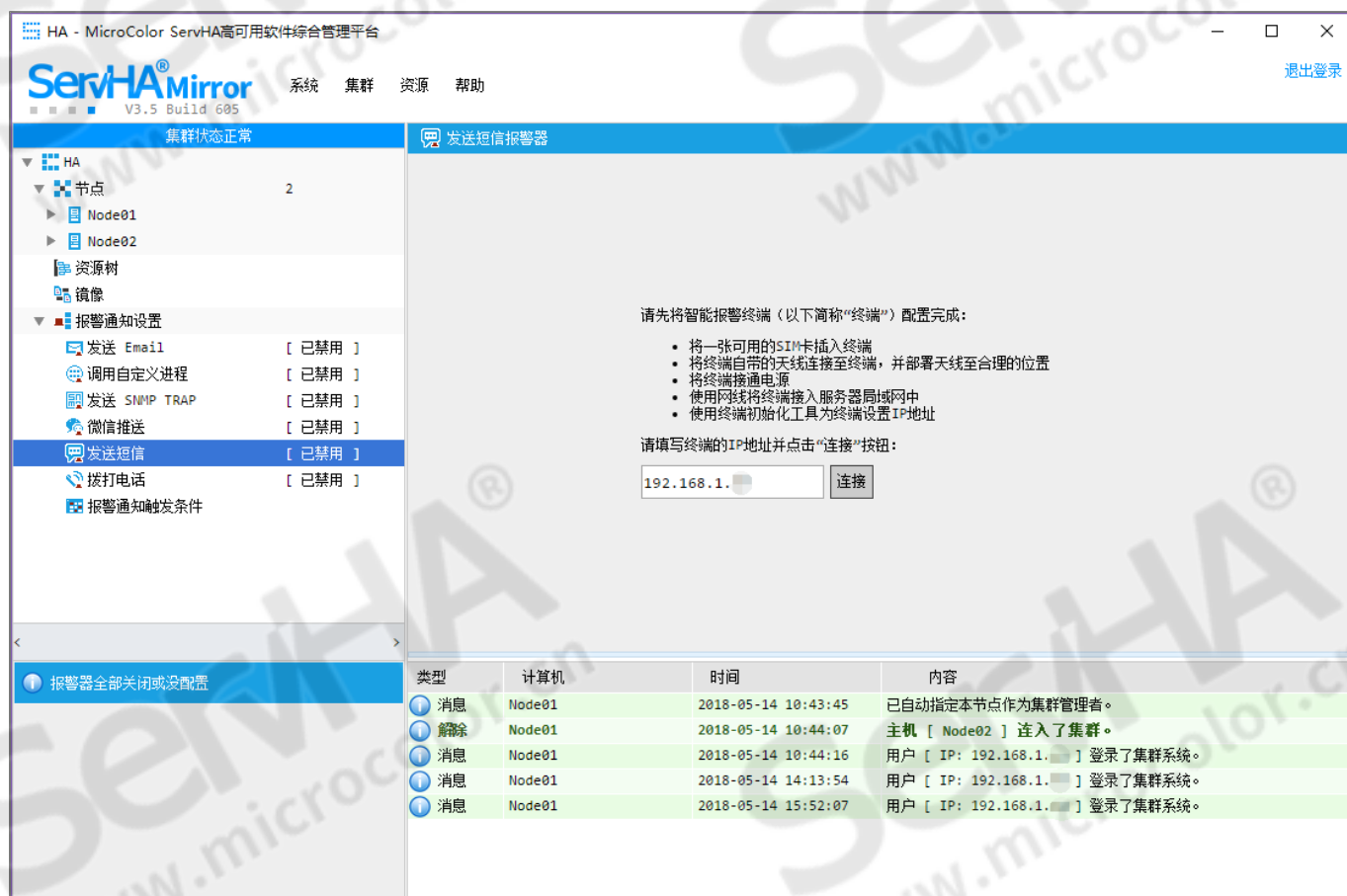
4、等待后台操作完成，并牢记设置的终端IP，如果终端中已插入可用的SIM卡，可以在此测试短信和拨打电话功能，完成后点击“结束配置”按钮退出程序，如下图；



5、将终端网络移动至服务器所在局域网，通电，并组装好终端天线，放置在合适位置；

6、确保终端中已插入了可用的SIM卡；

7、在ServHA主界面中选择“报警通知设置”下的“发送短信”或“拨打电话”中的任意一个节点，在右侧界面中填入之前设置的终端IP地址，点击“连接”按钮，如下图。当右侧界面中能看到终端信息时初始化完成；



## 触发条件

在报警器配置完成并启用后，需要为报警器绑定系统事件，简单的说就是当系统的某个模块发生某种情况时触发某种报警器，这样，配置好的报警器才能被正常触发。

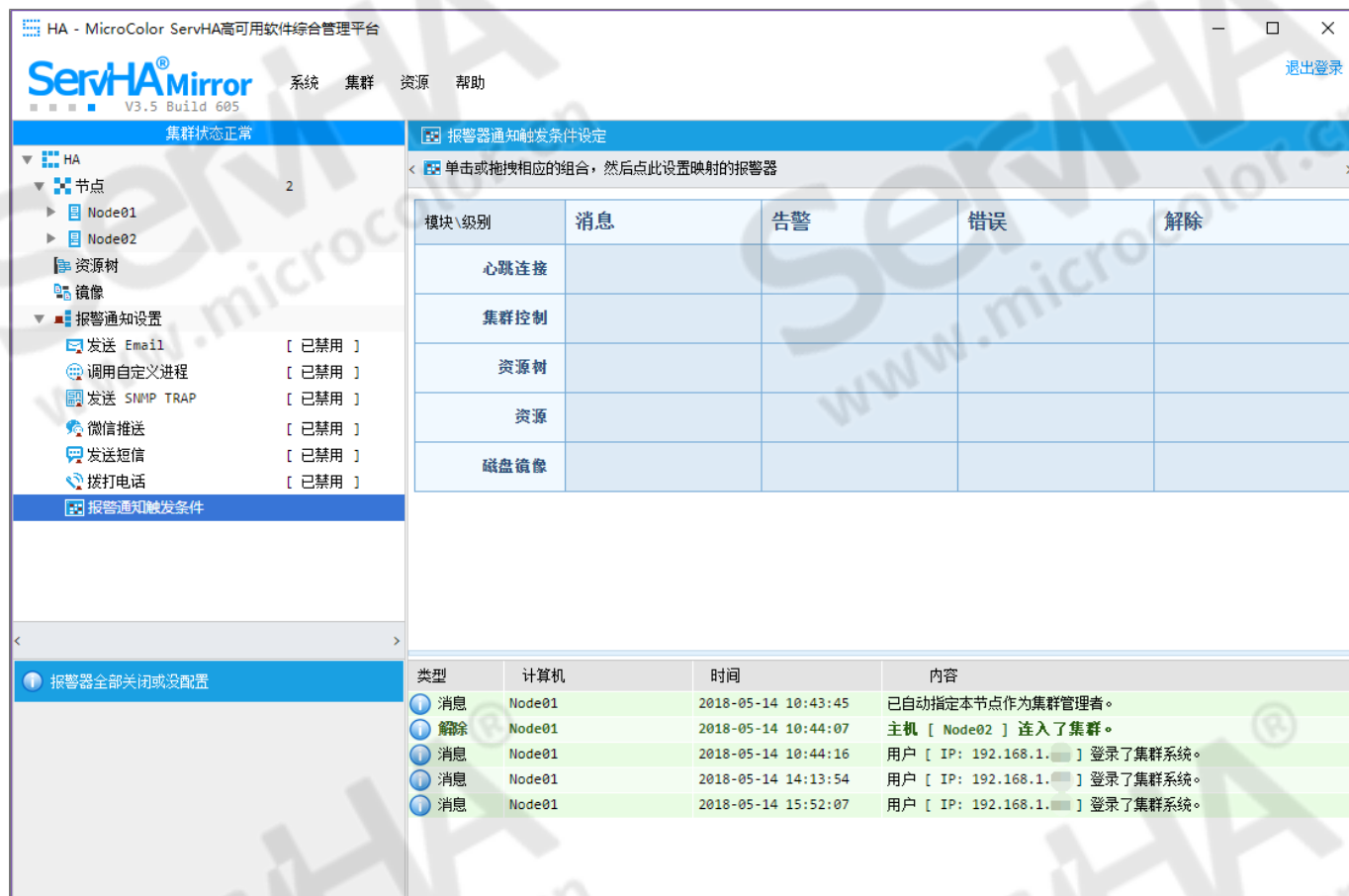
报警器与触发条件的绑定可以根据需求自由组合。以下是报警器触发条件的解释：

模块 \\级别	消息	告警	错误	解除
心跳 连接	--	心跳丢失	心跳连接使用的网卡故障	心跳恢复正常
集群 控制	集群通 常日志	部分仲裁失效 ，用户登录失败	集群全局错误，仲裁错误	仲裁恢复，集群全局 故障恢复
资源 树	--	资源树状态降级	资源树运行错误，资源树不能 运行等	资源树错误恢复，资 源树恢复可运行等
资源	资源通 常消息	资源运行中崩溃 等警告信息	资源启停操作错误，资源不满 足启动条件和其他错误等	资源恢复可运行
镜像	镜像通 常消息	部分镜像链路故 障灯	镜像脑裂，镜像盘丢失等错误	镜像恢复正常

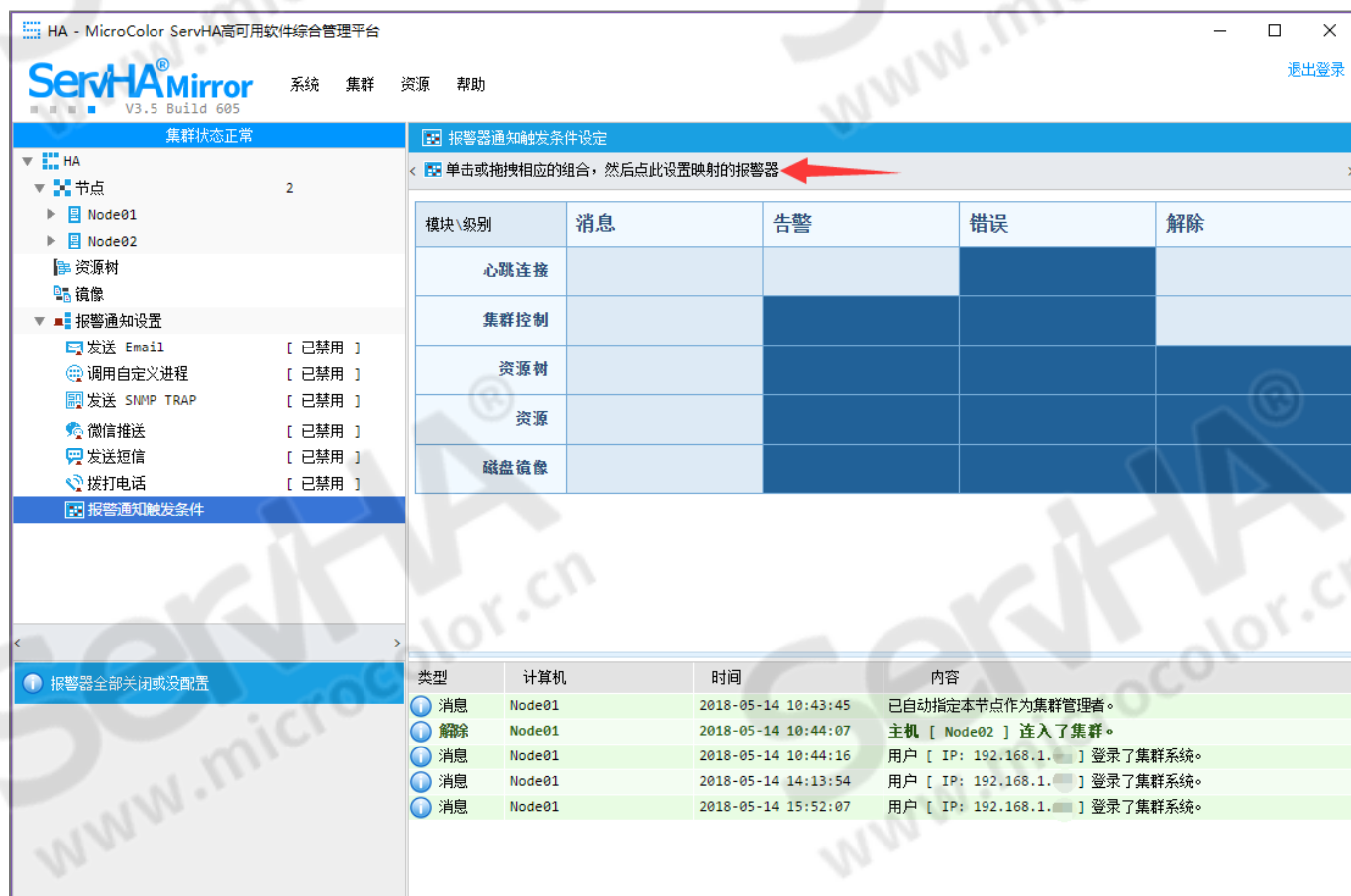
注：报警器只有在启用状态才生效。当对集群进行维护时，可以先暂时禁用报警器，防止维护操作触发报警，维护完毕后需手动启用报警器。

配置步骤：

1、在主界面左侧选择“报警通知设置”下的“报警通知触发条件”节点，如下图；



2、在主界面右侧的“报警器通知触发条件设定”界面中，选择一个或多个触发条件（可按住Ctrl、Shift键或拖动鼠标选择），并点击工具栏上的“单击或拖拽响应的组合，然后点此设置映射的报警器”按钮打开“修改报警器触发条件”界面，如下图；



3、在“修改报警器触发条件”界面中选择一种（或多种）报警器，点击保存按钮保存配置，如下图；

修改报警器触发条件 - MicroColor ServHA高可用软件综合管理平台

修改报警器触发条件

请在下面选择要设置的报警器，保持灰色选中状态为不修改该项目。

☒ 发送电子邮件

☐ 调用自定义进程

☐ 发送SNMPTRAP

☒ 微信推送

☒ 发送短信

☒ 拨打电话

保存

取消

4、回到“报警器通知触发条件设定”界面即可看到报警器的绑定信息，如下图；

HA - MicroColor ServHA高可用软件综合管理平台

ServHA Mirror

V3.5 Build 605

系统 集群 资源 帮助

退出登录

集群状态正常

HA

节点 2

Node01

Node02

资源树

镜像

报警通知设置

发送 Email

调用自定义进程

发送 SNMP TRAP

微信推送

发送短信

拨打电话

报警通知触发条件

报警器全部关闭或没配置

报警器通知触发条件设定

单击或拖拽相应的组合，然后点此设置映射的报警器

模块\级别	消息	告警	错误	解除
心跳连接				
集群控制				
资源树				
资源				
磁盘镜像				

类型	计算机	时间	内容
消息	Node01	2018-05-14 10:43:45	已自动指定本节点作为集群管理者。
解除	Node01	2018-05-14 10:44:07	主机 [ Node02 ] 连入了集群。
消息	Node01	2018-05-14 10:44:16	用户 [ IP: 192.168.1. ] 登录了集群系统。
消息	Node01	2018-05-14 14:13:54	用户 [ IP: 192.168.1. ] 登录了集群系统。
消息	Node01	2018-05-14 15:52:07	用户 [ IP: 192.168.1. ] 登录了集群系统。

# 授权

ServHA 安装后有 5 天的免费试用期限，继续使用需要购买正式授权或者延长试用。

注：试用授权与正式授权的软件功能完全一致，仅在使用期限上有限制。

序号号获取方式：

- 购买 ServHA 永久授权序列号
- 官网免费申请试用序列号（需注册官网账户）

授权方式分为在线授权与离线授权两种方式：

在线授权：当运行ServHA 集群配置工具的计算机能访问外网并能连入集群时，可以使用在线授权；

离线授权：任何情况下都能使用离线授权（需注册官网账户）。

## 在线授权

当集群中有一台服务器可以连接至外网时，可以执行在线授权步骤，在线授权只需填写您获取到的试用/永久序列号即可，具体步骤如下：

1、运行ServHA配置监控端,点击菜单栏“帮助”，在弹出的菜单里选择“授权管理”，如下图：



2、在授权界面中，点击“在线授权”，如下图：

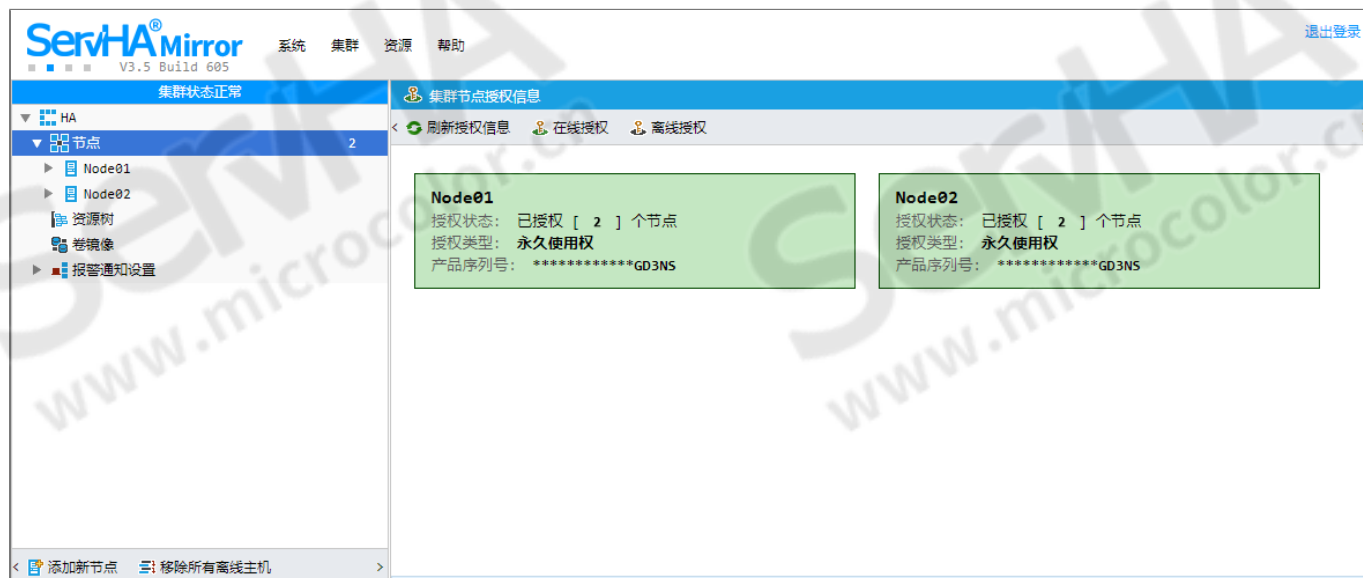




3、在弹出的新窗口中，填写获取到的序列号，点击“开始激活”按钮完成激活步骤。



4、激活成功后的界面如下图：



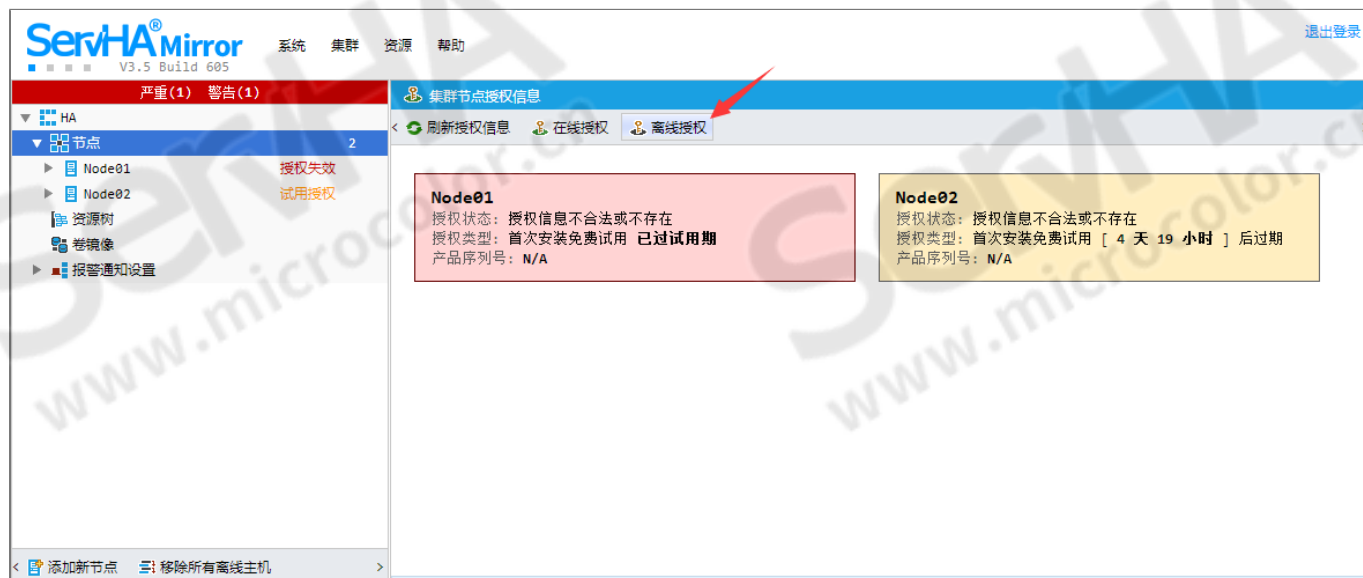
## 离线授权

当集群的所有服务器都不能访问外网时，只能执行离线授权步骤，您需要在ServHA管理工具的授权界面获取机器码，加上您获取到的软件序列号，在官网的个人中心生成离线授权码完成集群的离线授权，具体步骤如下：

1、运行ServHA管理工具,点击左上角“帮助”，在弹出的菜单里选择“授权管理”，如下图：



2、在授权界面中，点击“离线授权”，如下图：



3、在新窗口中，复制集群机器码信息，如下图：



4、用您注册的账户在官网登录后，进入“用户中心”，选择“离线授权”，填入您获得的序列号，输入验证码：

用户中心

我的信息

离线授权

试用申请

离线授权

请输入您的产品序列号，并点击下一步按钮!

序列号 (SN)

验证码

63w7 [换]

提交

5、粘贴机器码数据，输入验证码之后提交：

离线授权

请粘贴机器码数据到以下表单!

序列号 (SN)

机器码数据

验证码

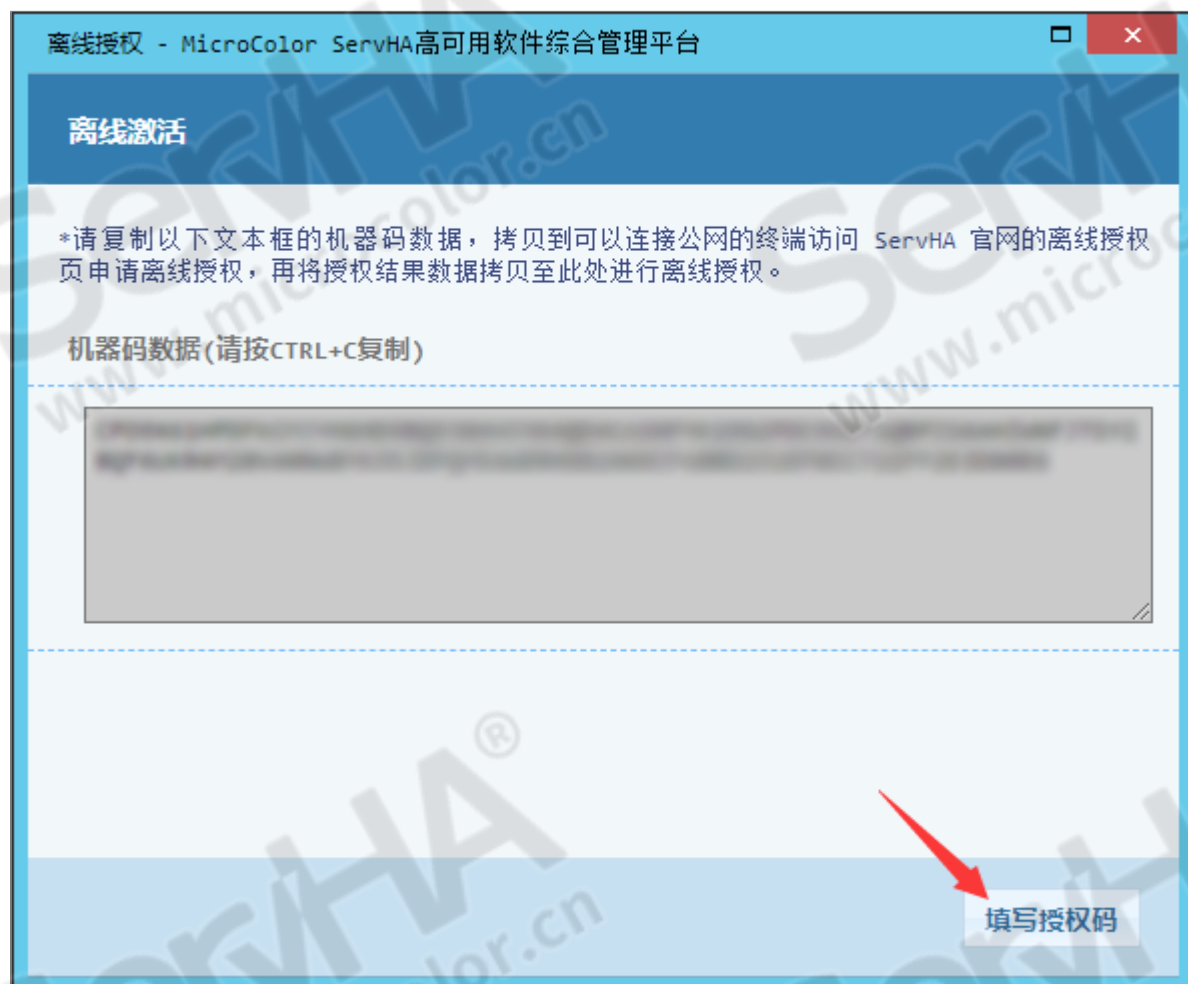
7wvd [换]

提交

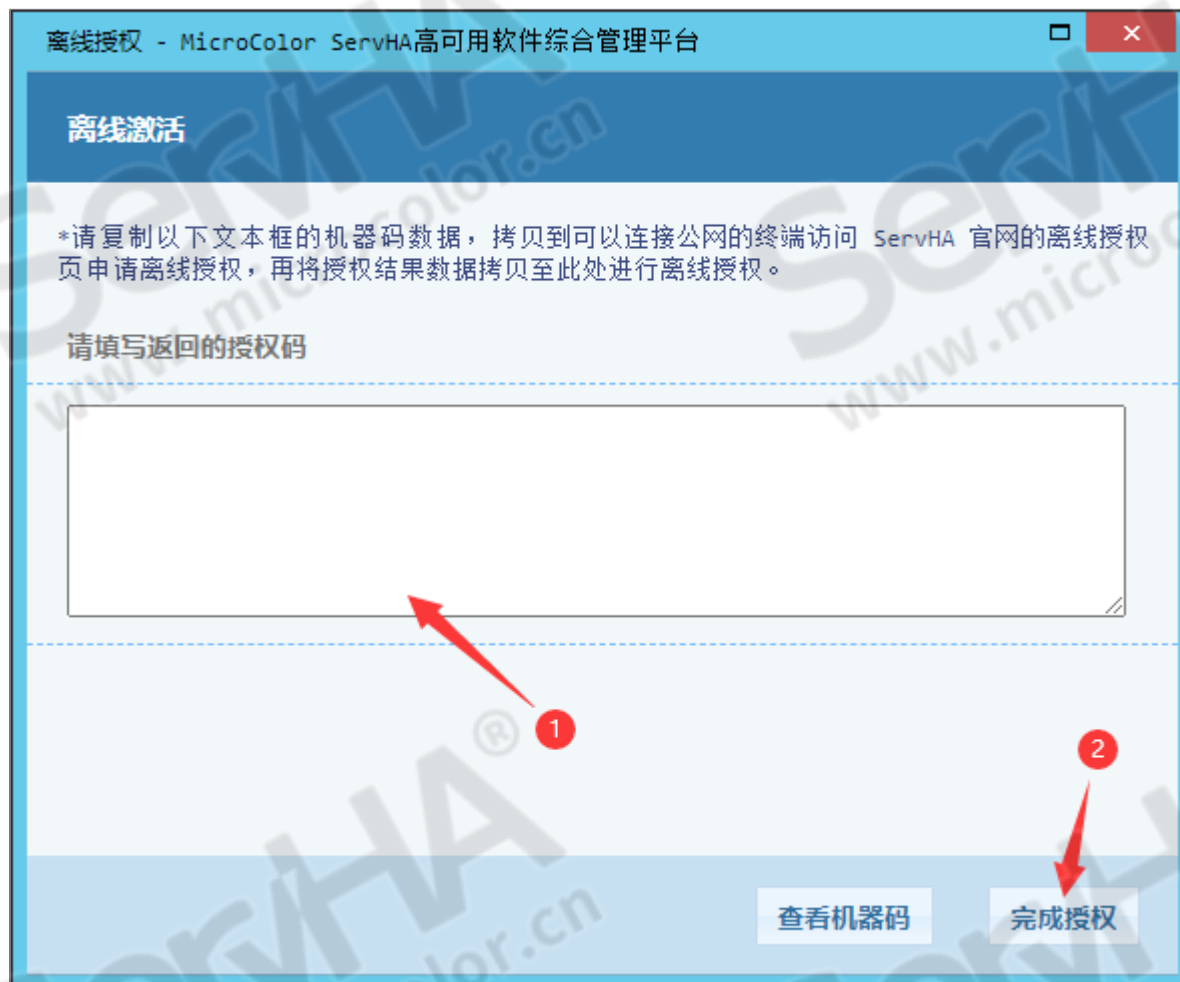
6、复制生成的授权码：

✓ 授权成功，请复制下面文本框的授权码粘贴到集群离线授权的授权码文本框中。

7、在ServHA监控端的离线授权界面里，点击“填写授权码”按钮，如下图：



8、在打开的页面文本框中输入授权码，点击“完成授权”，等待集群授权信息更新，如下图：



9、授权成功后如下图：



## 申请免费试用

1、使用注册的帐号登陆官网，登陆用户中心，点击左侧“试用申请”，如下图：





申请试用序列号	
<b>每个帐号每款产品可以免费申请1次试用序列号，请尽量完善您的注册信息（例如电话，公司等）！</b>	
选择产品	ServHA® Mirror 双机镜像软件 For Windows
应用行业	<input type="text"/>
应用项目	<input type="text"/>
备注	<input type="text"/>
验证码	<input type="text"/> 8k4w [换]
<input type="button" value="提交"/>	

2、提交完试用申请后，请耐心等待官方审核，通过审核后试用序列号将发送至您的注册邮箱里。

## 附件：注册官网账户

注：如果您已经注册了官网账户，则可以跳过本节所有步骤。

1、在 ServHA 官方网站 (www.microcolor.cn) 主页，点击“注册”，如下图：



2、填写您的用户信息，点击“提交”，如下图：

\* 类型

☒ 个人 ☐ 公司

\* 验证码

(必填) 验证码 

\* Email

(必填) Email

\* 昵称

(必填) 昵称

\* 密码

(必填) 密码

\* 确认密码

(必填) 确认密码

\* 姓名

(必填) 姓名

\* 公司名

(必填) 公司名

QQ

QQ

MSN

MSN

联系电话

联系电话

传真

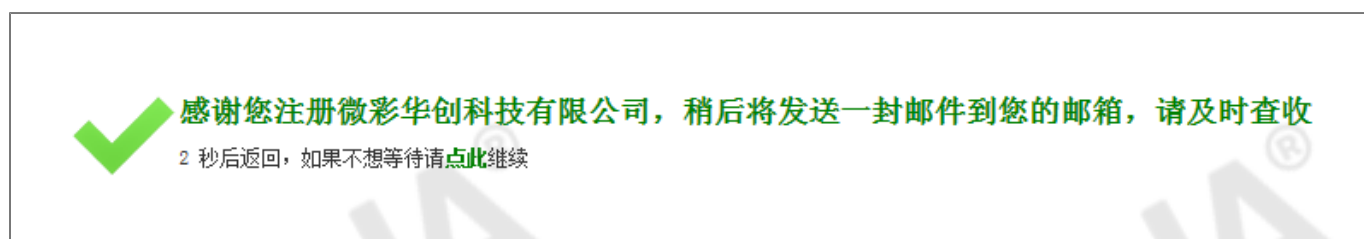
传真

手机

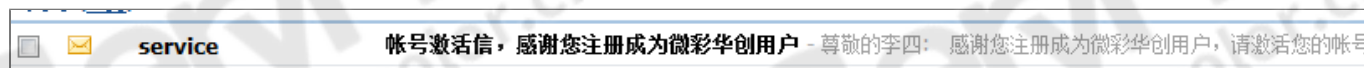
手机

[注册](#)

3、注册成功如下图：



4、在您刚才填写的注册邮箱中会收到一封激活邮件，如下图：



5、点击邮件内容中的激活账户连接，如下图：

尊敬的李四:

感谢您注册成为微彩华创用户, 请激活您的帐号, 帐号激活后您就可以使用本网站提供的所有功能, 免费领取软件试用序列号了。

请点击以下连接进行激活, 如果无法点击, 则请复制下面的连接地址到地址栏进行访问!

[http://www.microcolor.net/user/valid\\_email.aspx?](http://www.microcolor.net/user/valid_email.aspx?)

6、如果激活成功, 会跳转到如下页面:



**Email 校验成功, 感谢您注册!**

4 秒后返回, 如果不想等待请[点此继续](#)

## 概念定义

### 参考IP

参考IP是独立于两台服务器之外的其他设备的IP地址, 此IP地址需要两台服务器均能ping通, 并且要长期稳定存在 (一般为网关的IP)。

其存在的意义是当两台服务器之间的心跳全部断开后, 哪台服务器能ping通这个参考IP, 哪台服务器就会自认为是主机从而启动业务, ping不通的服务器将会自认为是备机从而停止业务, 如果网络环境中实在没有这样的IP地址的话, 可以使用127.0.0.1作为参考IP, 但绝不能使用两台服务器的实际IP和虚拟IP做参考IP!

### 虚拟IP

虚拟IP是需要提前在网络中预留的一个跟两台服务器的服务网络相同网段的一个IP地址, 这个IP地址是需要配置到热备软件中并由热备软件来管理的, 它可以随着业务的切换而切换, 客户端访问服务器只需要连这个虚拟IP就可以, 这样不管业务当前是运行在主机还是备机, 都能访问到当前正在对外提供服务的那台服务器。

虚拟IP的实现原理是给服务器的指定网卡多加一个IP地址, 其效力与实际IP一致且共存, 并继承该网卡的网关和DNS配置, 正常运行过程中, 服务网络中会同时存在两台服务器的IP和虚拟IP。虚拟IP切换时会先将主机的虚拟IP卸下, 然后再挂载到备机的网卡上, **并不是由热备软件来中转分发业务数据。**

使用虚拟IP需要满足下列的条件:

1. 虚拟IP必须和两台服务器服务网络的网段一致
2. 网关设备不能设置IP-MAC绑定
3. 应用程序如果需要监听指定IP地址的话, 需要指定监听虚拟IP

### 节点

参与热备的每台服务器在热备软件里都抽象成为了“节点”，若干个节点共同组成一个高可用集群。

## 心跳

心跳是一个高可用集群中节点之间相互通讯的逻辑链路，用于节点之间沟通状态和传送指令，通常是基于TCP或UDP协议的物理网络链路，只要节点之间能相互连通的物理网络链路，都可以是一组心跳。

一个集群中的心跳链路要尽可能多的冗余（包括物理上的链路），并且至少要有一组独立的物理网络链路（不受其他因素干扰，相对稳定）。

**心跳是高可用集群中最重要的元素**

## 资源

在热备软件中，所有由热备软件调度启停并监控的元素统称为资源。

资源的种类有很多，例如IPv4、IPv6这样的访问类资源，卷镜像、共享存储这样的存储类资源，还有服务、进程这样的程序类资源。

每个资源都必须拥有其对应元素的启动方式、停止方式和检测方式，其实除了自定义资源外，其他资源均已有现成的启停检测方式了，例如服务资源，热备软件可以通过系统的接口调用它的启、停，也能通过系统接口知道其是否在运行，而自定义资源则需要提供完整的信息才能让热备进行管理。

每个资源都有单独的高级属性：

- 启动前运行：在该资源启动前会先调用这里填的程序
- 启动后运行：在该资源启动后会调用这里填的程序
- 停止前运行：在该资源停止前会先调用这里填的程序
- 停止后运行：在该资源停止后会调用这里填的程序
- 停止失败时运行：当停止该资源失败时会调用这里填的程序

## 资源树

一个由若干个资源组成的集合称作为资源树。

所有资源可根据启停依赖顺序排列成一个树形结构，当资源树启动时，其内部的所有资源将按照从根到叶的顺序启动，当资源树停止时，其内部的所有资源将按照从叶到根的顺序停止。

## 镜像盘

镜像盘是服务器上的一个独立分区，例如Windows系统下的D盘、Linux系统下的/dev/sda2，使用卷镜像组件可以将两台服务器上的此分区内的所有数据进行实时同步。

该分区不能是操作系统所在的分区，也不能是热备软件所在分区，在Windows系统下，也不能将系统分页文件（虚拟内存）置于该分区中，也就是说，此分区仅用来存放数据和文件，**绝不能将操作系统所依赖的程序和文件放到该分区中。**



镜像分区的大小要足够一定时间的业务数据成长量，且两台服务器上该分区的大小要完全一致（精确到字节）或者备端略大一点，因为在新建立卷镜像的时候，主端会将自己镜像分区内的所有数据完整的同步给备端，备端镜像分区需要有大于等于主端镜像分区大小的空间来容纳这些数据。

在卷镜像建立好之后，两台服务器上的镜像分区是不能同时访问的，此为正常现象。

## 共享存储盘

共享存储盘是一个在支持共享访问的存储设备（一般为磁盘阵列或者iscsi设备）上单独划分的一个磁盘（LUN），并且同时映射给高可用集群中所有需要共享数据的服务器，其访问权是交由共享存储资源来控制的，热备软件会控制同时最多只能有一台服务器能操作该共享盘，以防止多设备同时写入数据造成的数据破坏。

## 仲裁盘

共享存储盘是一个在支持共享访问的存储设备（一般为磁盘阵列或者iscsi设备）上单独划分的一个磁盘（LUN），并且同时映射给高可用集群中所有服务器，该盘的空间不需要太大，100MB左右即可，过大会造成浪费。

仲裁盘类似于参考IP，是一种在心跳全部断开后的仲裁机制，不同的是仲裁盘使用的是数据链路，在共享存储的双机热备中，仲裁盘的重要性大于参考IP，是保护数据的最后一道也是最重要的一道屏障。

## 脑裂

脑裂一般分为两种：

集群脑裂：一般是在心跳全断并且所有仲裁无效的情况下才会发生，此时所有节点都会启动资源树对外提供业务，如果资源树中有镜像资源的话，在心跳恢复后会触发镜像脑裂，如果资源树中有共享存储资源的话，共享存储中的数据可能会被破坏。此类脑裂是可以通过合理的配置来避免的，请尽可能的保证如下配置：

- 使用共享存储资源的集群必须配置仲裁盘
- 配置合适的参考IP，尽量不要使用127.0.0.1来做参考IP
- 有时候IPv4或IPv6资源会防止脑裂的发生，因为当第二个节点想启动资源树的时候，若虚拟IP有冲突，会阻止资源树的启动
- 尽可能多的配置私网物理链路和心跳，降低心跳全断的几率
- 单心跳环境可以将资源树的切换策略设置为“仅最高优先级主机自动接管”来阻止自动的切换

镜像脑裂：是当两台服务器都在对方不在线的状态下往镜像盘中写入了数据才会触发的，该错误虽然是严重错误，但却能友好的改出，并不会对镜像内的数据造成毁灭性的破坏，最常见的触发情况是错误的开关机顺序导致的，集群内服务器的开关机操作要遵从“先关的后启”原则。镜像脑裂也可以通过合理的配置来尽可能的避免：

- 配置合适的参考IP，尽量不要使用127.0.0.1来做参考IP
- 尽可能多的配置私网物理链路和心跳，降低心跳全断的几率

- 单心跳环境可以将资源树的切换策略设置为“仅最高优先级主机自动接管”来阻止自动的切换
- 可以根据实际情况来配置脑裂自动恢复策略

## 维护指南

### 开关机顺序

在集群中的服务器，有时可能要对所有服务器进行开关机操作或者重启操作，为了防止脑裂（特别是镜像的脑裂）情况的发生，需要遵从“先关的后开”原则进行操作，例如：

- 关机时先关备机再关主机，开机时先开主机再开备机
- 重启时需要先重启一台服务器，待这台服务器重启完成后再重启下一台

### 默认密码

ServHA热备软件的默认管理密码是“1”

### 服务名

Linux版：

- servhad（核心模块服务）
- servwebd（配置监控代理模块服务）
- drbd（镜像模块服务）

### 查看日志

热备软件的日志在其安装目录下的log文件夹中：

- Windows版默认在C:\Program Files\ServHAMirror\log目录下
- Linux版默认在/usr/share/servha/log目录下

由于热备的日志仅记录自己产生的日志，应用方面的日志请查看应用本身生成的日志或查看系统日志。

### 更改IP

有时候可能要更改集群中服务器的IP地址以适配新的网络环境，此时分为以下几种情况：

- 更改虚拟IP

直接编辑对应的IPv4或IPv6资源，将IP改为新的IP地址即可

- 更改服务器的实际IP（有私网链路）



此情况一般不会涉及到私网链路IP的变更，因为私网链路不会对公网链路产生任何影响，所以不用更改私网IP和对应的心跳信息

1、先在操作系统里将实际IP更改

2、登入热备软件配置监控端，依次编辑心跳、参考IP、虚拟IP资源，如果有镜像使用了修改IP的链路还需要修改镜像

- 更改服务器的实际IP（无私网链路或私网链路也需要改IP）

此情况最为复杂，但也可以只通过修改来完成配置

1、先在操作系统里将实际IP更改

2、登入热备软件配置监控端，先编辑心跳，此时会提示其他服务器失联，让输入新的IP地址，根据节点名称输入新的IP地址后即可列出对应节点的网卡列表，选择对应的新IP地址的网卡保存即可

3、再依次编辑参考IP、虚拟IP资源，如果有镜像使用了修改IP的链路还需要修改镜像

## 更换节点

在实际使用过程中，难免的会碰到需要更换集群中服务器的情况，更换单节点的操作步骤大致如下：

实际操作过程根据环境的不同可能有所不同，本手册只是提供主要的思路

名词定义：老服务器——需要被更换的服务器，新服务器——要新换上去的服务器，现存服务器——不需要更换的服务器

1. 将老服务器断电、断网、下架
2. 安装新服务器上的操作系统、应用软件、热备软件

要与老服务器完全一致，包括但不限于版本、位数、安装路径、权限等

如果老服务器上有热备需要用到的脚本，也需要一并转移到新服务器上

3. 如果新服务器需要连接共享存储设备，请先确保不会有任何软件会往存储中写入数据！
4. 新服务器上架、联网、加电开机
5. 将老服务器上的IP配置到新服务器上
6. 登入现存服务器的HA配置监控端（之后的操作都在这个配置监控端里操作）
7. 先记录一下左侧【资源树】下哪些资源树绑定了老服务器的节点
8. 在左侧【节点】处右键--移除所有离线主机
9. 将所有资源树下的现存节点处右键--设置维护模式
10. 再在左侧【节点】处右键--添加新节点，走完向导
11. 若新节点参与了卷镜像、目录镜像，需要先删除不完整的镜像后重新建立（注：重建linux系统下的卷镜像会导致业务中断，请先预留好窗口时间）
12. 在左侧【资源树】下的所有之前记录的绑定了老服务器的资源树上依次右键--绑定节点，并走完向导
13. 授权

## 更新应用程序

当被做成资源的程序需要更新时，难免会遇到需要暂时单独停止一下该资源的运行的情况，而热备软件可能会在该程序停止的时候自动运行起来或者切换到备机，反倒造成了对更新操作的障碍，此时，只需要在热备配置监控端中，在该程序对应的资源图标上右键--设置维护模式，热备软件就不会管这个资源的状态了，再去停止对应的程序更新即可。

在更新操作完成后记得把对应资源的维护模式取消掉

如果更新的部分全在同步路径中，那只需在一台服务器上更新即可，否则需要在每台服务器上都进行一遍更新操作

## 脑裂恢复

脑裂恢复一般是指镜像脑裂的恢复。

如果您选择的是脑裂后自动处理，则无需人工干预，集群将根据您选择的策略自动进行处理。如果您选择的是脑裂后手动处理，操作步骤如下：

在配置监控端中选择脑裂的镜像，点击工具栏上“脑裂恢复”按钮，在弹出的“脑裂恢复”窗口中根据提示信息选择恢复即可。



比对两台机器的差异数据，选择合适的一端，点击“恢复脑裂”：

注意：脑裂后会造成两节点数据不一致，选择脑裂恢复必须丢弃其中一个节点的差异数据，请谨慎选择保留数据的一端。

为了方便用户确定镜像脑裂后的数据保留方，在镜像脑裂的状态下资源树也可以进行切换，但是数据不会同步到对机，您可以通过资源切换/建立快照，备份或者验证数据有效的一方，最后选择脑裂恢复方向，避免关键数据的丢失。



## 切换

### 手动切换

概述：手动切换其实是以手动的方式将资源树运行在之前未启动的一台服务器上。

#### 切换条件

1. 需要启动资源树的节点在线；
2. 需要启动资源树的节点上没有出现“决断错误”（A）；

#### 注意事项：

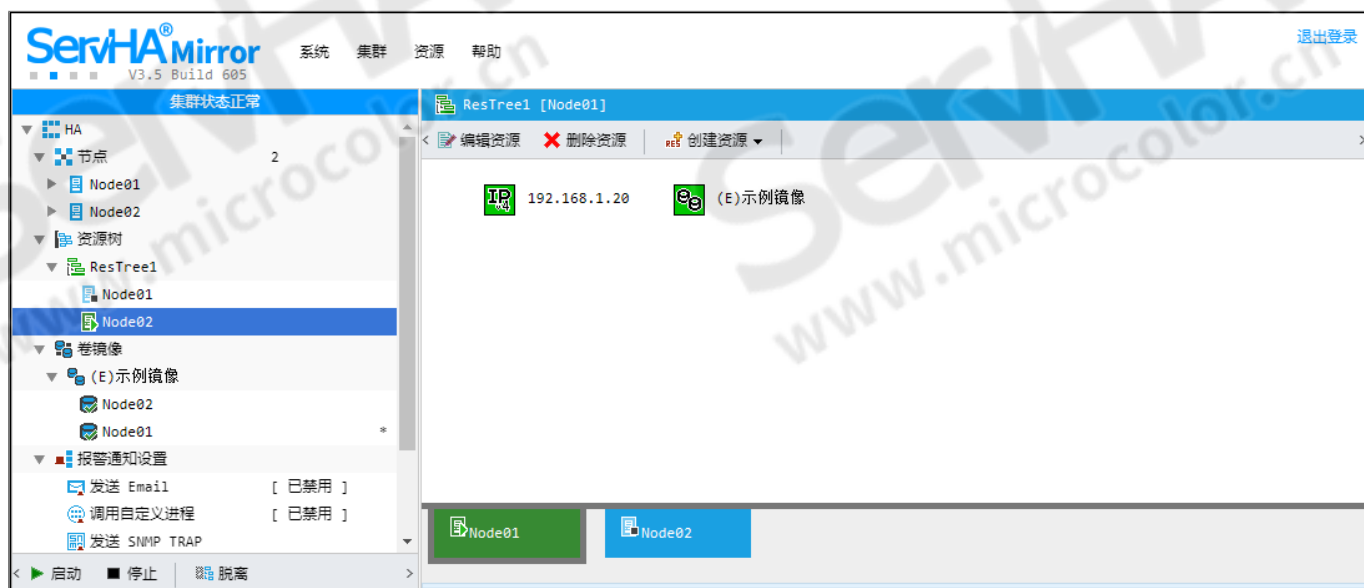
1. 切换过程中会先停止之前运行着资源树的服务器上的业务，请先预留好维护窗口时间；
2. 若切换发生失败，资源树不会自动切回至之前运行的服务器并等待手工处理；

#### 操作指南：

在热备界面左侧资源树下，单击要切换到的节点（蓝色的节点为当前的备机），点击启动按钮。



等待切换完毕。



当操作的节点变成主节点（绿色）后即为切换成功。

## 自动切换

概述：在当前主机发生热备软件能检测到的故障时，有可能会根据策略触发自动切换，此操作一般不需要人员干预。

触发情况：

- 主备机之间通过心跳链路无法联系到对机，且主机不能联通参考IP，有仲裁盘的情况下主机机还需不能联通仲裁盘，比如主机断电、关机、蓝屏等情况；
- 主机上的某资源不处于“维护模式”且出现了决断错误（A）并且异常停止（例如主机公网网线脱落、主机镜像盘/共享存储掉盘等情况）；
- 主机上的某资源不处于“维护模式”并且频繁停止（默认配置下是1分钟内异常停止超过3次）；

备机接管条件：

- 备机在线且授权未过期；
- 资源树的“启动模式”必须为“自动”，“切换策略”为“按主机优先级顺序自动接管”；
- 在心跳全断的情况下，备机需要能联通参考IP，有仲裁盘的情况下还需备机能联通仲裁盘；
- 备机要启动的虚拟IP地址未被其他设备占用；
- 备机的资源树下的资源没有出现决断错误（A）；
- 备机的镜像处于正常（实时同步）状态；

注意事项：

- 若自动切换出现了失败，资源树有可能还会切回来；

## 清除热备配置

\*该操作将会彻底清除所操作节点上的热备配置

Linux操作:

```
#在终端命令行中以root身份运行  
/usr/share/servha/op rm_clearup
```

Windows操作:

```
#使用管理员身份运行cmd  
"C:\Program Files\ServHAMirror\MCNTHA.exe" rm_clearup
```

## 软连接

软链接又叫符号链接,是操作系统自带的功能,它可以创建一个符号文件,这个文件包含了另一个文件的路径名。可以是任意文件或目录,可以链接不同文件系统的文件。

在对符号文件进行读或写操作的时候,系统会自动把该操作转换为对源文件的操作,但删除链接文件时,系统仅仅删除链接文件,而不删除源文件本身。

链接文件甚至可以链接不存在的文件,这就产生一般称之为“断链”的现象,链接文件甚至可以循环链接自己。类似于编程语言中的递归。

软链接文件只是其源文件的一个标记,当删除了源文件后,链接文件不能独立存在,虽然仍保留文件名,但却不能查看软链接文件的内容了。

Linux系统均支持软链接,而Windows系统仅在Windows Vista/Server 2008及以上的版本才支持。

在Linux系统上建立软连接:

```
#在终端命令行中运行  
ln -s "源路径" "符号路径"
```

在Windows系统上建立软连接:

```
#使用管理员身份运行cmd  
mklink /D "符号路径" "源路径"
```

## 无法访问共享存储盘、镜像盘、同步目录

镜像盘、共享存储盘、同步目录同时最多只能有一个节点能进行访问,这个是正常现象,如果所有节点上都无法访问,则需要检查其对应的资源所在的资源树是否有出现错误导致其无法启动。



## 自动登录系统桌面

某些应用程序的运行是需要有桌面环境的，其对应资源也必须勾选“运行在桌面上”的选项，此时该资源的启动先决条件将会对桌面环境进行检查，在服务器重启后默认是不会自动登录桌面的，这会导致需要桌面环境的资源出现决断失败 (A) 错误，影响资源树的启动。

解决方案就是配置操作系统的自动登录。

## Windows系统的自动登录

**注意：**需要至少保留一个已登录用户不注销，否则，带窗口的程序依旧无法运行。使用WIN+L锁定并返回登录界面不会影响窗口程序的运行。

打开注册表，选择

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\Winlogon]项，添加如下键（字符串值）：

```
"AutoAdminLogon"="1"
```

```
"DefaultUserName"="默认登录的用户名，一般为Administrator"
```

```
"DefaultPassword"="该用户的密码"
```

处于域环境的话再加一个键：

```
"DefaultDomainName"="所处域的名称"
```

如果出于安全需要，需要在Windows自动登录后自动返回登录界面，可以将下面的脚本保存为.vbs文件，并放入到Windows自动启动项中（Windows XP/2003一般在“C:\Documents and Settings\用户名\开始菜单\程序\启动”下，Windows Server 2008及以上一般在“C:\Users\用户名\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup”下）：

```
Dim oSys, Syscmd
Set oSys = CreateObject("WScript.Shell")
Syscmd = "%windir%\SYSTEM32\rundll32.exe user32.dll, LockWorkStation"
oSys.Run Syscmd, 0, False
```

## Linux系统的自动登录

由于Linux的版本与桌面环境众多，以下仅举例，具体环境上的配置请自行查阅

Redhat7.6系统下设置

修改/etc/gdm/custom.conf文件的[daemon]项配置，重启后即可生效

```
[daemon]
AutomaticLoginEnable=True
```



```
AutomaticLogin=root
```

## Ubuntu 18.04系统下设置

### 1. 允许使用root用户登录桌面

Ubuntu默认不允许使用root用户登录桌面的，首先需要进行如下配置

编辑/usr/share/lightdm/lightdm.conf.d/50-ubuntu.conf将该文件的配置修改成如下

```
[Seat:*]
user-session=ubuntu
greeter-show-manual-login=true #在登录时允许用户自己输入用户名和密码
```

Ubuntu中的root用户默认是没有设置密码的，所以还需要设置root密码

```
sudo passwd root #在当前桌面登录的用户下执行这条命令修改root密码
[sudo] user 的密码: #输入当前登录用户用户user的密码
输入新的 UNIX 密码: #输入给root密码
重新输入新的 UNIX 密码: #再次输入root密码
passwd: 已成功更新密码
```

### 2. 允许root开机自动登录桌面

再次编辑/usr/share/lightdm/lightdm.conf.d/50-ubuntu.conf，在该文件末尾添加如下两行

```
allow-guests=false #禁止访客用户登录
Autologin-user=root #在登录时允许用户自己输入用户名和密码
```

编辑完成之后，还需要修改图形桌面配置，编辑文件/etc/gdm3/custom.conf，找到[daemon]项，将该项的配置修改至如下

```
[daemon]
AutomaticLoginEnable=true #启用自动登录
AutomaticLogin=root #指定root用户为自动登录用户
```

### 3. 由于Ubuntu默认是禁止使用root用户进行ssh远程登录终端访问的，所以如果需要使用root登录ssh的话则需要再文件中增加如下配置

```
PermitRootLogin yes
```

### 4. 重启电脑生效