

弹性伸缩

常见问题

常见问题

进行弹性伸缩的ECS实例如何保证配置环境的一致，或者如何把之前的系统环境直接添加到新创建的弹性伸缩实例中？

创建弹性伸缩配置 ECS 实例时，您可以通过 ECS 自定义镜像模板来创建实例。在 ECS 实例运行过程中，如果需要做系统内部数据同步，建议您自定义安装 rsync 进行同步。

为什么弹出的实例中 /etc/hosts 新增的 127.0.0.1 被重置清除了？

目前用户在镜像中写了 /etc/hosts 并增加了自己的内容后，通过该镜像创建的自定义镜像来创建机器时，会还原到系统默认设置，所以会被清除。如果需要保留设置，您可以尝试在 rc.local 中添加相关脚本代码，检测 /etc/hosts 中是否存在相关信息，若不存在则进行添加。

如果弹性伸缩是自动创建的，并且量是不固定的，如何使用镜像市场的镜像来保证需求实例能正常弹出？

如果您需要弹出 n 台同类型的镜像，您需要提前购买n台镜像市场的镜像。

镜像市场的镜像是否支持批量购买？

暂不支持批量购买。

如果之前使用的镜像市场的镜像，已不存在该商品（如：jxsc000010），如何保证已经设置的弹性伸缩组的机器正常弹出？

此情况建议您选择镜像市场中可以替代的镜像进行替代。

1 个 product code 能否支持不同 region 的镜像？

支持，前提是该地域 region 已经支持该商品镜像。

购买了 100 个 product code 同样值的镜像，是否支持在所有的地域可用？

目前镜像市场的镜像已经具备 region 属性，请您购买需要使用的地域镜像。

弹性伸缩是否能够自动升降ECS的CPU、内存和带宽？

弹性伸缩是根据用户的业务需求和策略，自动调整其弹性计算资源的管理服务。其能够在业务增长时自动增加ECS实例，并在业务下降时自动减少ECS实例。弹性伸缩目前还不能支持**纵向扩展**，即弹性伸缩暂时无法自动升降ECS的CPU、内存和带宽。

1. 弹性伸缩自动创建的实例如何查看密码并进行登录？

由于弹性伸缩自动创建的实例没有统一的登录密码，在 Linux 环境下，推荐您通过设置 **公私钥** 认证的方式来进行 SSH 免密码登录。

如果您不需要通过设置 **公私钥** 认证的方式来进行 SSH 免密码登录，则需要在控制台重置密码（重启生效）才能登录。

2. 弹性伸缩的产生的服务器的密码为什么和我自定义镜像中的密码不一致？

创建云主机的时候，主机密码都不是自定义镜像中的密码。为了保证密码的安全性，推荐您通过设置 **公私钥** 认证的方式来进行 SSH 免密码登录。

如果您不需要通过设置 **公私钥** 认证的方式来进行 SSH 免密码登录，则需要在控制台重置密码（重启生效）才能登录。

3. 使用自定义镜像弹出的 Linux 系统实例，如何进行免密码登录管理？

您可以通过设置 **公私钥** 认证的方式来进行 SSH 登录。

在需要建立自定义镜像的 ECS Server 端实例上建立公钥和私钥。

```
1 #ssh-keygen -t dsa -f ~/.ssh/idc (idc名字可以自定义：例如aaa、bbb等)
2 Generating public/private dsa key pair.
3 Enter file in which to save the key (/root/.ssh/idc): ← 钥匙的文件名，保持默认直接回车
4 Enter passphrase (empty for no passphrase): ← 输入口令，如想无口令认证直接回车
5 Enter same passphrase again:
6 Your identification has been saved in /root/.ssh/idc.
7 Your public key has been saved in /root/.ssh/idc.pub.
8 The key fingerprint is:
9 1e:16:9e:d5:d2:6a:b6:e2:a0:de:14:b2:eb:d1:5e:03 root@localhost.localdomain
10 #cd ~/.ssh
11 #ls -l
12 total 8
13 -rw----- 1 root root 668 Feb 26 14:01 idc ← 确认私钥已被建立
14 -rw-r--r-- 1 root root 616 Feb 26 14:01 idc.pub ← 确认公钥已被建立
15 #cat idc.pub >>authorized_keys ← 公钥内容输出到相应文件中
16 #chmod 400 authorized_keys ← 将新建的公钥文件属性设置为400
```

将 ECS 实例上的 idc.pub 拷贝到客户端。

删除 Server 端的公钥。

```
1 | #rm -f ~/.ssh/idc.pub
```

← 删除原来的公钥文件

修改 Server 端的 ssh 配置文件。

```
1 | #cd /etc/ssh
2 | #cp sshd_config sshd_configback
3 | # vi sshd_config
4 | PubkeyAuthentication yes
5 | AuthorizedKeysFile .ssh/authorized_keys
6 | PasswordAuthentication no
```

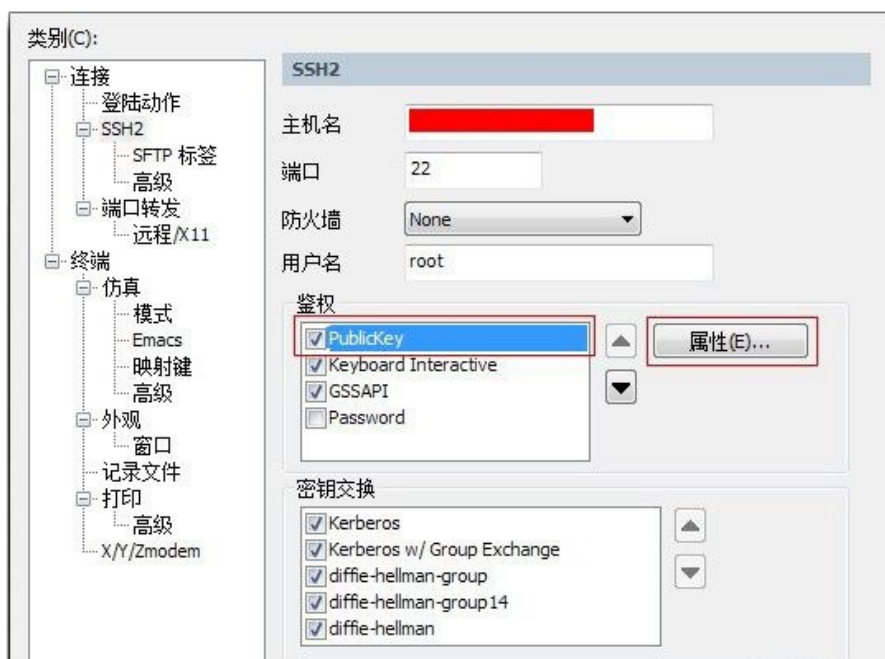
← 进入 ssh 配置文件
← 备份 ssh 配置文件
← 编辑配置文件修改一下参数信息
/* 启用 PublicKey 认证
/* PublicKey 文件路径
/* 禁止密码验证登录

配合客户端 Client 软件进行设置。

以 SecureCRT 配置为例：

选择对应的远程连接信息。

右键单击 **属性** 选项，出现如下图所示页面。



选择 SSH2。

取消勾选 **Password**，勾选 **PublicKey**。

单击右边的 **属性** 按钮，出现如下图所示页面。



选择 **使用会话公钥设置**。

选择 **使用身份或证书文件**，并选中 **idc.pub**（刚从服务器上拷贝下来的公钥和私钥文件）。

遇到弹性伸缩问题，应该提交什么信息？

您可以提供您的弹性伸缩组的活动 ID（activity id）或者相关日志，方便我们进行排查。

弹性伸缩创建伸缩配置的时候为什么不能选择已经购买过的云服务器？

弹性伸缩支持添加已有 ECS 实例，需要满足如下条件：

- 加入的ECS实例必须与伸缩组在同一个 region。
- 加入的ECS实例的状态必须是 **运行中** 状态。
- 加入的ECS实例不能已加入到其它伸缩组中。

弹性伸缩是否支持已有的包年包月实例添加？

目前弹性伸缩默认自动创建按量付费实例，但是同时也支持用户已有的包年包月和按量付费的实例添加。

同一台 ECS 实例可以添加进入不同的伸缩组吗？

目前无法支持。

弹性伸缩组是否可以添加不同规格的 ECS 实例？

每个伸缩组中只能设置一种伸缩配置的规格（CPU 和 Memory）。不过您可以设置多个伸缩组，在每个伸缩组设置不同的配置。

弹性伸缩服务内，一个伸缩组的最大实例数只能为 100 个，是否可以调高最大实例数？

您可以提交工单申请更高配额，我们的工作人员会在工作日时间内协助您添加，周末节假日顺延。

弹性伸缩服务内，伸缩组的 ECS 实例是否可以为 8 核，16 核配置？

可以。关于大配置的 ECS 实例，您可以在 ECS 实例的按量付费页面，提交申请更高配置，我们的工作人员会在工作日时间内协助您添加，周末节假日顺延。

将 ECS 实例移出伸缩组并释放，ECS 实例上的数据能否保存？

ECS 实例移除之后会释放，数据无法保留。

由于弹性伸缩会自动释放 ECS 实例，所以用于弹性伸缩的 ECS 实例不可以保存应用的状态信息（如 session）和相关数据（如数据库、日志等）。如果应用中需要保存状态信息，可以考虑把状态信息保存到独立的状态服务器、数据库（如 RDS）及集中日志存储（如 Log Service）

通过 API 接口调用 DisableScalingGroup 方法，会不会导致已经自动加入的 ECS 实例被释放？

执行 DisableScalingGroup 时，伸缩组内自动创建的按量付费实例不会被自动释放。

配置的伸缩组自动增加 ECS 实例时，ECS实例会自动添加到 RDS 和 OCS 的 IP 白名单吗？

弹性伸缩支持 RDS 访问白名单，在增加或减少 ECS 实例时，自动向 RDS 访问白名单中添加或移出该 ECS 实例的 IP。但目前不支持 OCS 访问白名单。

添加到伸缩组的 ECS 实例是否支持“纵向扩展”，支持CPU，内存升级？

弹性伸缩目前还不能支持 纵向扩展，即弹性伸缩暂时无法自动升降 ECS 的 CPU、内存和带宽。

弹性伸缩是否支持设置周期性的重复任务，在指定的时间内重复性地执行

？

您可以在弹性伸缩控制台的 **自动触发任务管理 > 定时任务** 中设置定时任务，如果未设置重复周期，则按指定的日期和时间执行一次。如果设置了重复周期，任务按照指定的时间点重复执行。

如何保证手工添加的 ECS 实例不被移出伸缩组？

假设您需要添加 N 台包年包月的 ECS 实例加入伸缩组，并且希望这些 ECS 实例不会自动被移出，那么您需要做以下两个配置（同时满足）：

- 将最小实例数（MinSize）设置为等于或者大于 N。
- 将移出策略（RemovalPolicy）的第一条挑选规则设置为 **最早伸缩配置对应的实例**（OldestScalingConfiguration）。

根据弹性伸缩的规则，手工添加的 ECS 实例不会对应任何伸缩配置（因为它们不是通过伸缩配置创建的），所以弹性伸缩会先挑选自动创建的 ECS 实例移出伸缩组，只有当自动创建的 ECS 实例移出伸缩组完毕，才会挑选手工添加的 ECS 实例移出伸缩组。

请注意自动创建的 ECS 实例，被移出伸缩组会自动释放；手工添加的 ECS 实例，移出伸缩组不会被释放。

注意：以上是正常的健康逻辑，如果您停止了这些手工添加的 ECS 实例，弹性伸缩会认为他们不健康，并将它们移出伸缩组，因为弹性伸缩需要保证在伸缩组里的 ECS 实例是健康的。

弹性伸缩创建服务器后，新创建的实例会自动加入到负载均衡中吗？

如果在伸缩组中指定了负载均衡实例，伸缩组会自动将加入伸缩组的ECS实例加入到指定的负载均衡实例中。

弹性伸缩在添加伸缩组时，能绑定多个负载均衡实例吗？

一个伸缩组默认只能绑定一个负载均衡实例，如果需要使用多个负载均衡实例，您可以提交工单申请更高配额，我们的工作人员会在工作日时间内协助您添加，周末节假日顺延。

弹性伸缩新增的ECS实例是否可以添加进入多个负载均衡中？

可以，目前仅支持一个负载均衡，多个负载均衡需要提交工单申请。

是否可以修改加入到弹性伸缩组负载均衡实例中的权重？

目前弹性伸缩负载均衡下ECS实例的默认权重为50。正常情况下，弹性伸缩组后端ECS实例所承载的业务和ECS实例的规格应该都是一样的，所以默认权重也应该是一样的。而负载均衡的权重分配是按比例而非按数字计算的，即，假设后端有两台ECS实例，权重设置成50和50（比例是1:1）的效果，和权重设置成100和100（比例也是1:1）是一样的。

我现在有公网的负载均衡，在创建伸缩配置时ECS实例是否需要公网带宽？

创建伸缩配置时，ECS实例可以不选择公网带宽。但是为了方便管理ECS实例，建议您购买至少1Mbps的ECS带宽。

弹性伸缩一定要搭配负载均衡、云监控、RDS才能使用吗？

弹性伸缩是一个开放的灵活扩展收缩平台。弹性伸缩可以单独扩展和收缩ECS实例，既可以搭配负载均衡和RDS一起部署，也可以不搭配负载均衡和RDS。弹性伸缩支持通过云监控触发任务扩展和收缩ECS实例，也可以通过弹性伸缩的Open API对接客户自己的监控系统，客户可以通过自己的监控系统，触发弹性伸缩的伸缩活动。

如何判断弹性伸缩中实例的可用性？

如果弹性伸缩在伸缩组里配置了负载均衡，负载均衡检查您后端的ECS端口正常之后，才会将请求转发给新的实例。

弹性伸缩中的告警触发条件有哪些？

可以针对CPU负载、内存利用率、系统平均负载、外网和内网出入流量等进行监测告警，自动增加或减少ECS实例。

弹性伸缩是否可以根据云监控中自定义报警项进行动态伸缩？

目前不支持根据自定义监控进行动态伸缩。

伸缩组内创建的ECS应用部署如何自动化？

在弹性伸缩自动创建ECS实例后，如果想要自动安装、更新程序和自动加载代码，需要您在自定义镜像中存放执行脚本，并设置命令使操作系统启动后自动执行该脚本。

注意：由于 CentOS 6 及以下系统采用 system V init 作为初始化进程，但 CentOS 7 采用了systemd来做初始化进程，它的工作原理和之前的 system V init 区别较大。以下分别针对 CentOS 6 以及 CentOS 7 进行说明。

CentOS 6 及以下系统

创建以下shell测试脚本：

```
#。/bin/sh
# chkconfig: 6 10 90
```

```
# description: Test Service
echo "hello world!"
```

针对上述脚本中的# chkconfig: 6 10 90，说明如下：

其中6默认启动级别，级别有 0 ~ 6 共 7 个级别。

- 等级0表示：关机
- 等级1表示：单用户模式
- 等级2表示：无网络连接的多用户命令行模式
- 等级3表示：有网络连接的多用户命令行模式
- 等级4表示：不可用
- 等级5表示：带图形界面的多用户模式
- 等级6表示：重新启动
- 10是启动优先级，90是停止优先级，优先级范围是 0 ~ 100，数字越大，优先级越低。

将测试文件放在/etc/rc.d/init.d/目录下，执行chkconfig --level 6 test on。

说明：以后每次开机启动都会执行该测试脚本。

示例

以脚本安装Phpwind为例，您可以将phpwind安装程序放到脚本中执行（数据库密码仍需要您自己输入），如下仅是示例，您可以按照示例进行修改。

```
cd /tmp

echo "phpwind"

yum install -y \
unzip \
wget \
httpd \
php \
php-fpm \
php-mysql \
php-mbstring \
php-xml \
php-gd \
php-pear \
php-devel
chkconfig php-fpm on \
&& chkconfig httpd on
wget http://pwfiles.oss-cn-hangzhou.aliyuncs.com/com/soft/phpwind_v9.0_utf8.zip \
&& unzip -d pw phpwind_v9.0_utf8.zip \
&& mv pw/phpwind_v9.0_utf8/upload/* /var/www/html \
&& wget http://ess.oss-cn-hangzhou.aliyuncs.com/ossupload_utf8.zip -O ossupload_utf8.zip \
&& unzip -d ossupload ossupload_utf8.zip \
&& /bin/cp -rf ossupload/ossupload_utf8/* /var/www/html/src/extensions/ \
&& chown -R apache:apache /var/www/html
```

```
service httpd start && service php-fpm start
echo "安装云监控"
wget http://update2.aegis.aliyun.com/download/quartz_install.sh
chmod +x quartz_install.sh
bash quartz_install.sh

echo "安装完成"
```

CentOS 7 系统

CentOS 7 采用了systemd来做初始化进程，工作原理与 system V init 区别较大。假设您已创建好脚本，并且测试其运行无误。参照如下步骤，使您使用systemd时，在系统关闭状态仍能运行脚本。

在/etc/systemd/system下创建一个文件 run-script-when-shutdown.service，文件内容如下（请把ExecStop变量的值，修改为您需要运行脚本的绝对路径）：

```
[Unit]
Description=service to run script when shutdown
After=syslog.target network.target

[Service]
Type=simple
ExecStart=/bin/true
ExecStop=/path/to/script/to/run
RemainAfterExit=yes

[Install]
WantedBy=default.target
```

执行如下命令，使能新创建的服务：

```
systemctl enable run-script-when-shutdown
systemd start run-script-when-shutdown
```

说明：

- 为了使当前服务立即生效，需另外执行restart命令。
- 您可以配置run-script-when-shutdown运行固定的脚本。需要的时候，相关人员可以修改这个固定的脚本，使其更加灵活实用。

无需运行上述服务时，请执行如下命令：

```
systemctl disable run-script-when-shutdown
```