

阿里云 云服务器 ECS

部署与运维

文档版本：20190223

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或惩罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	禁止： 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	警告： 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	说明： 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 确定。
courier 字体	命令。	执行 cd /d C:/windows 命令，进入Windows系统文件夹。
##	表示参数、变量。	bae log list --instanceid <i>Instance_ID</i>
[]或者[a b]]	表示可选项，至多选择一个。	ipconfig [-all] [-t]
{}或者{a b} }	表示必选项，至多选择一个。	switch {stand slave}

目录

法律声明.....	I
通用约定.....	I
1 部署集.....	1
1.1 部署集概述.....	1
1.2 创建部署集.....	2
1.3 在部署集内创建实例.....	3
1.4 管理部署集.....	4
2 云助手.....	6
2.1 云助手概述.....	6
2.2 配置云助手客户端.....	10
2.3 云助手自动化部署实践.....	13
3 系统事件.....	18
3.1 系统事件概述.....	18
3.2 实例自动恢复.....	22

1 部署集

1.1 部署集概述

部署集是控制实例分布的策略，使您能在创建实例的时候就设计容灾能力和可用性。

部署策略

您可以使用部署集将业务涉及到的 ECS 实例分散部署在不同的物理服务器上，以此保证业务的高可用性和底层容灾能力。在部署集内创建实例时，我们会根据您事先设置的部署策略在指定地域下分散启动实例。如果您没有为实例设定部署集，我们则会尽可能在不同的物理机上启动实例，保障服务可用性。

目前，部署集支持高可用策略：

- 采用高可用策略后，部署集内所有 ECS 实例会在指定地域内严格分散在不同的物理服务器上。适用于需要将几台 ECS 实例相互隔离的应用架构，大幅降低服务不可用的几率。
- 高可用策略下，遇到地域内供货紧缺时，可能无法创建 ECS 实例。设置了 [停机不收费](#) 的按量付费实例会造成重启失败。一般情况下，您可以等待一段时间后重试创建或重启操作。

部署示例

下图是利用部署集能力提升业务可靠性的典型示例，您的四台 ECS 实例分布在四台不同的物理宿主机上。

如果您需要部署集内的实例能实现低延时通信，建议您保持实例的网络类型一致性。例如，创建实例时为这些实例选择同一个 VPC。

计费详情

使用部署集不会收取服务费用，您的消费详情只包括您创建和使用的实例、磁盘、快照、镜像和带宽等服务。更多详情，请参阅 [计费概述](#)。

注意事项

在使用部署集之前，您需要注意：

- 部署集之间不支持相互合并。
- 部署集内不能创建 [抢占式实例](#)。
- 部署集不支持创建 [专有宿主机](#)。

- 在部署集内创建实例时，一个可用区内最多能创建 7 台 ECS 实例，一个阿里云地域下能创建的 ECS 实例数量则为 7^* （可用区数量）。
- 部署集现阶段能创建的实例规格族仅支持：c5、d1、d1ne、g5、hfc5、hfg5、i2、ic5、r5、se1ne、sn1ne 和 sn2ne。更多有关实例规格及其性能的信息，请参阅 [#unique_9](#)。
- 在部署集内创建实例或者重启停机不收费的按量付费实例时，供货紧缺依然会导致请求失败。

相关操作

控制台操作

- [创建部署集](#)
- [在部署集中创建ECS实例](#)
- [修改实例所在的部署集](#)
- [管理部署集](#)

API操作

- 创建部署集：[CreateDeploymentSet](#)
- 将一台实例加入一个部署集，或者将实例从一个部署集调整到另外一个部署集：[ModifyInstanceDeployment](#)
- 修改部署集属性：[ModifyDeploymentSetAttributes](#)
- 查询部署集列表：[DescribeDeploymentSets](#)
- 删除部署集：[DeleteDeploymentSet](#)

1.2 创建部署集

本文描述了如何在ECS管理控制台上创建一个部署集。

背景信息

部署集是实例分布的控制策略，使您能在创建实例时就开始设计容灾能力和业务可用性。您可以使用部署集将业务涉及到的几台ECS实例分散部署在不同的物理服务器上，以此保证业务的高可用性和底层容灾能力。更多详情，请参阅[部署集](#)。如果您是 API 用户，可以调用[CreateDeploymentSet](#)。

操作步骤

1. 登录[ECS管理控制台](#)。
2. 选择地域。
3. 在左侧导航栏，选择网络和安全 > 部署集。

4. 在部署集列表页面上，单击创建部署集。
5. 在创建部署集页面上，输入部署集名称和描述。部署集策略目前仅支持高可用，更多关于部署集策略的详情，请参阅[#unique_22/unique_22_Connect_42_Strategy](#)。

后续操作

创建了部署集后，您可以：

- [在部署集中创建实例](#)。
- 将一台ECS实例加入当前部署集（缺少链接）

1.3 在部署集中创建实例

本文描述了如何通过ECS管理控制台在部署集中创建实例。

背景信息

一个部署集在一个可用区内只能创建7台实例，一个地域内可以创建（7*可用区）数量的实例。创建实例时，您可以搭配使用实例启动模板或者批量创建功能，避免数量限制带来的不便。

如果您是API用户，可以调用[RunInstances](#)并在请求中指定DeploymentSetId参数。

操作步骤

1. 登录[ECS管理控制台](#)。
2. 选择地域。
3. 在左侧导航栏，选择网络和安全 > 部署集。
4. 在部署集列表页面，找到目标部署集。您可以在部署集的操作菜单中单击创建实例，也可以勾选部署集后在实例列表处单击创建实例。

5. 在跳转后的自定义购买页面完成实例配置选项，具体操作步骤可参阅[使用向导创建实例](#)。其中，您需要注意的是：

- 基础配置：

- 地域：实例必须和目标部署集在同一个地域。
 - 可用区：部署集的每一个可用区内均不能超过7台实例。
 - 实例：部署集现阶段能创建的实例规格族仅支持c5、g5、hfc5、hfg5、r5、se1ne、sn1ne和sn2ne。更多有关实例规格及其性能的信息，请参阅[实例规格族](#)。
 - （可选）您可以在购买实例数量处指定此次操作的实例数量，并需要考虑部署集的当前可用区内已有的实例数量。
 - （可选）系统配置 > 有序后缀：批量创建实例后，可以为实例名称和主机名添加有序后缀。有序后缀从001开始递增，最大不会超过999。
 - 分组配置 > 部署集：选择目标部署集即可。

 - （可选）确认订单 > 保存为启动模板：保存配置为启动模板，便于下次快速创建实例。更多详情，请参阅[实例启动模板](#)。
6. 确认无误后，单击创建实例。
7. 在左侧导航栏，单击部署集，您可以查看在部署集内成功创建的实例信息。

后续操作

创建实例后，您可以：

- 在部署集内查看并管理实例。更多详情，请参阅[用户指南下实例相关操作文档](#)。
- 更换实例所在的部署集（缺少链接）。

1.4 管理部署集

创建了部署集后，您可以重新修改部署集名称和描述，或者删除不再需要的部署集以确保不超过使用限制。

修改部署集信息

在ECS管理控制台上修改部署集名称和描述的步骤如下：

1. 登录[ECS管理控制台](#)。
2. 选择地域。
3. 在左侧导航栏中，选择网络和安全 > 部署集。

4. 找到需要编辑的部署集。
5. 在下列两种方式中，任选一种方式修改信息：

- 鼠标悬浮至部署集名称处，单击出现的图标，重新填写部署集名称和描述。
- 在目标部署集操作菜单中，单击修改信息，重新填写部署集名称和描述。

6. 单击确定完成修改。

您也可以使用ECS API [ModifyDeploymentSetAttributes](#)修改部署集名称和描述。

删除部署集



说明:

当部署集中仍有实例存在时，您将无法删除部署集。

在ECS管理控制台上删除一个或多个部署集的步骤如下：

1. 登录[ECS管理控制台](#)。
2. 选择地域。
3. 在左侧导航栏中，选择网络和安全 > 部署集。
4. 勾选一个或多个需要删除的部署集，鼠标悬浮至操作，单击出现的删除部署集按键。
5. 单击确定删除部署集。

您可以使用ECS API [DeleteDeploymentSet](#)删除部署集。

2 云助手

2.1 云助手概述

云助手可以帮您自动地批量地执行日常维护命令，快速完成运行自动化运维脚本、轮询进程、安装或者卸载软件、更新应用以及安装补丁等一些日常任务。您可以使用云助手安全便捷地管理您的ECS实例。

应用场景

云助手可帮您完成包括但不限于以下部署与运维任务：

- 运行自动化运维脚本
- 运行实例上已有的脚本
- 轮询进程
- 安装或者卸载软件
- 更新应用以及安装补丁
- 部署代码或者应用
- 从OSS或者yum源下载代码或者软件
- 查看应用安装情况
- 修改主机名
- 修改登录密码

计费方式

云助手是免费功能，不收取任何费用。但是云助手是ECS实例的便捷部署与运维工具，使用ECS实例会产生计费，更多详情，请参阅[计费概述](#)。

功能详情

通过在实例中安装云助手客户端，您可以在ECS管理控制台上或者调用API对运行中（Running）的一台以及多台实例执行bat/PowerShell（Windows实例）脚本或者Shell脚本（Linux实例）。并且多台实例间互相不影响。您也可以设置命令执行周期，使实例维持在某种状态、获取实例监控以及日志信息或者守护进程等。云助手不会主动发起任何操作，所有的操作都在您的可控范围内。

下表罗列了您需要了解的云助手具体组成结构：

表 2-1: 云助手具体组成结构表

名词	常用名	描述
云助手	云助手	云助手官方名称，可以帮助您自动以及批量执行日常维护任务。
云助手客户端	客户端	安装在实例中的客户端程序，任务进程名称为 AliyunService 中。所有在实例中完成的命令都会通过客户端完成。
命令	Command	需要在实例中执行的具体命令操作，如具体的 Shell 脚本。
单次执行	Invocation	在一台或者多台实例中执行某个命令，即为一次执行（Invocation）。
周期执行	Timed Invocation	在一台或者多台实例中执行某个命令时，您可以指定执行时序/周期，周期性地执行命令进程。
执行状态	InvokeStatus	命令执行状态之间的关系。更多详情，请参阅 命令运行的生命周期 。

使用限制

使用云助手有如下限制：

- 您需要以管理员的身份安装和使用云助手，其中 Linux 实例的管理员为 root，Windows 实例的管理员为 administrator。
- 在一个阿里云地域下，您最多能创建 100 条云助手命令。
- 在一个阿里云地域下，您每天最多能运行 100 次云助手命令。
- 所有阿里云地域均支持云助手服务。
- 对于周期性云助手命令，Timed 设定间隔不能小于 10 秒。
- 创建的 bat/PowerShell 脚本或者 Shell 脚本的在 Base64 编码后大小不能超过 16 KB。
- 目标 ECS 实例的状态必须处于运行中（Running）。
- 您只能在 Windows Server 2008/2012/2016、Ubuntu 12/14/16、CentOS 5/6/7、Debian 7/8/9、RedHat 5/6/7、SUSE Linux Enterprise Server 11/12、OpenSUSE、Aliyun Linux 和 CoreOS 操作系统中运行云助手命令。

命令运行的生命周期

在一台实例上运行一条命令，命令会有以下状态信息：

表 2-2: 一条命令在一台实例上被执行的状态

命令状态	API 状态	状态描述
执行中	Running	命令正在被执行。
停止	Stopped	您在命令执行过程中自行停止了命令。
执行完成	Finished	命令顺利执行完成，执行完成并不代表执行成功，您需要根据命令进程的实际输出信息（Output）查看是否成功执行。
执行失败	Failed	命令进程过了超时时间（Timeout）后，因为各种原因未执行完成。

为便于管理批量执行或者周期执行，我们可以从 总执行状态、实例级别执行状态 以及 执行记录级别的状态 概念出发管理命令运行的生命周期。状态各级别之间的包含关系如下图所示：

图 2-1: 状态管理的包含关系示意图

表 2-3: 单次批量执行命令的生命周期

状态	命令运行	状态显示
总执行状态	所有实例级别执行状态均为 执行完成（Finished）时	执行完成
	部分实例级别执行状态为 执行完成（Finished） 并且部分实例级别执行状态为 停止（Stopped）时	
	所有实例级别执行状态均为 执行失败（Failed）时	执行失败
	所有实例级别执行状态均为 停止（Stopped）时	停止
	所有实例级别执行状态均为 执行中（Running）或者部分实例级别执行状态为 执行中（Running）时	执行中
	部分实例级别执行状态为 执行失败（Failed）时	部分失败
实例级别执行状态	单次批量执行命令为一次性操作，所以实例级别执行状态与执行记录级别的状态相同。	
执行记录级别的状态	参阅表格 一条命令在一台实例上被执行的状态	

以三台 ECS 实例为例，下图为多台实例同时做单次批量执行命令时，总执行状态与实例级别执行状态的关系。

图 2-2: 单次批量执行命令的生命周期示意图

表 2-4: 周期命令执行的生命周期

状态	状态描述
总执行状态	除用户自行停止所有实例的执行之外，总执行状态一直维持为 执行中（Running）
实例级别执行状态	除用户自行停止之外，实例级别执行状态一直维持为 执行中（Running）
执行记录级别的状态	参阅表格 一条命令在一台实例上被执行的状态

相关链接

您需要在 ECS 实例中安装 [云助手客户端](#)后才能使用云助手。

云助手实践请参见[云助手自动化部署实例实践](#)。

控制台操作：

- [创建命令](#)
- [执行命令](#)
- [查询执行结果与状态](#)
- [管理命令](#)

API 操作：

- [*CreateCommand*](#): 创建命令
- [*InvokeCommand*](#): 对目标 ECS 实例执行已创建的命令
- [*DescribeInvocations*](#): 查看命令的执行状态
- [*DescribeInvocationResults*](#): 查看命令的执行结果，即在指定 ECS 实例中的实际输出信息（Output）
- [*StopInvocation*](#): 停止正在进行的命令进程
- [*ModifyCommand*](#): 修改已创建的命令的内容
- [*DescribeCommands*](#): 查询您已经创建的命令
- [*DeleteCommand*](#): 删除已创建的命令

2.2 配置云助手客户端

云助手客户端是为ECS实例执行云助手命令的代理程序。本文提供安装、升级以及禁用云助手客户端的操作指示。

安装客户端：通过下载链接安装



说明：

- 2017年12月01日之后使用公共镜像创建的ECS实例，默认预装云助手客户端。如果您的实例是2017年12月01日之前购买的，若需要使用云助手相关功能，请自行安装客户端。
- 使用自定义镜像或者云市场镜像创建的实例，请参见[云助手](#)查询您的实例规格以及操作系统是否能使用云助手服务，再自行安装客户端。

Windows实例

1. 远程连接Windows实例。
2. 下载云助手客户端。
3. 双击客户端文件，根据安装向导安装。

Linux实例

请根据您的操作系统，选择下列安装方式的其中一种。

- RPM包安装，适合CentOS、RHEL和SUSE Linux等操作系统：
 1. 远程连接Linux实例。
 2. 运行`wget https://repo-aliyun-assist.oss-cn-beijing.aliyuncs.com/download/aliyun_assist.rpm`下载云助手客户端RPM包。
 3. 运行`rpm -ivh aliyun_assist.rpm`安装云助手客户端。
- deb包安装，适合Debian和Ubuntu等操作系统：
 1. 远程连接Linux实例。
 2. 运行`wget https://repo-aliyun-assist.oss-cn-beijing.aliyuncs.com/download/aliyun_assist.deb`下载云助手客户端deb包。
 3. 运行`dpkg -i aliyun_assist.deb`安装云助手客户端。

- 源码编译安装：

1. 远程连接Linux实例。
2. 运行`git clone https://github.com/aliyun/aliyun_assist_client`下载云助手源码。
3. 进入源码存放目录。
4. 运行`cmake .`生成编译文件。



说明:

若编译时报错CMAKE_MINIMUM_REQUIRED, 请前往[CMake官网](#)升级CMake服务至3.1及以上版本。

5. 运行`make`开始编译。
6. 运行`./cmake_install.sh`安装云助手客户端。

安装客户端：通过阿里云CLI安装

前提条件

- 本示例在命令行工具中完成，请确保您已经安装了阿里云命令行工具CLI。如何获取和安装阿里云CLI，请参见[阿里云命令行工具CLI](#)。
- 已获取地域ID，示例命令中需要填写。详细的地域ID，请参见[地域和可用区](#)。

操作步骤

1. 调用[DescribeCloudAssistantStatus](#)查询目标实例是否安装了云助手客户端。

```
aliyun ecs DescribeCloudAssistantStatus --RegionId TheRegionId  
--InstanceId.1 i-bp1g6zv0ce8ogXXXXXXp --output  
cols=CloudAssistantStatus
```

当返回CloudAssistantStatus=true结果时，表示您的实例已安装云助手客户端。否则请继续进入下一步。

2. 调用[InstallCloudAssistant](#)为实例安装客户端。

```
aliyun ecs InstallCloudAssistant --RegionId TheRegionId --  
InstanceId.1 i-bp1g6zv0ce8ogXXXXXXp
```

升级客户端

云助手客户端每隔一个小时会运行升级资源检测进程，该进程程序所在路径为：

- Windows实例：`C:\ProgramData\aliyun\assist\${version}/aliyun_assist_update`

- Linux实例: `/usr/local/share/aliyun-assist/${version}/aliyun_assist_update`

自动升级失效时，您可以[创建升级命令](#) (*CreateCommand*)，定期执行。以RPM安装包为例，以下云助手命令可以升级云助手客户端：

```
wget https://repo-aliyun-assist.oss-cn-beijing.aliyuncs.com/download/aliyun_assist.rpm  
rpm -U aliyun_assist.rpm
```

禁止升级客户端

云助手客户端升级进程默认自启动，您也可以禁用该服务：

- Windows实例：在命令提示符窗口CMD中运行`rename aliyun_assist_update`
- Linux实例：运行`chmod a-x aliyun_assist_update`

禁用客户端

Windows实例

1. [远程连接Windows实例](#)。
2. 找到计算机管理 > 服务和应用程序 > 服务，并找到AliyunService。



警告：

AliyunService是云助手服务的运行名称，禁用AliyunService相当于禁用云助手客户端，这可能会导致实例异常并无法在ECS管理控制台停止运行中的实例。建议您谨慎禁用云助手客户端。

3. 单击停止此服务。

Linux实例

1. [远程连接Linux实例](#)。

2. 根据您的init系统选择以下命令关闭云助手客户端服务。

- Debian、CentOS和Red Hat等操作系统:

```
systemctl stop agentwatch
```

- 使用其他init系统的操作系统:

```
chkconfig agentwatch off
```

相关链接

云助手客户端代码完全开源，您可以访问[阿里云Github仓库](#)参与开源计划。

以下帮助文档中，可能需要使用云助手客户端：

- [云助手](#)
- [InvokeCommand](#)
- [云助手自动化部署实践](#)

2.3 云助手自动化部署实践

本教程指引您如何通过云助手在ECS实例上批量执行一条echo 123的命令，达到自动化运维实例的目的。通过云助手，您可以自动化并批量处理日常维护任务，避免手动维护花费的大量时间和精力，并且保持ECS实例的最佳状态以及确保排错效率。

前提条件

无论您通过哪种方式使用云助手，目标实例必须满足以下条件：

- 实例已安装云助手客户端。如何安装，请参见[云助手客户端](#)。
- 实例状态为运行中（Running）。
- 实例网络类型为[专有网络（VPC）](#)。
- 执行PowerShell命令时，Windows实例已经配置了PowerShell模块。更多详情，请参见[安装Windows PowerShell](#)。

方法一：通过阿里云CLI调用API

前提条件

- 本示例在命令行工具中完成，请确保您已经安装了阿里云命令行工具CLI。如何获取和安装阿里云CLI，请参见[阿里云命令行工具CLI](#)。
- 已获取地域ID，示例命令中需要填写。详细的地域ID，请参见[地域和可用区](#)。

操作步骤

本示例调用API时仅传入了部分基本必需参数，执行结果均采用默认设置。您可以阅读API文档后自定义运行命令，优化运行效果。例如，在创建命令时为命令自定义设置执行目录（WorkingDir）和超时时间（TimeOut）等。

1. (可选) 检查实例状态，若实例的状态不是运行中（Running），调用[StartInstance](#)运行目标实例。

```
aliyun ecs StartInstance --InstanceId i-bp1g6zv0ce8ogXXXXXXp
```

2. (可选) 调用[DescribeCloudAssistantStatus](#)查询目标实例是否安装了云助手客户端。

```
aliyun ecs DescribeCloudAssistantStatus --RegionId TheRegionId  
--InstanceId.1 i-bp1g6zv0ce8ogXXXXXXp --output  
cols=CloudAssistantStatus
```

当返回CloudAssistantStatus=true结果时，表示您的实例已安装云助手客户端。否则请调用[InstallCloudAssistant](#)为实例安装客户端。

3. 调用[CreateCommand](#)创建一份名为test的云助手命令，命令内容为echo 123，并使用Base64编码明文命令。

```
aliyun ecs CreateCommand --RegionId TheRegionId --  
CommandContent ZWNobyAxMjM= --Type RunShellScript --Name test --  
Description test --Output cols=CommandId
```

云助手支持如下三种命令类型：

- Shell脚本（RunShellScript）：支持Linux实例。
- PowerShell脚本（RunPowerShellScript）：支持Windows实例。
- Bat脚本（RunBatScript）：支持Windows实例。

如果目标实例为Windows实例，将Type修改为RunBatScript或者RunPowershellScript

◦

4. 调用[InvokeCommand](#)为一台或多台实例执行已创建的云助手命令。

```
aliyun ecs InvokeCommand --RegionId TheRegionId --InstanceId.1 i-  
bp1g6zv0ce8ogXXXXXXp --InstanceId.2 i-bp1g6zv0ce8ogXXXXXXp --  
CommandId your-command-id --Timed false --Output cols=InvokeId
```

- Timed表示是否为周期性任务，True表示周期性任务，False表示不是周期性任务。
- 当命令为周期性任务时，通过参数Frequency指定运行周期，例如0 */20 * * * 表示周期为每20分钟。更多关于Cron表达式详情，请参见[Cron表达式取值说明](#)。

5. 调用[DescribeInvocations](#)查看命令执行状态。其中，InvokeId是执行命令时返回的执行ID。

```
aliyun ecs DescribeInvocations --RegionId TheRegionId --  
InvokeId your-invoke-id
```



说明:

返回参数InvokeStatus为Finished时仅表示命令进程执行完成，不代表一定有预期的命令效果，您需要通过[DescribeInvocationResults](#)中的参数Output查看实际的具体执行结果。

6. 调用[DescribeInvocationResults](#)查看指定实例的命令的实际执行结果。

```
aliyun ecs DescribeInvocationResults --RegionId TheRegionId --  
InstanceId i-bp1g6zv0ce8ogXXXXXXp --InvokeId your-invoke-id
```

方法二：通过ECS Python SDK调用API

前提条件

- 您需要使用2.1.2及以上版本的Python SDK。更多详情，请前往[GitHub Repo Alibaba Cloud](#)获取最新版本ECS SDK。
- 示例SDK中需要您自行填写AccessKey和地域ID，请确保您已经获取了AccessKeyId和AccessKeySecret。更多详情，请参见[创建AccessKey](#)和[地域和可用区](#)。

示例代码

```
# coding=utf-8  
# If the python sdk is not installed, run 'sudo pip install aliyun-  
python-sdk-ecs'.  
# Make sure you're using the latest sdk version. Run 'sudo pip install  
--upgrade aliyun-python-sdk-ecs' to upgrade.  
  
import json  
import logging  
import os  
import time  
import datetime  
import base64  
from aliyunsdkcore import client  
  
from aliyunsdkecs.request.v20140526.CreateCommandRequest import  
CreateCommandRequest  
from aliyunsdkecs.request.v20140526.InvokeCommandRequest import  
InvokeCommandRequest  
from aliyunsdkecs.request.v20140526.DescribeInvocationResultsRequest  
import DescribeInvocationResultsRequest  
  
# configuration the log output formatter, if you want to save the  
output to file,  
# append ",filename='ecs_invoke.log'" after datefmt.  
logging.basicConfig(level=logging.INFO,
```

```
format='%(asctime)s %(filename)s[line:%(lineno)d]
%(levelname)s %(message)s',
        datefmt='%a, %d %b %Y %H:%M:%S',filename='
aliyun_assist_openapi_test.log', filemode='w')
#access_key = 'Your Access Key Id'
#acess_key_secrect = 'Your Access Key Secret'
#region_name = 'cn-shanghai'
#zone_id = 'cn-shanghai-b'

access_key = 'LTAIXXXXXXXXXXXXXX' # 请根据实际情况填写
acess_key_secrect = '4dZXXXXXXXXXXXXXXXXXXXXXX' # 请根据实际情况填写
region_name = 'cn-hangzhou' # 请根据实际情况填写
zone_id = 'cn-hangzhou-f' # 请根据实际情况填写

clt = client.AcsClient(access_key, acess_key_secrect, region_name)

def create_command(command_content, type, name, description):
    request = CreateCommandRequest()
    request.set_CommandContent(command_content)
    request.set_Type(type)
    request.set_Name(name)
    request.set_Description(description)
    response = _send_request(request)
    if response is None:
        return None
    command_id = response.get('CommandId')
    return command_id;

def invoke_command(instance_id, command_id, timed, cronat):
    request = InvokeCommandRequest()
    request.set_Timed(timed)
    InstanceIds = [instance_id]
    request.set_InstanceIds(InstanceIds)
    request.set_CommandId(command_id)
    request.set_Frequency(cronat)
    response = _send_request(request)
    invoke_id = response.get('InvokeId')
    return invoke_id;

def get_task_output_by_id(instance_id, invoke_id):
    logging.info("Check instance %s invoke_id is %s", instance_id,
invoke_id)
    request = DescribeInvocationResultsRequest()
    request.set_InstanceId(instance_id)
    request.set_InvokeId(invoke_id)
    response = _send_request(request)
    invoke_detail = None
    output = None
    if response is not None:
        result_list = response.get('Invocation').get('Invocation
Results').get('InvocationResult')
        for item in result_list:
            invoke_detail = item
            output = base64.b64decode(item.get('Output'))
            break;
    return output;

def execute_command(instance_id):
    command_str = 'yum check-update'
    command_id = create_command(base64.b64encode(command_str), '
RunShellScript', 'test', 'test')
    if(command_id is None):
        logging.info('create command failed')
        return
```

```
invoke_id = invoke_command(instance_id, command_id, 'false', '')
if(invoke_id is None):
    logging.info('invoke command failed')
    return

time.sleep(15)

output = get_task_output_by_id(instance_id, invoke_id)
if(output is None):
    logging.info('get result failed')
    return

logging.info("output: %s is \n", output)

# send open api request
def _send_request(request):
    request.set_accept_format('json')
    try:
        response_str = clt.do_action(request)
        logging.info(response_str)
        response_detail = json.loads(response_str)
        return response_detail
    except Exception as e:
        logging.error(e)

if __name__ == '__main__':
    execute_command('i-bp17zhpXXXXXX')
```

方法三：通过控制台使用云助手

通过控制台使用云助手的步骤，请参见[新建命令](#)。

相关链接

云助手提供了以下API供您使用：

- [*CreateCommand*](#): 创建一条云助手命令
- [*InvokeCommand*](#): 在一台或多台实例上运行一条云助手命令
- [*DescribeInvocations*](#): 查询命令运行状态
- [*DescribeInvocationResults*](#): 查询命令运行结果
- [*StopInvocation*](#): 停止正在进行的命令进程
- [*ModifyCommand*](#): 修改已创建的命令的内容
- [*DescribeCommands*](#): 查询您已经创建的命令
- [*DeleteCommand*](#): 删除已创建的命令
- [*DescribeCloudAssistantStatus*](#): 查询目标实例是否安装了云助手客户端
- [*InstallCloudAssistant*](#): 为实例安装云助手客户端

3 系统事件

3.1 系统事件概述

系统事件是影响实例运行状态的有计划或非预期事件。特指因探测到更新维护、违规操作、系统故障、软硬件故障、因预付费即将到期或账号欠费时，发生的重启、停止或者释放 ECS 实例等操作的事件。

概述

实例是您构建业务应用的核心基础组件，当您完成了实例选型、购买、初始化配置并开始部署业务后，实例的健康状态对于您的业务连续性和系统可靠性有关键作用。为提升系统可靠性、系统性能和安全防护能力，ECS 会主动对实例所宿的物理服务器做日常维护并修复潜在的系统故障。当探测到物理服务器出现故障或者隐患时，ECS 将实例在线热迁移到健康的服务器，以保持实例的健康运行状态。这些维护是日常维护。日常维护有别于系统事件，日常维护通常是无感知完成的，不会发送维护通知，也不对实例造成影响。

发生系统事件时，ECS 及时给您发送通知、应对措施和事件周期等信息。对于有计划的系统事件，会提前告知该事件对实例的影响和预计执行时间点等信息。您可以在执行系统事件之前及时备份数据、做好应用层面的准备进而防备对业务的可靠性和连续性的潜在影响。对于已处理的系统事件，您可以查询过去两个月内实例的历史系统事件，获取故障诊断和复盘分析数据。

使用限制

除已停售的实例规格族之外，例如

sn2、sn1、t1、s1、s2、s3、m1、m2、c1、c2、c4、ce4、cm4、n1、n2、e3 等，所有的实例规格族均支持系统事件功能。更多详情，请参阅 [实例规格族](#)。

事件类型

系统事件类型如下表所示。

类别	事件类型	对应参数
实例重启	因系统维护实例重启	SystemMaintenance.Reboot
实例异常重启	因系统错误实例重新启动	SystemFailure.Reboot
	因实例错误实例重启	InstanceFailure.Reboot
实例停止	因预付费期限到期实例停止	InstanceExpiration.Stop
	因账号欠费按量付费资源停止	AccountUnbalanced.Stop

类别	事件类型	对应参数
实例释放	因预付费期限到期实例释放	InstanceExpiration.Delete
	因账号欠费按量付费资源释放	AccountUnbalanced.Delete

事件状态

系统事件的生命周期中，有如下事件状态。

事件状态	状态属性	状态描述
待执行	中间状态	系统事件已发生，但尚未执行
已避免	稳定状态	您在 用户操作窗口期 内提前操作，避免了系统事件在计划时间点执行
执行中	中间状态	正在执行系统事件
已执行	稳定状态	系统事件已完成
已取消	稳定状态	ECS 主动取消了已计划的系统事件
失败	稳定状态	系统事件执行失败

事件窗口

系统事件有如下窗口期：

- 用户操作窗口期：系统事件计划发起到执行之间的时间段。通常系统维护相关事件用户操作窗口期为 24~48 小时，预付费实例即将到期而停止的操作窗口期为 3 天，账号欠费而停止的操作窗口期通常小于 1 小时。因计费问题引发系统事件实例会被立刻停止、实例会在 15 天后被释放。
您可以根据该事件对业务的影响情况，选择推荐的应对方式提前修复系统事件，或者等待执行默认应对动作。对于 ECS 主动修复系统故障触发的系统事件，ECS 根据系统维护操作计划安排提前一段时间向您发送事件通知。
- 事件执行窗口期：若您没有提前应对系统事件，修复故障类系统事件通常会在进入事件执行窗口期的 6 小时内自动完成。随后您会收到系统事件的执行结果。



说明：

对于突发故障或者违规操作等非技术因素导致的非预期系统事件，通常不会存在用户操作窗口期，只有短暂的事件执行窗口期。您可以查询历史系统事件用于故障诊断、原因分析或数据还原。

查看系统事件

控制台操作步骤

如果存在有计划的系统事件，ECS 控制台 待处理事件 按钮上会出现显著标示提醒您查看。

1. 登录 [ECS管理控制台](#)。
2. 在左侧导航栏里，单击概览。
3. 在概览 页的常用操作区域，单击待处理事件。
4. 单击系统计划事件页面，您可以看到实例 ID、地域、实例运行状态、事件类型和推荐操作等相关信息。您可以在操作列执行需要的操作。

API 操作步骤

本文API示例的调用工具为阿里云CLI。更多有关如何调用API的详情，请参阅 [ECS API快速入门](#)。

1. 获取实例ID。

```
aliyun ecs DescribeInstances --RegionId <TheRegionId> --output cols=InstanceId,InstanceName
```

2. 使用 [DescribeInstancesFullStatus](#) 查询实例的系统事件。

```
aliyun ecs DescribeInstancesFullStatus --RegionId <TheRegionId> --InstanceId.1 <YourInstanceId> --output cols=EventId,EventTypeName
```

实例 Metadata 操作步骤

更多详情，请参见[实例元数据](#)动态实例元数据项。

查看历史系统事件

您可以查看过去两个月内发生的全部系统事件记录，为故障诊断和复盘分析提供数据支撑。

控制台操作步骤

1. 登录 [ECS管理控制台](#)。
2. 在左侧导航栏里，单击 概览。
3. 在 概览 页的常用操作区域，单击 待处理事件。
4. 在左侧导航栏中，单击 全部事件。在 全部事件 页面，单击 系统计划事件 > 实例，查看实例 ID、事件类型和事件状态等相关信息。

API 操作步骤

1. 获取实例ID。

```
aliyun ecs DescribeInstances --RegionId <TheRegionId> --output cols=InstanceId,InstanceName
```

2. 使用 [DescribeInstanceHistoryEvents](#) 查询实例的历史系统事件。

```
aliyun ecs DescribeInstanceHistoryEvents --RegionId <TheRegionId> --InstanceId.1 <YourInstanceId> --output cols=EventId,EventTypeName
```

订阅事件通知

所有系统事件均可以通过云监控配置报警规则，当事件发生时会及时通知您。更多详情，请参阅 [云监控 文档 云产品系统事件监控](#)。

应对建议

您可以通过控制台操作或者 API 调用应对系统事件。从系统事件出发管理 ECS，可以提高您对 ECS 的底层感知能力，优化 ECS 实例运维。

事件类型	对实例影响	建议措施
因系统维护实例重启	实例将在计划维护时间重新启动	<p>在用户操作窗口期某个合适的时间点选择以下任一种方法：</p> <ul style="list-style-type: none">· ECS控制台 #unique_69。· 调用 API RebootInstance。 <p> 说明： 在实例内部重启无法应对系统事件。</p> <ul style="list-style-type: none">· 在应用层面，切换流量。或从 负载均衡 实例中移除有计划维护的 ECS 实例，避免影响您的业务。 <p>并视情况为实例挂载的磁盘 #unique_72 (CreateSnapshot) 备份数据。</p>
因系统错误实例重启	发生非预期的宿主机故障时，实例重启后会重新响应	当您收到事件通知时，实例正在或已完成重新启动，建议您： <ul style="list-style-type: none">· 验证实例和应用是否恢复正常。· 订阅事件 通知到您的消息服务 MNS 队列或主题，联动切换流量或实现负载均衡，实现故障转移。

事件类型	对实例影响	建议措施
因实例错误实例重启	因操作系统崩溃实例重新启动	<p>当您收到事件通知时，实例正在或已完成重新启动，建议您：</p> <ul style="list-style-type: none"> · 查看实例系统日志和屏幕截图排查故障，检查操作系统发生崩溃的原因，避免再次引发系统崩溃问题。 · 验证实例和应用是否恢复正常。
因预付费实例到期停止	预付费资源即将被停止	建议您根据需要 续费资源 ，或等待实例过期。
因账号欠费后按量付费资源停止	按量付费资源即将被停止	建议您保持支付方式的额度充足，避免停止实例。
因预付费实例到期释放	预付费资源即将被释放	建议您根据需要 续费资源 ，或等待实例释放。
因账户欠费后按量付费资源释放	按量付费资源即将被释放	建议您保持支付方式的额度充足，避免释放实例。

3.2 实例自动恢复

本文介绍实例的自动恢复事件如何改善您使用阿里云ECS时的运维效率和整体体验。

什么是自动恢复

在托管ECS实例的底层硬件意外崩溃的情况下，如果确认故障不可逆转并且实例无法修复，则实例会在计划的维护时段自动重新启动，实现宕机迁移。在这种情况下，所恢复实例的所有[实例元数据](#)都将保持不变，例如实例ID以及私有和公有IP地址等。

自动恢复是一类[系统事件](#)，其事件代码为SystemFailure.Reboot。自动恢复事件有明确的维护时间表但不允许用户自行操作。因此，自动恢复不同于热迁移和其他类型的系统事件，热迁移是无感知且无需人工干预的，而您也可以在事件窗口期内自行响应其他类型的系统事件。下表区分了自动恢复与其他常见的运维事件：

运维事件类型	是否可以查询运维计划	是否可以人工干预
自动恢复	是	否
热迁移	否	否
其他类型系统事件	是	是

使用限制

- 实例预约了自动恢复事件后，您无法自行重启实例。

- 仅当底层硬件可在意外崩溃后重新启动时，本地盘实例才可自动恢复。如果无法恢复使用本地磁盘的实例，您可以立即[提交工单](#)检查磁盘上的数据是否保留，或实例是否已重新部署到其他物理服务器。

调用API查看自动恢复事件

本文示例通过阿里云CLI调用[DescribeInstancesFullStatus](#)查看实例是否有计划中的自动恢复事件。

```
aliyun ecs DescribeInstancesFullStatus --RegionId TheRegionId --InstanceId.1 YourInstanceId --output cols=EventId,EventTypeName
```



说明:

如果EventTypeName返回的事件代码是SystemFailure.Reboot，说明实例已经有计划中的自动恢复系统事件。

有关如何使用ECS控制台查看自动恢复事件，请参见[系统事件](#)。有关如何使用其他开发者工具的说明，请参见[ECS API快速入门](#)。

如何提高容错率

若需充分利用实例的自动恢复功能和故障转移操作，请确保您完成了以下操作：

- 将您的核心应用程序（例如SAP HANA）添加到自启动项列表中，避免业务操作的任何中断。
- 开启应用程序的自动重新连接功能。例如，允许应用程序自动连接到MySQL、SQL Server或Apache Tomcat。
- 对于[负载均衡](#)用户，请将多台ECS实例部署在集群环境中，当某一台ECS实例处于自动恢复过程中时，其余ECS实例可以继续提供业务访问能力。
- 定期备份本地磁盘上的数据，以实现数据冗余和提供实例重新部署的数据文件。