

# Alibaba Cloud Elastic Compute Service

鏡像

檔案版本：20190806

# 法律聲明

阿里雲提醒您在閱讀或使用本文檔之前仔細閱讀、充分理解本法律聲明各條款的內容。如果您閱讀或使用本文檔，您的閱讀或使用行為將被視為對本聲明全部內容的認可。

1. 您應當通過阿里雲網站或阿里雲提供的其他授權通道下載、擷取本文檔，且僅能用於自身的合法合規的商務活動。本文檔的內容視為阿里雲的保密資訊，您應當嚴格遵守保密義務；未經阿里雲事先書面同意，您不得向任何第三方披露本手冊內容或提供給任何第三方使用。
2. 未經阿里雲事先書面許可，任何單位、公司或個人不得擅自摘抄、翻譯、複製本文檔內容的部分或全部，不得以任何方式或途徑進行傳播和宣傳。
3. 由於產品版本升級、調整或其他原因，本文檔內容有可能變更。阿里雲保留在沒有任何通知或者提示下對本文檔的內容進行修改的權利，並在阿里雲授權通道中不時發布更新後的使用者文檔。您應當即時關注使用者文檔的版本變更並通過阿里雲授權渠道下載、擷取最新版的使用者文檔。
4. 本文檔僅作為使用者使用阿里雲產品及服務的參考性指引，阿里雲以產品及服務的”現狀“、“有缺陷”和“當前功能”的狀態提供本文檔。阿里雲在現有技術的基礎上盡最大努力提供相應的介紹及操作指引，但阿里雲在此明確聲明對本文檔內容的準確性、完整性、適用性、可靠性等不作任何明示或暗示的保證。任何單位、公司或個人因為下載、使用或信賴本文檔而發生任何差錯或經濟損失的，阿里雲不承擔任何法律責任。在任何情況下，阿里雲均不對任何間接性、後果性、懲戒性、偶然性、特殊性或刑罰性的損害，包括使用者使用或信賴本文檔而遭受的利潤損失，承擔責任（即使阿里雲已被告知該等損失的可能性）。
5. 阿里雲網站上所有內容，包括但不限於著作、產品、圖片、檔案、資訊、資料、網站架構、網站畫面的安排、網頁設計，均由阿里雲和/或其關係企業依法擁有其智慧財產權，包括但不限於商標權、專利權、著作權、商業秘密等。非經阿里雲和/或其關係企業書面同意，任何人不得擅自使用、修改、複製、公開傳播、改變、散布、發行或公開發表阿里雲網站、產品程式或內容。此外，未經阿里雲事先書面同意，任何人不得為了任何營銷、廣告、促銷或其他目的使用、公布或複製阿里雲的名稱（包括但不限於單獨為或以組合形式包含”阿里雲”、Aliyun”、“萬網”等阿里雲和/或其關係企業品牌，上述品牌的附屬標誌及圖案或任何類似公司名稱、商號、商標、產品或服務名稱、網域名稱、圖案標示、標誌、標識或通過特定描述使第三方能夠識別阿里雲和/或其關係企業）。
6. 如若發現本文檔存在任何錯誤，請與阿里雲取得直接聯絡。

## 通用約定

格式	說明	範例
	該類警示資訊將導致系統重大變更甚至故障，或者導致人身傷害等結果。	 <b>禁止：</b> 重設操作將丟失使用者配置資料。
	該類警示資訊可能導致系統重大變更甚至故障，或者導致人身傷害等結果。	 <b>警告：</b> 重啟操作將導致業務中斷，恢復業務所需時間約10分鐘。
	用於補充說明、最佳實務、竅門等，不是使用者必須瞭解的內容。	 <b>說明：</b> 您也可以通過按Ctrl + A選中全部檔案。
>	多級菜單遞進。	設定 > 網路 > 設定網路類型
<b>粗體</b>	表示按鍵、菜單、頁面名稱等UI元素。	單擊 <b>確定</b> 。
<code>courier</code> 字型	命令。	執行 <code>cd / d C :/ windows</code> 命令，進入Windows系統檔案夾。
##	表示參數、變數。	<code>bae log list --instanceid Instance_ID</code>
[ ]或者[a b]	表示可選項，至多選擇一個。	<code>ipconfig [-all -t]</code>
{ }或者{a b}	表示必選項，至多選擇一個。	<code>swich {stand   slave}</code>

# 目錄

---

法律聲明.....	I
通用約定.....	I
1 鏡像.....	1
2 選擇鏡像.....	4
3 公用鏡像.....	9
3.1 Aliyun Linux 17.01 特性說明.....	9
3.2 已知問題.....	9
4 自訂鏡像.....	16
4.1 建立自訂鏡像.....	16
4.1.1 使用快照建立自訂鏡像.....	16
4.1.2 使用執行個體建立自訂鏡像.....	19
4.1.3 使用Packer建立自訂鏡像.....	19
4.1.4 使用 Packer 建立並匯入本地鏡像.....	22
4.2 管理自訂鏡像.....	29
4.3 匯入鏡像.....	30
4.3.1 鏡像規範偵查工具.....	30
4.3.2 匯入鏡像必讀.....	34
4.3.3 安裝cloud-init.....	36
4.3.4 安裝virtio驅動.....	43
4.3.5 配置 Customized Linux 自訂鏡像.....	47
4.3.6 轉換鏡像格式.....	52
4.3.7 匯入自訂鏡像.....	54
4.4 複製鏡像.....	56
4.5 共用鏡像.....	57
4.6 匯出鏡像.....	59
5 雲市場鏡像.....	61
6 鏡像相關開源工具.....	63
7 更換作業系統.....	64
8 常見問題.....	65
8.1 Windows Server半年渠道鏡像與執行個體管理.....	65
8.2 Linux鏡像如何開啟或關閉Meltdown與Spectre安全性漏洞補丁.....	70

# 1 鏡像

鏡像（Image）是Elastic Compute Service執行個體運行環境的模板，模板中包括了特定的作業系統資訊，有時也額外包括了一些預裝的應用程式。鏡像檔案相當於副本檔案，該副本檔案包含了一個或多個磁碟中的所有資料，對於ECS而言，這些磁碟可以是單個系統盤，也可以是系統盤加資料盤的組合。

## 鏡像種類

ECS提供了以下靈活多樣的鏡像種類，讓您方便地獲取鏡像資源。

種類	描述	來源
公共鏡像	<p>阿里雲官方提供的公共鏡像，支援幾乎所有Windows Server和Linux的主流鏡像版本。包括：</p> <ul style="list-style-type: none"> <li>· Windows Server</li> <li>· CentOS</li> <li>· Ubuntu</li> <li>· Debian</li> <li>· SUSE Linux</li> <li>· OpenSUSE</li> <li>· Aliyun Linux</li> <li>· CoreOS</li> <li>· FreeBSD</li> </ul> <p>此類鏡像具有高度穩定性，皆以正版授權，您可以在公共鏡像的基礎上個人化部署應用環境。</p>	阿里雲官方發行
自訂鏡像	<p>根據您現有的物理機伺服器、虛擬機器或者雲主機建立的自訂鏡像。此類鏡像靈活性高，能滿足您的個人化需求。</p>	<ul style="list-style-type: none"> <li>· 可以是您根據現有的ECS執行個體建立的。</li> <li>· 可以是您從線下環境匯入到ECS相應地域的。</li> </ul>
雲市場鏡像	<p>第三方服務商（ISV，Independent Software Vendor）提供的鏡像。雲市場的鏡像不僅包括應用所需的作業系統，並且提供配置環境，免除安裝和配置的繁瑣過程，達到一鍵部署ECS的效果。</p>	雲市場


種類	描述	來源
共用鏡像	其他阿里雲使用者共用給您的鏡像。	其他阿里雲使用者製作的自訂鏡像

### 鏡像格式

支援的鏡像檔案格式為VHD、qcow2和RAW，其他格式的鏡像檔案需要預先[使用工具轉換鏡像格式後](#)方能運行在ECS中。

### 費用詳情

使用鏡像會涉及到資源計費，建議您提前瞭解Elastic Compute Service的計費方式。更多詳情，請參閱[計費概述](#)。ECS的各類鏡像的費用詳情如下：

種類	費用描述
公共鏡像	免費
自訂鏡像	<p>免費。潛在的費用有：</p> <ul style="list-style-type: none"> <li>如果您 <a href="#">使用快照建立自訂鏡像</a>： <ul style="list-style-type: none"> <li>系統盤快照使用的鏡像來自於雲市場，可能涉及的費用為雲市場鏡像的費用和快照容量費用。</li> <li>系統盤快照使用的鏡像不是來自於雲市場，可能涉及的費用為快照容量費用。</li> </ul> </li> </ul> <div style="background-color: #f0f0f0; padding: 5px; margin: 10px 0;">  <b>说明：</b> 目前快照容量免費。 </div> <ul style="list-style-type: none"> <li>如果您 <a href="#">使用執行個體建立自訂鏡像</a> 時，若ECS執行個體使用的鏡像來自雲市場，付費方式以雲市場第三方服務商提供的資訊為準。</li> </ul>
雲市場鏡像	以雲市場第三方服務商提供的鏡像付費方式為準。
共用鏡像	如果您的共用鏡像的最終來源為雲市場鏡像時，以雲市場第三方服務商提供的鏡像付費方式為準。

### 使用限制

除公共鏡像外，阿里雲的自訂鏡像、雲市場鏡像和共用鏡像是地域等級的資源，鏡像資源會因為地域不同而不同。更多詳情，請參閱[地域與可用性區域](#)。

### 相關操作

#### 控制台操作

- 您可以 [使用已有的鏡像資源建立ECS執行個體](#)。

- 您可以通過以下方式更換ECS執行個體系統盤：
  - [使用公共鏡像更換系統盤](#)
  - [使用非公共鏡像更換系統盤](#)
- 您可以通過以下方式獲取自訂鏡像：
  - [使用快照建立自訂鏡像](#)
  - [使用執行個體建立自訂鏡像](#)
  - [匯入線下自訂鏡像](#)
- 建立自訂鏡像後，您可以：
  - [複製自己的自訂鏡像到其他地域](#)
  - [共用自己的自訂鏡像給其他阿里雲使用者](#)
  - [匯出自訂鏡像到本地測試或線下私人雲端環境](#)

#### API操作

您可以查看開發指南 [鏡像相關API](#)。

## 2 選擇鏡像

建立ECS執行個體時，您必須選擇鏡像。本文旨在協助您從眾多鏡像類型和作業系統中選擇合適的鏡像。

選擇鏡像時，您可能需要考慮以下幾個因素：

- 地區
- 鏡像類型
- 鏡像費用
- 作業系統
- 內建軟體（如鏡像中是否安裝了MySQL等應用軟體）

### 地區

鏡像是一種地區性資源，您不能跨地區使用鏡像建立執行個體。例如，在華北2地區建立執行個體時，您只能使用位於華北2地區的鏡像。更多阿里雲地區，請參見[地區和可用性區域](#)。

如需使用其他地區鏡像建立執行個體，可先複製該鏡像到當前地區，詳情請參見[複製鏡像](#)。

### 鏡像類型

ECS鏡像根據來源不同，分為公用鏡像、自訂鏡像、共用鏡像和鏡像市場的鏡像。詳情請參見[鏡像種類](#)。

### 鏡像費用

使用鏡像，可能會產生費用。

### 作業系統

選擇作業系統時，您可能需要考慮以下方面：

- 選擇系統架構（32位或64位）

系統架構	適用記憶體	使用限制
32位	適用於4 GiB以下記憶體	<ul style="list-style-type: none"><li>- 執行個體規格記憶體大於4 GiB時，無法使用32位作業系統。</li><li>- Windows 32位作業系統最高支援4核CPU。</li></ul>
64位	適用於4 GiB及以上記憶體	若您的應用需使用4 GiB以上記憶體或未來需擴充記憶體至4 GiB以上，請使用64位作業系統。



- 選擇作業系統類型（Windows或Linux/類Unix系統）

作業系統類型	登入方式	特點	適用情境
Windows	支援遠端桌面方式	Windows系統的公用鏡像內含正版已啟用系統。	<ul style="list-style-type: none"> <li>- 適合運行Windows下開發的程式，如.NET等。</li> <li>- 支援SQL Server等資料庫（需自行安裝）。</li> </ul>
Linux/類Unix	支援SSH方式	<ul style="list-style-type: none"> <li>- 常用的伺服器端作業系統，具備安全性和穩定性。</li> <li>- 開源，輕鬆建立和編譯原始碼。</li> </ul>	<ul style="list-style-type: none"> <li>- 一般用於高效能Web等伺服器應用，支援常見的PHP、Python等程式設計語言。</li> <li>- 支援MySQL等資料庫（需自行安裝）。</li> </ul>

阿里雲提供的Windows和Linux/類Unix系統的公用鏡像列表，請參見[公用鏡像概述](#)。

- 選擇Windows系統版本

建議您選擇高版本Windows系統。相對低版本來說，高版本系統漏洞更少，並且IIS 7.5比IIS 6提供了更多功能以及更方便的控制台。

請您閱讀以下注意事項並根據實際需要選擇合適的硬體設定和Windows版本：

- vCPU為1核且記憶體為1 GiB的執行個體規格無法啟動MySQL資料庫。
- Windows執行個體用於建站、部署Web環境時，最少需要2 GiB記憶體。
- 為了保證效能體驗，使用Windows 2008/2012系統時，建議您選擇記憶體為2 GiB及以上的執行個體規格。
- 使用Windows Server 2016/2019系統必須選擇記憶體為2 GiB及以上的執行個體規格，否則售賣頁的公用鏡像列表中不會出現Windows Server 2016/2019。
- 阿里雲不再為Windows Server 2003系統鏡像提供支援人員，詳情請參見[阿里雲不再支援Windows Server 2003系統鏡像](#)。

- 選擇Linux/類Unix系統的發行版

阿里雲的Linux/類Unix公用鏡像包含以下發行版：

- Aliyun Linux

Aliyun Linux是阿里雲研發的Linux作業系統，旨在為ECS執行個體上的應用程式提供更加安全、穩定、高效能的運行環境。其中Aliyun Linux 2支援各種雲情境負載，並支援各

規格執行個體（除使用傳統網路的執行個體和非I/O最佳化執行個體），詳情請參見[Aliyun Linux 2](#)。

- Red Hat系列

■ CentOS

■ Red Hat

CentOS和Red Hat兩種作業系統的特點和關係見下表：

作業系統	軟體包格式	安裝軟體	費用	特點	兩者關係
CentOS	rpm包	yum	免費使用	<ul style="list-style-type: none"> <li>■ 穩定, 但補丁更新速度慢於Red Hat。</li> <li>■ 支援線上升級, 可即時更新系統。</li> </ul>	<ul style="list-style-type: none"> <li>■ CentOS是Red Hat的開源版本。</li> <li>■ 兩者rpm包通用。</li> <li>■ 兩者命令通用。</li> </ul>

作業系統	軟體包格式	安裝軟體	費用	特點	兩者關係
Red Hat			付費使用	穩定，可獲得企業級支援人員。	

- Debian系列

■ Debian

■ Ubuntu

Debian和Ubuntu兩種作業系統的特點和關係見下表：

作業系統	軟體包格式	安裝軟體	特點	兩者關係
Debian	deb包	aptitude	系統相對穩定。	Ubuntu基於Debian開發，可簡單認為Ubuntu是Debian的功能加強版。
Ubuntu		apt-get	<ul style="list-style-type: none"> <li>■ 系統配置更人性化。</li> <li>■ 軟體的更新更為激進。</li> <li>■ 新手友好度更高，易用性更好，更容易上手。</li> </ul>	

- SUSE系列

- SUSE Linux

- OpenSUSE

OpenSUSE和SUSE Linux的關係和對比見下表：

作業系統	兩者對比	兩者關係
OpenSUSE SUSE Linux	<ul style="list-style-type: none"> <li>■ OpenSUSE是社區版，SUSE Linux Enterprise是企業版。</li> <li>■ SUSE Linux Enterprise更成熟、穩定，官方源中軟體與OpenSUSE相比較少。</li> <li>■ OpenSUSE軟體版本更新，可擴充性強（可用於案頭或伺服器），免費更新（也可購買官方支援人員）。</li> <li>■ SUSE Linux Enterprise更適用於工作、生產環境，娛樂功能較差，OpenSUSE可用於個人娛樂和其他較專業的用途。</li> </ul>	<ul style="list-style-type: none"> <li>■ SUSE Linux在版本10.2以後更名為OpenSUSE。</li> <li>■ OpenSUSE使用的核心與SUSE Linux相同。</li> </ul>

- CoreOS

CoreOS是一款基於Linux核心的輕量級作業系統，為電腦叢集的基礎設施建設而生，專注於自動化，輕鬆部署，安全，可靠，規模化。作為一個作業系統，CoreOS提供了在應用程式容器內部署應用所需要的基礎功能環境以及一系列用於服務發現和配置共用的內建工具。

- FreeBSD

FreeBSD是一款類Unix作業系統，為專注於功能、速度和穩定性的平台提供。FreeBSD提供了先進的網路、效能、安全和相容性功能，這些功能在其他動作系統甚至最好的商業作業系統中仍然缺失。詳情請參見[FreeBSD官方文檔](#)。

## 內建軟體

鏡像市場的鏡像，通常在作業系統中預裝了ECS執行個體的運行環境或軟體應用，您可根據實際需要，購買相應的鏡像建立執行個體。更多詳情，請參見[雲市場鏡像](#)。

## 下一步

- 使用鏡像建立執行個體。詳情請參見[使用向导创建实例](#)。
- 使用鏡像更換作業系統。詳情請參見[更換作業系統](#)。

## 3 公用鏡像

---

### 3.1 Aliyun Linux 17.01 特性說明

Aliyun Linux鏡像是公用鏡像的一種，是阿里雲針對 ECS 執行個體提供的定製化原生作業系統。

本文介紹Aliyun Linux鏡像 17.01（Golden Toad）版本的特性。

- 整合阿里雲核心組維護的 4.4 系列 Kernel
  - 支援 swap 記憶體壓縮可保證顯著減少 I/O；
  - 支援動態補丁機制 kpatch，升級核心無需重啟系統；
  - 支援超大記憶體 kdump（至多 3 TB）；
  - 支援 Automatic NUMA；
  - 使用者態 vsyscall 擷取時間戳記效能提升；
  - 全面支援 Docker 技術棧；
  - 支援 pv panic，方便協助雲使用者快速定位核心問題；
  - 重新設計的無鎖 TCP listener，提升 2 ~ 3 個數量級防 SYN 攻擊能力。
- 高效建立和啟動
  - 基於虛擬化級最佳化棧建立執行個體，有效降低建立虛擬機器所需時間。
  - OS 搭載阿里雲定製啟動服務，提升啟動速率，使建立到可登入時間控制在 30 秒內。
- 檔案系統  
新增支援檔案系統：XFS、Btrfs、AUFS 和 OverlayFS。
- 全新工具鏈  
GCC 4.8.x glibc 2.17 GDB 7.6.1。
- 安全提升
  - OpenSSH 支援 chroot shell 登入，限制使用者登入在一個受限的根檔案系統中。
  - OpenSSH 可指定多個驗證條件以加強登入訪問。
- 相容CentOS 7.2

### 3.2 已知問題

本文介紹了不同發行平台的阿里雲鏡像的已知故障、故障涉及範圍以及解決方案。

## Debian 9.6: 傳統網路配置問題

- 問題描述: 無法Ping通使用Debian 9公用鏡像建立的傳統網路類型執行個體。
- 問題原因: 因為Debian系統預設禁用了systemd-networkd服務, 傳統網路類型執行個體無法通過DHCP (Dynamic Host Configuration Protocol) 模式自動分配IP。
- 涉及鏡像ID: debian\_9\_06\_64\_20G\_alibase\_20181212.vhd。
- 修複方案: 您需要運行下列命令解決該問題。

```
systemctl enable systemd - networkd
systemctl start systemd - networkd
```

## CentOS 6.8: 裝有NFS Client的執行個體異常崩潰的問題

- 問題描述: 載入了NFS用戶端 (NFS Client) 的CentOS 6.8執行個體出現超長等待狀態, 只能通過重啟執行個體解決該問題。
- 問題原因: 在2.6.32-696 ~ 2.6.32-696.10的核心版本上使用NFS服務時, 如果通訊延遲出現毛刺 (glitch, 電子脈衝), 核心nfsclient會主動斷開TCP串連。若NFS服務端 (Server) 響應慢, nfsclient發起的串連可能會卡頓在FIN\_WAIT2狀態。正常情況下, FIN\_WAIT2狀態的串連預設在一分鐘後逾時並被回收, nfsclient可以發起重連。但是, 由於此類核心版本的TCP實現有缺陷, FIN\_WAIT2狀態的串連永遠不會逾時, 因此nfsclient的TCP串連永遠無法關閉, 無法發起新的串連, 造成使用者請求卡死 (hang死), 永遠無法恢復, 只能通過重啟ECS執行個體進行修複。
- 涉及鏡像ID: centos\_6\_08\_32\_40G\_alibase\_20170710.vhd和centos\_6\_08\_64\_20G\_alibase\_20170824.vhd。
- 修複方案: 您可以升級核心至2.6.32-696.11及以上版本。

```
yum update
```



说明:

操作執行個體時, 請確保您已經提前[建立了快照備份資料](#)。

## CentOS 7: 重啟系統後主機名稱大寫字母被修改

- 問題描述: 第一次重啟ECS執行個體後, 部分CentOS 7執行個體的主機名稱 (hostname) 存在大寫字母變成小寫字母的現象, 如下表所示。

執行個體hostname樣本	第一次重啟後樣本	後續是否保持小寫不變
izM5e1qe*****sxx1ps5zX	izm5e1qe*****sxx1ps5zx	是
ZZHost	zzhost	是
NetworkNode	networknode	是

- 涉及鏡像：以下CentOS公用鏡像，和基於以下公用鏡像建立的自訂鏡像。
  - centos\_7\_2\_64\_40G\_base\_20170222.vhd
  - centos\_7\_3\_64\_40G\_base\_20170322.vhd
  - centos\_7\_03\_64\_40G\_alibase\_20170503.vhd
  - centos\_7\_03\_64\_40G\_alibase\_20170523.vhd
  - centos\_7\_03\_64\_40G\_alibase\_20170625.vhd
  - centos\_7\_03\_64\_40G\_alibase\_20170710.vhd
  - centos\_7\_02\_64\_20G\_alibase\_20170818.vhd
  - centos\_7\_03\_64\_20G\_alibase\_20170818.vhd
  - centos\_7\_04\_64\_20G\_alibase\_201701015.vhd
- 涉及Hostname類型：如果您的應用有hostname大小寫敏感現象，重啟執行個體後會影響業務。您可根據下面的修復方案修復以下類型的hostname。

hostname類型	是否受影響	何時受影響	是否繼續閱讀文檔
在控制台或通過API建立執行個體時，hostname中有大寫字母	是	第一次重啟執行個體	是
在控制台或通過API建立執行個體時，hostname中全是小寫字母	否	不適用	否
hostname中有大寫字母，您登入執行個體後自行修改了hostname	否	不適用	是

- 修復方案：如果重啟執行個體後需要保留帶大寫字母的hostname時，可按如下步驟操作：
  1. 遠端連線執行個體。
  2. 查看現有的hostname。

```
[ root @ izbp193 ***** 3i161uynzz x ~]# hostname
izbp193 ***** 3i161uynzz x
```

3. 運行以下命令固化hostname。

```
hostnamectl l set - hostname -- static iZbp193 *****
3i161uynzz X
```

4. 運行以下命令查看更新後的hostname。

```
[ root @ izbp193 ***** 3i161uynzz x ~]# hostname
```

```
iZbp193 ***** 3i161uynzz X
```

- 下一步：如果您使用的是自訂鏡像，請更新cloud-init軟體至最新版本後，再次建立自訂鏡像。避免使用存在該問題的自訂鏡像建立新執行個體後發生同樣的問題。更多詳情，請參見[安裝cloud-init](#)和[使用執行個體建立自訂鏡像](#)。

#### Linux：pip操作時的逾時問題

- 問題描述：pip請求偶有逾時或失敗現象。
- 涉及鏡像：CentOS、Debian、Ubuntu、SUSE、OpenSUSE、Aliyun Linux
- 原因分析：阿里雲提供了以下三個pip源地址。其中，預設訪問地址為mirrors.aliyun.com，訪問該地址的執行個體需能訪問公網。當您的執行個體未分配公網IP時，會出現pip請求逾時故障。
  - （預設）公網：mirrors.aliyun.com
  - Virtual Private Cloud內網：mirrors.cloud.aliyuncs.com
  - 傳統網路內網：mirrors.aliyuncs.com



- 修復方案：您可採用以下任一方法解決該問題。

- 方法一

為您的執行個體分配公網IP，即為執行個體綁定一個Elastic IP Address (EIP)，詳情請參見[綁定彈性網卡](#)。

預付費執行個體還可通過升降配重新分配公網IP，詳情請參見[#unique\\_32](#)。

- 方法二

一旦出現pip響應延遲的情況，您可在ECS執行個體中運行指令碼 `fix_pypi . sh`，然後再重試pip操作。具體步驟如下：

1. 遠端連線執行個體。詳情請參見[#unique\\_33](#)。
2. 運行以下命令擷取指令檔。

```
wget http://image-offline.oss-cn-hangzhou.aliyuncs.com/fix/fix_pypi.sh
```

3. 運行指令碼。

■ Virtual Private Cloud執行個體：運行命令 `bash fix_pypi.sh "mirrors.cloud.aliyuncs.com"`。

■ 傳統網路執行個體：運行命令 `bash fix_pypi.sh "mirrors.aliyuncs.com"`。

4. 重試pip操作。

`fix_pypi . sh` 指令碼內容如下：

```
#!/bin/bash
function config_pip() {
    pypi_source=$1
    if [[ ! -f ~/.pydistutilrc.cfg ]]; then
        cat > ~/.pydistutilrc.cfg << EOF
        [ easy_install ]
        index-url = http://$pypi_source/pypi/simple/
        EOF
    else
        sed -i "s#index-url#index-url=http://$pypi_source/pypi/simple/#" ~/.pydistutilrc.cfg
    fi
    if [[ ! -f ~/.pip/pip.conf ]]; then
        mkdir -p ~/.pip
        cat > ~/.pip/pip.conf << EOF
        [ global ]
        index-url = http://$pypi_source/pypi/simple/
        [ install ]
        trusted-host = $pypi_source
        EOF
    else
        sed -i "s#index-url#index-url=http://$pypi_source/pypi/simple/#" ~/.pip/pip.conf
    fi
}
```

```
sed -i "s # trusted - host .*# trusted - host =$
pypi_source # " ~/. pip / pip . conf
fi
}
config_pip $ 1
```

#### Aliyun Linux 2：開啟核心選項CONFIG\_PARAVIRT\_SPINLOCK可能導致效能問題

- 問題描述：開啟核心選項 CONFIG\_PARAVIRT\_SPINLOCK 後，當ECS執行個體vCPU數量較多，且應用中有大量鎖競爭操作時，應用效能會受到較大影響（例如，Nginx應用的短串連處理能力會因此大幅下降），您可能會在應用中觀察到效能下降的問題。
- 涉及鏡像：Aliyun Linux 2
- 修復方案：核心選項 CONFIG\_PARAVIRT\_SPINLOCK 在Aliyun Linux 2上預設處於關閉狀態。如果您不確定如何處理核心問題，請勿開啟 CONFIG\_PARAVIRT\_SPINLOCK。

#### Aliyun Linux 2：核心特性透明大頁THP開關置為always可能會導致系統不穩定或效能下降

- 問題描述：在您的生產環境系統中，將透明大頁THP（Transparent Hugepage）開關置為 always ，可能會引發系統不穩定和效能下降等問題。
- 涉及鏡像：Aliyun Linux 2
- 修復方案：在Aliyun Linux2中，核心特性透明大頁（Transparent Hugepage ， THP ）開關已預設置為 madvise 。雖然在某些特殊情境下，例如運行一些效能基準測試套件時，開關置為 madvise 時，相對開關置為 always 時效能表現較差。但此類效能基準測試情境的測試結果不具備通用參考性。因此，除非您對THP有較為深入的理解，為了避免對系統造成其他競爭影響，建議您保持 madvise 選項不變。

#### Aliyun Linux 2：NFS v4.0版本中委託（Delegation）功能可能存在問題

- 問題描述：NFS委託（Delegation）功能在v4.0版本中可能存在問題。詳情請參見[NFS委託功能v4.0版本](#)。
- 涉及鏡像：Aliyun Linux 2
- 修復方案：使用NFS v4.0版本時，建議您不要開啟Delegation功能。如需從伺服器端關閉該功能，請參見[社區文檔](#)。

#### Aliyun Linux 2：NFS v4.1/4.2版本中存在缺陷可能導致應用程式無法退出

- 問題描述：在NFS的v4.1和v4.2版本中，如果您在程式中使用非同步I/O（AIO）方式下發請求，且在所有I/O返回之前關閉對應的檔案描述符，有一定幾率觸發活鎖，導致對應進程無法退出。
- 涉及鏡像：Aliyun Linux 2

- 修復方案：該問題已在核心4.19.30-10.a17及以上版本中修復。由於該問題出現機率極低，您可根據實際需要決定是否升級核心修復該問題。如需修復，請在終端執行 `sudo yum update kernel - y` 命令，升級完成後重啟系統即可。



注意：

- 升級核心版本可能會導致系統無法開機等風險，請謹慎操作。
- 升級核心前，請確保您已建立快照或自訂鏡像備份資料。詳情請參見[建立快照](#)或[使用執行個體建立自訂鏡像](#)。

### Aliyun Linux 2：Meltdown/Spectre漏洞修復會影響系統效能

- 問題描述：Aliyun Linux 2核心中，預設開啟了針對處理器硬體高危安全性漏洞Meltdown和Spectre的修復功能。由於此修復功能會影響系統效能，因此在常見的效能基準套件測試中，可能會觀察到不同程度的效能下降現象。
- 涉及鏡像：Aliyun Linux 2
- 修復方案：Meltdown和Spectre是英特爾晶片中發現的兩個高危漏洞，攻擊者可通過這兩個漏洞來訪問核心記憶體，從而竊取應用程式中的敏感資訊，因此，建議您不要在系統中關閉此類高危漏洞的修復功能。但如果您對系統效能有極高要求，可運行以下命令關閉該修復功能：

```
#追加 nopti nospectre_ v2 到核心啟動參數中
sudo sed -i 's /\( GRUB_CMDLI NE_LINUX =".*\)" /\ 1 nopti
nospectre_ v2 "/' / etc / default / grub
sudo grub2 - mkconfig - o / boot / grub2 / grub . cfg

#重啟系統
sudo reboot
```

## 4 自訂鏡像

### 4.1 建立自訂鏡像

#### 4.1.1 使用快照建立自訂鏡像

自訂鏡像是ECS執行個體系統硬碟某一時刻的快照。同時開通的Elastic Compute Service執行個體的配置既可以相同，也可以不同。示意圖如下。

您可以使用快照建立自訂鏡像，將快照的作業系統、資料環境資訊完整的包含在鏡像中。然後使用自訂鏡像建立多台具有相同作業系統和資料環境資訊的執行個體，非常方便的複製執行個體。

您也可以使用執行個體建立鏡像，請參考 [使用实例创建自定义镜像](#)。

為了增加快照建立自訂鏡像操作的安全性，請參考 [阿里雲自訂鏡像安全建議](#)。



说明:

- 建立的自訂鏡像不能跨地區使用。
- 通過自訂鏡像開通的ECS執行個體可以更換作業系統。更換系統後原來的自訂鏡像還可以繼續使用。請參見 [#unique\\_38](#)。
- 使用自訂鏡像開通的ECS執行個體可以升級CPU、記憶體、頻寬、磁碟等。
- 自訂鏡像功能不受付費模式限制，即不區分預付費和隨用隨付。預付費ECS執行個體的自訂鏡像，可以用於開通隨用隨付的ECS執行個體；反之亦然。
- 用於建立自訂鏡像的ECS執行個體到期或資料釋放後（即用於快照的系統硬碟到期或釋放），建立的自訂鏡像不會受影響，使用自訂鏡像開通的ECS執行個體也不會受影響。但自動快照則會隨著ECS執行個體釋放而被清除。

#### Linux執行個體注意事項

- 在使用Linux執行個體的系統硬碟建立自訂鏡像時，不要在 `/ etc / fstab` 檔案中載入資料盤的資訊，否則使用該鏡像建立的執行個體無法啟動。
- 強烈建議您在製作自訂鏡像前 `umount` Linux執行個體上掛載的所有檔案系統，然後再對系統硬碟打快照並建立自訂鏡像，否則有可能造成以該自訂鏡像建立的ECS執行個體不能啟動或使用。
- 請勿隨意升級核心或作業系統版本。
- 請勿調整系統硬碟分區。系統硬碟目前只支援單個根分區。

- 請檢查系統硬碟使用剩餘空間，確保系統硬碟沒有被寫滿。
- 請勿修改關鍵系統檔案，如 `/sbin`、`/bin`、`/lib` 目錄等。
- 請勿修改預設登入使用者名 `root`。

## 操作步驟

1. 登入 [雲端服務器管理主控台](#)。
2. 選擇地區。
3. 在左側導覽列裡，單擊 **執行個體**。
4. 找到目標執行個體，單擊執行個體ID，或者在 **操作列**，單擊 **管理**。
5. 在左側導覽列裡，單擊 **本執行個體快照**。找到目標系統硬碟，在 **操作列**，單擊 **建立自訂鏡像**。

快照的磁碟屬性必須是系統硬碟。資料盤不能用於建立自訂鏡像。

您也可以通過 **快照 > 快照列表**，選擇一個磁碟屬性為系統硬碟的快照，再 **建立自訂鏡像**。

6. 在彈出的 **建立自訂鏡像** 對話方塊中，完成以下操作：

- 確認快照ID。
- 指定自訂鏡像的名稱和描述。
- (可選) 如果您希望在建立的鏡像中同時包含資料盤的資訊，應該選擇 **添加資料盤快照**，並單擊 **增加** 來添加資料盤。



### 說明：

請將資料盤中的敏感性資料刪除之後再建立自訂鏡像，避免資料安全隱患。

如果快照ID為空白，則該磁碟會作為空白盤建立，預設容量為5 GiB。

如果選擇了快照，則磁碟容量為快照的容量。

7. 單擊 **建立**。自訂鏡像建立成功。您可以單擊左側導航中的 **鏡像**，然後查看剛建立的鏡像。

## Linux鏡像FAQ

如何 `umount` 和刪除disk table裡的資料？

假設 `/dev/hda5` 已經掛載在 `/mnt/hda5` 上，用以下三條命令均可卸載掛載的檔案系統：

```
umount / dev / hda5
umount / mnt / hda5
umount / dev / hda5 / mnt / hda5
```

`/etc/fstab` 是Linux下比較重要的設定檔，它包含了系統在啟動時掛載檔案系統和存放裝置的詳細資料。如果不想在執行個體啟動時掛載指定分區，需要在這個檔案裡面刪除對應的行，刪除下述語句可以在啟動的時候斷開xvdb1：`/ dev / xvdb1 / leejd ext4 defaults 0 0`。

如何確認資料盤已經卸載，並可以開始建立自訂鏡像？

需要確認fstab檔案裡面對應的自動掛載資料盤分區語句行已經刪除。

使用 `mount` 命令可以查看所有裝置的掛載資訊，請確認執行結果中不包含對應的資料盤分區資訊。

## 相關設定檔

自訂遠端連線連接埠的詳細操作，請參見 伺服器預設遠程連接埠修改。

設定檔	配置說明	修改該設定檔的風險
<code>/etc/issue*</code> , <code>/etc/*-release</code> , <code>/etc/*_version</code>	系統發行版資訊設定檔	修改 <code>/etc/issue*</code> 會導致系統發行版無法被正常識別，導致系統建立失敗。
<code>/boot/grub/menu.lst</code> , <code>/boot/grub/grub.conf</code>	系統引導啟動設定檔	修改 <code>/boot/grub/menu.lst</code> 會導致核心無法正確載入，導致系統無法啟動。
<code>/etc/fstab</code>	系統啟動掛載分區設定檔	修改該檔案會導致異常分區無法被載入，導致系統無法啟動。
<code>/etc/shadow</code>	系統密碼相關設定檔	修改該檔案為唯讀會導致無法修改密碼檔案，導致系統建立失敗。
<code>/etc/selinux/config</code>	系統安全性原則設定檔	修改 <code>/etc/selinux/config</code> 開啟SELinux導致系統無法啟動。

## 4.1.2 使用執行個體建立自訂鏡像

您可以基於執行個體建立一個自訂鏡像，把執行個體中的所有磁碟，包括系統硬碟和資料盤中的資料，完整地複製到自訂鏡像中。

建立自訂鏡像的過程中，該執行個體的每塊磁碟都自動建立一個快照，這些快照組合起來構成一個自訂鏡像，如下圖所示。

您也可以基於快照建立自訂鏡像。請參考[使用快照建立自訂鏡像](#)。

### 操作須知

- 請將執行個體中的敏感性資料刪除之後再建立自訂鏡像，避免資料安全隱患。
- 建立鏡像的過程中，不要改變執行個體的狀態，不要停止、啟動或者重啟執行個體，避免建立失敗。
- 使用帶資料盤的自訂鏡像建立執行個體時，預設根據鏡像建立資料盤；如果建立執行個體的同時建立了資料盤，則預設根據掛載點情況使用該鏡像的資料盤快照建立執行個體的資料盤。
- 支援匯出帶資料盤的自訂鏡像。
- 更換系統硬碟時，不能使用帶資料盤的自訂鏡像。

### 操作步驟

1. 登入 [ECS 管理主控台](#)。
2. 在執行個體列表頁面頂部，選擇目標執行個體所在的地區。
3. 單擊左側導覽列中的 執行個體。
4. 找到需要的執行個體。單擊列表最右側的 更多 > 建立自訂鏡像。
5. 輸入鏡像名稱和描述資訊。
6. 單擊 建立。

所有磁碟的快照全部建立結束後，鏡像才能使用。請耐心等待。

### 後續操作

成功建立自訂鏡像後，您可能想 [#unique\\_41](#)。

## 4.1.3 使用Packer建立自訂鏡像

開源工具 [Packer](#) 能夠便捷建立自訂鏡像。它能運行在大多數主流作業系統上。參閱本文安裝並使用Packer，輕鬆建立自訂鏡像。

**说明:**

本文描述僅適用於運行Linux作業系統的伺服器。

## 1. 安裝Packer

您可以進入 [Packer 官網下載頁面](#)，選擇下載與您作業系統對應的版本，參閱以下步驟或者訪問 [Packer 官方安裝說明](#) 安裝 Packer。

1. 串連並登入到Linux伺服器。如果您的伺服器為ECS Linux執行個體，可參閱文檔 [使用使用者名密碼驗證串連 Linux 執行個體](#)。
2. 執行命令 `cd /usr/local/bin` 進入/usr/local/bin目錄。

**说明:**

/usr/local/bin 目錄為環境變數目錄，您可以將 Packer 安裝到該目錄下或其他已添加到環境變數的目錄下。

3. 執行命令 `wget https://releases.hashicorp.com/packer/1.1.1/packer_1.1.1_linux_amd64.zip` 獲取Packer安裝包。您可以訪問 [Packer 下載頁面](#)獲取其他版本Packer安裝包。
4. 執行命令 `unzip packer_1.1.1_linux_amd64.zip` 解壓檔案。
5. 執行命令 `packer -v` 驗證Packer安裝狀態。若Linux伺服器返回Packer版本號碼，表示您已正確安裝Packer。若Linux伺服器提示command not found表示Packer未正確安裝。

## 2. 定義Packer模板

**说明:**

使用Packer建立自訂鏡像時，需要建立一個JSON格式的模板檔案。在該模板檔案中，您需要指定建立自訂鏡像的 [Alicloud Image Builder（產生器）](#) 和 [Provisioners（配置器）](#)。Packer 具有多種配置器，可用於配置自訂鏡像的內容生成方式，以下以常用的 [Shell](#) 配置器為例，定義 Packer 模板。

在Linux伺服器中建立名為alicloud的json檔案並粘貼以下內容。

```
{
  "variables": {
    "access_key": "{{ env `ALICLOUD_ACCESS_KEY` }}",
    "secret_key": "{{ env `ALICLOUD_SECRET_KEY` }}"
  },
  "builders": [
    {
      "type": "alicloud-ecs",
      "access_key": "{{ user `access_key` }}",
      "secret_key": "{{ user `secret_key` }}"
    }
  ]
}
```



```

" region ":" cn - beijing ",
" image_name ":" packer_base_image ",
" source_image ":" centos_7_0_2_64_20G_alibase_20170818.vhd ",
" ssh_username ":" root ",
" instance_type ":" ecs.n1.tiny ",
" internet_charge_type ":" PayByTraffic ",
" io_optimized ":" true "
}],
" provisioners ":[{
" type ":" shell ",
" inline ":[
" sleep 30 ",
" yum install redis.x86_64 -y "
]
}]
}

```



说明:

您需要自訂以下參數值。

參數	描述
access_key	您的AccessKeyID。更多詳情，參閱 <a href="#">建立AccessKey</a> 。   说明: 由於AccessKey許可權過大，為防止錯誤操作，建議您創建RAM用戶，並使用RAM子帳號 <a href="#">建立AccessKey</a>
secret_key	您的AccessKeySecret。更多詳情，參閱 <a href="#">建立AccessKey</a>
region	建立自訂鏡像時使用臨時資源的地域
image_name	自訂鏡像的名稱
source_image	基礎鏡像的名稱，可以從阿里雲公共鏡像列表獲得
instance_type	建立自訂鏡像時生成的臨時執行個體的類型
internet_charge_type	建立自訂鏡像時臨時執行個體的公網頻寬付費類型
provisioners	建立自訂鏡像時使用的 <a href="#">Packer 配置器</a> 類型

### 3. 使用Packer建立自訂鏡像

指定Packer模板檔案生成自訂鏡像：

1. 匯入您的AccessKeyID：運行 `export ALICLOUD_ACCESS_KEY =您的AccessKeyID`。
2. 匯入您的AccessKeySecret：運行 `export ALICLOUD_SECRET_KEY =您的AccessKeySecret`。

3. 執行命令 `packer build alicloud . json` 建立自訂鏡像。

樣本運行結果如下，以下樣本將建立含Redis的自訂鏡像：

```
alicloud - ecs output will be in this color .
==> alicloud - ecs : Prevalidating alicloud image name ...
alicloud - ecs : Found image ID : centos_7_0_2_64_20G_a
libase_201_70818 . vhd
==> alicloud - ecs : Start creating temporary keypair :
packer_59e_44f40 - c8d6 - 0ee3 - 7fd8 - b1ba08ea94_b8
==> alicloud - ecs : Start creating alicloud vpc
-----
==> alicloud - ecs : Provisioning with shell script : / var
/ folders / 3q / w38xx_js6c_l6k5mwkrqs_nw7w0000gn / T / packer -
shell25746_6182
alicloud - ecs : Loaded plugins : fastestmirror
-----
alicloud - ecs : Total 1.3 MB / s | 650 kB 00 : 00
alicloud - ecs : Running transaction check
-----
==> alicloud - ecs : Deleting temporary keypair ...
Build 'alicloud - ecs' finished .
==> Builds finished . The artifacts of successful builds
are :
--> alicloud - ecs : Alicloud images were created :
cn - beijing : m - 2ze12578be_10a4ovs6r9
```

下一步

[#unique\\_41](#)

參考連結

- 訪問阿里雲GitHub Packer倉庫 [packer-provider](#) 獲取更多資訊。
- 參閱 [Packer官方文檔](#) 瞭解更多Packer使用詳情。

## 4.1.4 使用 Packer 建立並匯入本地鏡像

**Packer** 是一個便捷的建立本地鏡像的開源工具，能運行在大多數主流作業系統上。

自行建立本地鏡像並上傳到雲平台是一個非常繁複的過程，此處示範了如何在 Ubuntu 16.04 伺服器中建立 CentOS 6.9 本地鏡像並上傳到阿里雲雲平台，如果您需要製作其他動作系統類別型的鏡像，請根據需要自訂 Packer 模板。

前提條件

- 您已經 [建立 AccessKey](#)，用於輸出到設定檔 `centos.json` 裡。



说明:

由於 AccessKey 許可權過大，為防止資料泄露，建議您 [创建 RAM 用户](#)，並使用 RAM 使用者子帳號 [建立 AccessKey](#)。

- 您已經 [开通OSS服务](#)，便於上傳鏡像檔案到雲平台。

## 使用樣本

1. 運行 `grep "(svm | vmx)" /proc/cpuinfo` 確保您的原始伺服器或者虛擬機器是否支援 KVM。當返回下列資訊時，表示支援 KVM。

```
pat pse36 clflush dts acpi mmx fxsr sse sse2
ss ht tm pbe syscall nx pdpe1gb rdtscp lm
constant_tsc art arch_perfmon on pebs bts rep_good
nopl xtopology nonstop_tsc c_aperfperf tsc_known_freq
pni pclmulqdq dtes64 monitor ds_cpl vmx est tm2
ssse3 sdbg fma cx16 xtpr pdcm pcid sse4_1 sse4_2
x2apic movbe popcnt tsc_deadline_timer aes xsave avx
f16c rdrand lahf_lm abm 3dnowprefetch tch epb intel_pt
tpr_shadow vnmi flexpriority ept vpid fsgsbase
tsc_adjust bmi1 avx2 smep bmi2 erms invpcid mpx
rdseed adx smap clflushopt xsaveopt xsavec xgetbv1
xsaves dtherm ida arat pln pts hwp hwp_notify
hwp_act_window hwp_epp
flags : fpu vme de pse tsc msr pae mce cx8 apic
sep mtrr pge mca cmov
```

2. 若您的原始伺服器或者虛擬機器支援 KVM，運行以下命令安裝 KVM。

```
sudo apt-get install qemu-kvm qemu-virt-manager
virt-viewer libvirt-bin bridge-utils #安裝 KVM 及相關控
制項。
sudo virt-manager #開啟 virt-manager。
```

如果可以通過 GUI 建立虛擬機器，表示您已經成功安裝 KVM。

3. 安裝 Packer。

您可以參閱 [使用Packer创建自定义镜像](#) 部分內容安裝 Packer。

4. 依次執行以下命令定義 Packer 模板：



说明：

以下樣本製作的本地鏡像基於 CentOS 6.9 作業系統，如果您需要建立其他類型作業系統的鏡像，您可以根據需要自訂 Packer 模板設定檔 `centos.json`。

```
cd /user/local #切换所在目錄。
wget https://raw.githubusercontent.com/alibaba/packer-provider/master/examples/alicloud/local/centos.json
#下載阿里雲官方發行的 centos.json 設定檔。
wget https://raw.githubusercontent.com/alibaba/packer-provider/master/examples/alicloud/local/http/centos-6.9/ks.cfg
#下載阿里雲官方發行的 ks.cfg 設定檔。
mkdir -p http/centos-6.9 #建立目錄。
```

```
mv ks . cfg http / centos - 6 . 9 / #移動檔案 ks . cfg 到目錄
http / centos - 6 . 9 中。
```

5. 依次執行以下命令建立本地鏡像。

```
export ALICLOUD_ACCESS_KEY =您的 AccessKeyId #匯入您的
AccessKeyId 。
export ALICLOUD_SECRET_KEY =您的 AccessKeySecret #匯入您的
AccessKeySecret 。
packer build centos . json #建立本地鏡像。
```

樣本運行結果如下：

```
qemu output will be in this color .
==> qemu : Downloading or copying ISO
qemu : Downloading or copying : http :// mirrors . aliyun
.com / centos / 6 . 9 / isos / x86_64 / CentOS - 6 . 9 - x86_64 -
minimal . iso
.....
==> qemu : Running post - processor : alicloud - import
qemu ( alicloud - import ) : Deleting import source https
:// oss - cn - beijing . aliyuncs . com / packer / centos_x86 _64
Build ' qemu ' finished .
==> Builds finished . The artifacts of successful builds
are :
--> qemu : Alicloud images were created :
cn - beijing : XXXXXXXX
```

6. 等待一段時間後，在對應地區的 [ECS 控制台鏡像詳情頁](#) 查看上傳結果，如樣本中的華北二。

### 後續操作

您可以使用該自訂鏡像建立 ECS 執行個體，具體步驟參閱 [#unique\\_48](#)。

### 自訂 Packer 模板

以上 [使用樣本](#) 製作的本地鏡像基於 CentOS 6.9 作業系統，如果您需要建立其他類型作業系統的鏡像，您可以根據需要自訂 Packer 模板。

例如，以下 JSON 檔案是基於 Packer 模板產生的，用於建立 CentOS 6.9 鏡像。

```
{ " variables " : {
  " box_basename " : " centos - 6 . 9 " ,
  " build_time_stamp " : " {{ isotime ` 2006010215 0405 ` }}" ,
  " cpus " : " 1 " ,
  " disk_size " : " 4096 " ,
  " git_revision " : " __unknown__ git_revision__ " ,
  " headless " : "" ,
  " http_proxy " : " {{ env ` http_proxy ` }}" ,
  " https_proxy " : " {{ env ` https_proxy ` }}" ,
  " iso_checksum_type " : " md5 " ,
  " iso_checksum " : " af4a1640c0 c6f348c6c4 1f1ea9e192 a2 " ,
  " iso_name " : " CentOS - 6 . 9 - x86_64 - minimal . iso " ,
  " ks_path " : " centos - 6 . 9 / ks . cfg " ,
  " memory " : " 512 " ,
```

```

" metadata ": " floppy / dummy_meta_data . json ",
" mirror ": " http :// mirrors . aliyun . com / centos ",
" mirror_directory ": " 6 . 9 / isos / x86_64 ",
" name ": " centos - 6 . 9 ",
" no_proxy ": "{{ env ` no_proxy ` }}",
" template ": " centos - 6 . 9 - x86_64 ",
" version ": " 2 . 1 . TIMESTAMP "
},
" builders ":[
{
" boot_command ": [
" text ks = http ://{{ . HTTPIP }}:{{ . HTTPPort }}/{{ user `
ks_path `}}"
],
" boot_wait ": " 10s ",
" disk_size ": "{{ user ` disk_size `}}",
" headless ": "{{ user ` headless `}}",
" http_directory ": " http ",
" iso_checksum ": "{{ user ` iso_checksum `}}",
" iso_checksum_type ": "{{ user ` iso_checksum_type `}}",
" iso_url ": "{{ user ` mirror `}}/{{ user ` mirror_directory `}}/
{{ user ` iso_name `}}",
" output_directory ": " packer -{{ user ` template `}}- qemu ",
" shutdown_command ": " echo ' vagrant ' | sudo - S / sbin / halt
- h - p ",
" ssh_password ": " vagrant ",
" ssh_port ": 22 ,
" ssh_username ": " root ",
" ssh_wait_timeout ": " 10000s ",
" type ": " qemu ",
" vm_name ": "{{ user ` template `}}. raw ",
" net_device ": " virtio - net ",
" disk_interface ": " virtio ",
" format ": " raw "
}
],
" provisioners ": [{
" type ": " shell ",
" inline ": [
" sleep 30 ",
" yum install cloud - util cloud - init - y "
]
}],
" post - processors ":[
{
" type ":" alicloud - import ",
" oss_bucket_name ": " packer ",
" image_name ": " packer_import ",
" image_os_type ": " linux ",
" image_platform ": " CentOS ",
" image_architecture ": " x86_64 ",
" image_system_size ": " 40 ",
" region ":" cn - beijing "
}
]
}
}

```

### Packer 產生器參數說明

**使用樣本** 中使用的產生器為 QEMU Builder，該產生器用於建立虛擬機器鏡像。其必需參數如下所示。

必需參數	類型	描述
iso_checksum	string	作業系統的 ISO 檔案的校正和。Packer 在啟動掛載了 ISO 的虛擬機器之前檢查該參數。至少指定 iso_checksum 或者 iso_checksum_url 參數；當您指定了參數 iso_checksum 時，參數 iso_checksum_url 的值被自動忽略。
iso_checksum_type	string	已指定的作業系統的 ISO 檔案的校正和類型。取值範圍： <ul style="list-style-type: none"> <li>· none：忽略校正和過程。不推薦指定該值</li> <li>· md5</li> <li>· sha1</li> <li>· sha256</li> <li>· sha512</li> </ul>
iso_checksum_url	string	一個包含了作業系統 ISO 檔案校正和的檢驗和檔案的連結（URL），樣式為 GNU 或者 BSD。您需要至少指定 iso_checksum 或者 iso_checksum_url 參數；當您指定了參數 iso_checksum 時，參數 iso_checksum_url 的值被自動忽略。

iso_url	string	<p>一個指向 ISO 並包含安裝鏡像的 URL。這個 URL 可以是一個 HTTP 連結,也可以是一個檔案路徑:</p> <ul style="list-style-type: none"> <li>· 如果是 HTTP 連結, Packer 從 HTTP 連結下載檔案並緩衝運行。</li> <li>· 如果是 IMG 或者 QCOW2 檔案連結, QEMU 直接在檔案基礎上啟動。當您指定了檔案路徑時,請將參數 disk_image 置為 true 。</li> </ul>
headless	boolean	<p>Packer 預設通過啟動圖形化虛擬機器介面構建 QEMU 虛擬機器。當您將 headless 置為 True 時,將啟動一個沒有控制台的虛擬機器。</p>

其餘選擇性參數可參閱 Packer [QEMU Builder](#) 頁面。

#### Packer 配置器部分參數說明

[使用樣本](#) 中使用的配置器中包含了 Post-Processor 模組,該模組實現自動化上傳本地鏡像到 ECS 雲平台。Post-Processor 模組的必需參數如下所示。

必需參數	類型	描述
access_key	string	您的 AccessKeyID。由於 AccessKey 許可權過大,為防止資料泄露,建議您 <a href="#">創建 RAM 用戶</a> ,並使用 RAM 使用者子帳號 <a href="#">建立 AccessKey</a> 。
secret_key	string	您的 AccessKeySecret。由於 AccessKey 許可權過大,為防止資料泄露,建議您 <a href="#">創建 RAM 用戶</a> ,並使用 RAM 使用者子帳號 <a href="#">建立 AccessKey</a> 。
region	string	本地鏡像上傳的地區,如本樣本中的 cn-beijing。請參閱 <a href="#">地區和可用性區域</a> 查看更多地區資訊。

image_name	string	您的本地鏡像的名稱。 <ul style="list-style-type: none"> <li>由 [2, 128] 位英文或中文字元組成。</li> <li>必須以大小寫字母或中文字元開始。</li> <li>可以包含數字、底線 (_) 或者連字號 (-)。</li> <li>不能以 http:// or https:// 開頭。</li> </ul>
oss_bucket_name	string	您的 OSS Bucket 名稱。若您指定了一個不存在的 Bucket 名稱，Packer 上傳鏡像時自動建立一個同名 Bucket。
image_os_type	string	鏡像類型，取值範圍： <ul style="list-style-type: none"> <li>linux</li> <li>windows</li> </ul>
image_platform	string	鏡像發行版本，如本樣本中的 CentOS。
image_architecture	string	鏡像發行版本的架構平台，取值範圍： <ul style="list-style-type: none"> <li>i386</li> <li>x86_64</li> </ul>
format	string	鏡像的格式，取值範圍： <ul style="list-style-type: none"> <li>RAW</li> <li>VHD</li> </ul>

其餘選擇性參數可參閱 [Packer Alicloud Post-Processor](#) 頁面。

#### 下一步

您可以使用該鏡像建立 ECS 執行個體，參閱 [#unique\\_48](#)。

#### 參考連結

- 您可以參閱 [Packer 官方文檔](#) 瞭解 Packer 使用詳情。
- 您可以訪問 GitHub Packer 倉庫 [packer](#) 擷取更多資訊。
- 您可以訪問 GitHub Alibaba 倉庫 [opstools](#) 擷取更多資訊。
- 您可以訪問 GitHub Alibaba&Packer 倉庫 [packer-provider](#) 擷取更多資訊。
- 您可以參閱 [Anaconda Kickstart](#) 瞭解設定檔 ks.cfg 的更多資訊。



## 4.2 管理自訂鏡像

建立了自訂鏡像後，ECS建議您合理管理自己的鏡像資源。您可以重新修改自訂鏡像的名稱和描述，或者刪除不再需要的自訂鏡像以確保鏡像額度不超過使用限制。

### 修改自訂鏡像資訊

在管理主控台上修改自訂鏡像名稱和描述的步驟如下：

1. 登入 [ECS管理主控台](#)。
2. 在左側導覽列中，選擇 **快照和鏡像** > **鏡像**。
3. 選擇地區。
4. 找到需要編輯的自訂鏡像。
5. 單擊 **表徵圖**，重新填寫鏡像名稱。
6. 單擊 **編輯描述**，在彈出的對話方塊中：
  - 自訂鏡像描述：重新填寫鏡像描述。
  - （可選）標籤：重新選擇標籤。
7. 單擊 **確定** 完成修改自訂鏡像描述資訊。

您也可以使用ECS API [ModifyImageAttribute](#) 修改自訂鏡像資訊。

### 刪除自訂鏡像

在管理主控台上刪除自訂鏡像的步驟如下：

1. 登入 [ECS管理主控台](#)。
2. 在左側導覽列中，選擇 **快照和鏡像** > **鏡像**。
3. 選擇地區。
4. 選中一份或者多份需要刪除的自訂鏡像，單擊 **刪除**。
5. 在彈出的對話方塊中，選擇刪除方式：
  - **刪除**：正常刪除自訂鏡像。
  - **強制移除**：強行刪除自訂鏡像，並需要勾選 **確定要強制移除**。



**注意：**

強制移除鏡像後，使用該自訂鏡像建立的 ECS 執行個體無法 [#unique\\_51](#)。

6. 單擊 **確定** 刪除自訂鏡像。

您也可以使用 ECS API `DeleteImage` 刪除自訂鏡像。

## 4.3 匯入鏡像

### 4.3.1 鏡像規範偵查工具

ECS 支援使用自訂鏡像建立執行個體。該自訂鏡像可以是您基於線下伺服器、虛擬機器或者其他雲平台的雲主機建立的。您匯入的鏡像需要滿足一定鏡像要求，更多詳情，請參閱 [匯入鏡像注意事項](#)。為縮短鏡像製作及建立執行個體的周期，我們推薦您使用 ECS 鏡像規範偵查工具，本文簡稱 偵查工具，製作符合規範的鏡像。偵查工具能基於伺服器環境檢測各項配置指標、定位不合規原因、產生 TXT 和 JSON 格式檢測報告並給出解決意見。

#### 使用限制

目前偵查工具僅支援檢測 Linux 類型鏡像，包括 Ubuntu、CentOS、Debian、RedHat、SLES (SUSE Linux Enterprise Server)、OpenSUSE、FreeBSD、CoreOS 和 Others Linux 發行版本。

#### 使用樣本

以 CentOS 7.4 64 位元伺服器為例。

1. 登入您的伺服器、虛擬機器或者其他雲平台的雲主機。
2. [下載](#) 偵查工具。
3. 使用 root 許可權運行 `image_chec k`，避免偵查工具無法讀取有許可權控制的設定檔。

```
chmod +x image_chec k
sudo image_chec k -p [目標路徑]
```



#### 说明:

您可以使用 `-p [目標路徑]` 指定檢測報告的產生路徑，不指定則預設輸出在偵查工具所在的路徑。

4. 等待偵查工具檢測系統配置。

```
Begin check your system ...
The report is generating .
-----
The information you need to input when you
import your image to Alibaba Cloud website :
Current system : CentOS # 系統資訊 1 : 伺服器作業系統
Architectu re : x86_64 # 系統資訊 2 : 系統架構
System disk size : 42 GB # 系統資訊 3 : 伺服器系統硬碟
容量
-----
```

```

# 檢測項
Check driver [ OK ]
Check shadow file authority [ OK ]
Check security [ OK ]
Check qemu - ga [ OK ]
Check network [ OK ]
Check ssh [ OK ]
Check firewall [ OK ]
Check filesystem [ OK ]
Check device id [ OK ]
Check root account [ OK ]
Check password [ OK ]
Check partition table [ OK ]
Check lvm [ FAILED ]
Check lib [ OK ]
Check disk size [ OK ]
Check disk use rate [ WARNING ]
Check inode use rate [ OK ]
-----
15 items are OK
1 items are failed
1 items are warning
-----
The report is generated : / root / image_chec k_report_2
018 - 05 - 14_18 - 18 - 10 . txt
Please read the report to check the details

```

5. 在相應位置查看檢測報告。報告格式為 `image_chec k_report_ 日期 _ 時間. txt` 或 `image_chec k_report . json`。

## 檢測項

為避免使用您的自訂鏡像建立的 ECS 執行個體功能不全，偵查工具主要檢測伺服器中以下配置項。

檢測項	不合規後果	合規建議
driver	無法正常啟動 ECS 執行個體	安裝虛擬化驅動，如 <a href="#">安裝 virtio 驅動</a>
/etc/shadow	無法修改密碼檔案，進而無法根據自訂鏡像建立 ECS 執行個體	不要使用 <code>chattr</code> 命令鎖定 /etc/shadow 檔案
SELinux	無法正常啟動 ECS 執行個體	不要通過修改 /etc/selinux/config 開啟 SELinux
qemu-ga	ECS 所需要的部分服務不可用，執行個體功能不全	卸載 qemu-ga
network	ECS 執行個體網路功能不穩定	關閉或刪除 Network Manager 並開啟 network 服務，新版 Linux 推薦使用 Network Manager 和 nmcli 命令設定網路

檢測項	不合規後果	合規建議
ssh	無法從控制台 <a href="#">遠端連線</a> ECS 執行個體	開啟 SSH 服務，不要設定 PermitRootLogin
firewall	系統無法自動設定您的 ECS 執行個體環境	關閉防火牆 Iptables、Firewalld、IPFILTER (IPF)、IPFIREWALL (IPFW) 或 PacketFilter (PF)
file system	無法 <a href="#">擴容磁碟</a>	使用 xfs、ext3、ext4 檔案系統，容許 ext2、ufs、udf。ext4 檔案系統不要使用 64 bit feature
root	無法使用使用者名和密碼遠端連線 ECS 執行個體	保留 root 帳號
passwd	ECS 執行個體無法新增使用者	保留 passwd 命令，或重新安裝 passwd
分區表	無法正常啟動 ECS 執行個體	採用 MBR 分區
Logical Volume Manager (LVM)	無法正常啟動 ECS 執行個體	更換為其他分區服務
/lib	無法自動化配置 ECS 執行個體	/lib 和 /lib64 的指向位置不能為絕對路徑，修改連結檔案 /lib 和 /lib64 的指向位置為相對路徑
system disk	N/A	增加系統硬碟容量，系統硬碟空間最佳為 40 GiB 至 500 GiB。並在匯入鏡像時根據鏡像的虛擬檔案大小而非使用容量配置系統硬碟容量
disk usage	無法為 ECS 執行個體安裝必要的驅動或服務	保持足夠的磁碟空間
inode usage	無法為 ECS 執行個體安裝必要的驅動或服務	保持足夠的 inode 資源

偵查工具會根據檢測項給出 `OK`、`FAILED` 或者 `WARNING` 檢測結果。

- `OK`：檢測項均符合要求。
- `FAILED`：檢測項不符合要求，使用該自訂鏡像建立的 ECS 執行個體無法正常啟動。建議您修複報錯項後再製作鏡像，以提高執行個體啟動效率。

- **WARNING**：檢測項不符合要求，使用該自訂鏡像建立的 ECS 執行個體可以安全啟動但 ECS 無法通過有效途徑幹預您的執行個體配置。您可以選擇立即修複，也可以暫時忽略問題前往製作鏡像環節。

## 輸出項

檢測系統內容後，會輸出 TXT 和 JSON 兩種格式檢測報告。輸出路徑可以使用 `-p [目標路徑]` 指定，不指定則預設輸出在偵查工具所在的路徑。

- TXT 格式報告名稱為 `image_check_report_日期_時間.txt`，報告內容包括伺服器配置資訊以及檢測結果。以 CentOS 7.4 64 位伺服器為例：

```
The information you need to input when you import
your image to Alibaba Cloud Website :
Current system is : CentOS           #伺服器作業系統
Architecture : x86_64               #系統架構
System disk size : 42 GB            #伺服器系統硬碟容量
-----
Check driver                         #檢測項名稱
Pass : kvm drive is exist           #檢測結果
Alibaba Cloud supports kvm virtualization
technology
We strongly recommend installing kvm driver .
```

- JSON 格式報告名稱為 `image_check_report.json`，報告內容包括伺服器配置資訊以及檢測結果。以 CentOS 7.4 64 位伺服器為例：

```
" platform " : " CentOS " ,           \\伺服器作業系統
" os_big_version " : " 7 " ,          \\作業系統版本號碼 (高位)
" os_small_version " : " 4 " ,        \\作業系統版本號碼 (低位)
" architecture " : " x86_64 " ,       \\系統架構
" system_disk_size " : " 42 " ,        \\伺服器系統硬碟容量
" version " : " 1 . 0 . 2 " ,          \\偵查工具版本
" time " : " 2018 - 05 - 14_19 - 18 - 10 " , \\檢測時間
" check_items " : [ {
  " name " : " driver " ,              \\檢測項名稱
  " result " : " OK " ,                \\檢測結果
  " error_code " : " 0 " ,             \\錯誤碼
  " description " : " Pass : kvm driver exists . " , \\描述
  " comment " : " Alibaba Cloud supports kvm virtualization
technology . We strongly recommend installing kvm
driver ."
} ]
}
```

## 後續步驟

1. 查看 [匯入鏡像注意事項](#)
2. [安裝 virtio 驅動](#)
3. (可選) [轉換鏡像格式](#)
4. [匯入鏡像](#)
5. [使用自訂鏡像建立執行個體](#)

## 4.3.2 匯入鏡像必讀

為保證自訂鏡像的可用性和提高鏡像匯入效率，請在匯入鏡像前閱讀本文中提及的阿里雲ECS自訂鏡像限制條件。

### Windows作業系統鏡像

#### 系統限制

- 請確認檔案系統的完整性。
- 請勿修改關鍵系統檔案。
- 請檢查系統硬碟的剩餘空間，確保系統硬碟沒有被寫滿。
- 請根據鏡像的虛擬磁碟大小而非使用容量配置匯入的系統硬碟大小，系統硬碟容量範圍支援40GiB–500GiB。
- 關閉防火牆，並允許存取RDP 3389連接埠。
- administrator帳號的登入密碼必須為8–30個字元，同時包含大小寫英文字母、數字和特殊符號中的三類字元。特殊符號可以是( ) ` ~ ! @ # \$ % ^ & \* - \_ + = | { } [ ] : ; ' < > , . ? /。其中，不能以斜線號 (/) 為密碼首字元。

#### 不支援項

- 不支援ISO鏡像檔案。請線上下環境通過VirtualBox等工具製作後，轉換成阿里雲ECS支援的鏡像格式RAW、VHD或qcow2等再匯入。
- 不支援在鏡像中安裝qemu-ga，否則會導致ECS所需要的部分服務不可用。
- 不支援Windows XP、專業版和企業版Windows 7、Windows 8和Windows 10。

#### 支援項

- 支援多分區系統硬碟。
- 支援NTFS檔案系統，支援MBR分區。
- 支援RAW、qcow2和VHD格式鏡像。匯入其他格式的鏡像前，請先[轉換鏡像格式](#)後再匯入。
- 支援匯入包含以下作業系統版本的鏡像：
  - Microsoft Windows Server 2016
  - Microsoft Windows Server 2012 R2
  - Microsoft Windows Server 2012
  - Microsoft Windows Server 2008 R2
  - Microsoft Windows Server 2008
  - 含Service Pack 1 (SP1) 的Windows Server 2003或更高版本

## Linux作業系統鏡像

### 系統限制

- 請確認檔案系統的完整性。
- 請勿修改關鍵系統檔案，如 `/sbin`、`/bin` 和 `/lib *` 等目錄。
  - 請勿修改 `/etc / issue *`，否則ECS無法正常識別系統發行版，從而建立系統失敗。
  - 請勿修改 `/boot / grub / menu.lst`，否則無法啟動ECS執行個體。
  - 請勿修改 `/etc / fstab`，否則無法載入異常分區從而無法啟動ECS執行個體。
  - 請勿修改 `/etc / shadow` 為唯讀，否則將導致無法修改密碼檔案，從而建立系統失敗。
  - 請勿修改 `/etc / selinux / config` 開啟SELinux，否則系統無法啟動。
- 請檢查系統硬碟的剩餘空間，確保系統硬碟沒有被寫滿。
- 關閉防火牆，並允許存取SSH 22連接埠。
- 開啟DHCP (Dynamic Host Configuration Protocol) 服務。
- 安裝XEN或者KVM虛擬化驅動。更多詳情，請參見[安裝virtio驅動](#)。
- [安裝cloud-init](#)，以保證能成功配置hostname、NTP源和yum源。
- root帳號的登入密碼必須是8-30個字元，並且同時包含大寫或小寫字母、數字和特殊符號中的三類字元。特殊符號可以是( ) ` ~ ! @ # \$ % ^ & \* - \_ + = | { } [ ] : ; ' < > , . ? /。

### 不支援項

- 不支援ISO鏡像檔案。請線上下環境通過VirtualBox等工具製作後，轉換成阿里雲ECS支援的鏡像格式RAW、VHD或qcow2等再匯入。
- 不支援多個網路介面。
- 不支援IPv6地址。
- 不支援調整系統硬碟分區，目前只支援單個根分區。
- 不支援在鏡像中安裝qemu-ga，否則會導致ECS所需要的部分服務不可用。

### 支援項

- 支援RAW、qcow2和VHD格式鏡像。匯入其他格式的鏡像前，請先[轉換鏡像格式](#)後再匯入。
- 支援xfs、ext3和ext4檔案系統，支援MBR分區。



说明:

ext4檔案系統中不能包含特性 (feature) 64bit，並且特性 project 和 quota 不能成對出現。您可以運行 `tune2fs -l ext4##### | grep features` 命令查看ext4檔案系統中包含的特性列表。

- 支援匯入包含以下作業系統版本的鏡像：
  - Aliyun Linux
  - CentOS 5/6/7
  - CoreOS 681.2.0+
  - Debian 6/7
  - FreeBSD
  - OpenSUSE 13.1
  - RedHat
  - RHEL (Red Hat Enterprise Linux)
  - SUSE Linux 10/11/12
  - Ubuntu 10/12/13/14/16/18

#### 非標準平台Linux鏡像

為便於區分，如果您匯入的Linux系統鏡像不在ECS提供的公用鏡像列表裡，該作業系統平台鏡像為非標準平台鏡像。非標準平台鏡像雖然來自標準作業系統平台，但是系統關鍵性設定檔、系統基礎環境和應用方面沒有遵守標準平台要求的鏡像。如果您需要使用非標準平台鏡像，請在在匯入鏡像時選擇以下任何一種鏡像類型：

- Others Linux：ECS統一標識為其他系統類別型。如果匯入Others Linux平台鏡像，ECS不會對所建立的執行個體做任何處理。如果您在製作鏡像前開啟了DHCP，ECS會自動為您配置網路。完成執行個體建立後，請通過ECS管理主控台的[遠端連線](#)功能串連執行個體，再自行配置IP、路由和密碼等。
- Customized Linux：定製版鏡像。匯入Customized Linux鏡像後，請按照ECS標準系統配置方式配置執行個體的網路和密碼等。更多詳情，請參見[定製Linux自訂鏡像](#)。

### 4.3.3 安裝cloud-init

在使用Linux類型自訂鏡像時，為保證運行該鏡像的執行個體能成功完成初始化配置，建議您根據本文描述在原始伺服器上安裝cloud-init。

#### 什麼是cloud-init

cloud-init是雲平台為Linux作業系統的虛擬機器做系統初始化配置的開源服務軟體。阿里雲、AWS、Azure和OpenStack等主流雲平台均支援cloud-init。更多詳情，請參見[cloud-init官方文檔](#)。



阿里雲cloud-init能在執行個體啟動階段完成初始化配置，包括網路、NTP、軟體源、主機名稱和SSH金鑰組等，同時進行執行個體自訂資料（User data）指令碼的執行。

### 適用對象

阿里雲所有公用鏡像預設安裝cloud-init。如果您在以下情境中使用自訂鏡像時，為保證建立的執行個體能自動初始化系統配置，建議您為Linux伺服器安裝阿里雲版本cloud-init：

- 準備遷移上雲的，但未安裝cloud-init的Linux伺服器。
- 已安裝cloud-init，但版本低於0.7.9的Linux伺服器。
- 已在阿里雲啟動並執行，但未安裝cloud-init的ECS執行個體。

### 檢查版本

不同雲平台及不同版本之間的cloud-init可能互相不相容，請選擇合適版本並配置合適的資料來源（datasource）。其中阿里雲cloud-init的版本為0.7.6a，資料來源為 `Aliyun`。

安裝cloud-init後，預設開啟開機自啟動選項，若所選版本不相容或資料來源配置不當，下次重啟伺服器時可能導致cloud-init運行異常和系統啟動緩慢，甚至無法正常啟動系統。安裝前請做好資料備份，不準備遷移上雲的伺服器需謹慎安裝。

- 如何檢查是否已安裝cloud-init：`which cloud - init`



说明：

無輸出資訊表示未安裝，您需要安裝阿里雲版本cloud-init。

- 如何檢查cloud-init版本號碼命令：`cloud - init -- version`



说明：

如果版本低於社區0.7.9版本（0.7.6a版本除外），您需要安裝阿里雲版本cloud-init。

（推薦）安裝阿里雲版本cloud-init

1. 檢查並安裝python-pip依賴庫。
2. 下載阿里雲版本cloud-init並解壓到目前的目錄：

```
wget http://ecs-image-utils.oss-cn-hangzhou.aliyuncs.com/cloudinit/ali-cloud-init-latest.tgz
```

```
tar -zxvf ali-cloud-init-latest.tgz
```

3. 進入 `cloud-init` 的 `tools` 目錄，執行 `cloud-init` 安裝指令碼 `deploy.sh`：

```
bash ./deploy.sh <issue> <major_version>
```

• 參數說明如下：

- `issue`：作業系統平台類型，取值範圍：`centos` | `redhat` | `rhel` | `debian` | `ubuntu` | `opensuse` | `sles`。參數取值均大小寫敏感，其中 `sles` 表示 SUSE/SLES。
- `major_version`：作業系統平台主要版本號碼。例如，CentOS 6.5 的主要版本號碼為 6。

• 命令樣本如下：

- 在 CentOS 6.5 中安裝 `cloud-init`：

```
bash ./deploy.sh centos 6
```

- 在 Ubuntu 14.04 中安裝 `cloud-init`：

```
bash ./deploy.sh ubuntu 14
```

4. 確認是否成功安裝。若返回 `"description": "success"`，表示安裝成功。

## 安裝阿里雲版本 `cloud-init` 樣本

以下為不同發行平台的安裝命令樣本供您參考：

• CentOS 6/7

```
# 檢查安裝 python - pip
if ! python -c 'import setuptools' >& /dev/null;
then
  yum -y install python-pip
fi
# 備份舊版 cloud-init 配置
test -d /etc/cloud && mv /etc/cloud /etc/cloud-old
# 下載並解壓阿里雲版本 cloud-init
wget http://ecs-image-utils.oss-cn-hangzhou.aliyuncs.com/cloudinit/ali-cloud-init-latest.tgz
tar -zxvf ./ali-cloud-init-latest.tgz
# 安裝 cloud-init
issue_major=$(cat /etc/redhat-release | awk '{printf $3}' | awk -F '.' '{printf $1}')
```

```
bash ./ cloud - init -*/ tools / deploy . sh centos "$
issue_majo r "
```

- RHEL 6/7

```
# 檢查安裝 python - pip
if ! python - c ' import  setuptools ' >& / dev / null ;
then
    yum - y  install  python - pip
fi
# 備份舊版 cloud - init 配置
test - d / etc / cloud && mv / etc / cloud / etc / cloud -
old
# 下載並解壓阿里雲版本 cloud - init
wget http :// ecs - image - utils . oss - cn - hangzhou .
aliyuncs . com / cloudinit / ali - cloud - init - latest . tgz
tar - zxvf ./ ali - cloud - init - latest . tgz
# 安裝 cloud - init
issue_majo r =$( cat / etc / os - release | grep  VERSION_ID
| awk - F ' "' '{ printf $ 2 }' | awk - F '.' '{ printf $ 1
}')
bash ./ cloud - init -*/ tools / deploy . sh rhel "$
issue_majo r "
```

- Ubuntu 14/16/18

```
# 檢查安裝 python - pip
if ! python - c ' import  setuptools ' >& / dev / null ;
then
    apt - get  install  python - pip - y
fi
# 備份舊版 cloud - init 配置
test - d / etc / cloud && mv / etc / cloud / etc / cloud -
old
# 下載並解壓阿里雲版本 cloud - init
wget http :// ecs - image - utils . oss - cn - hangzhou .
aliyuncs . com / cloudinit / ali - cloud - init - latest . tgz
tar - zxvf ./ ali - cloud - init - latest . tgz
# 安裝 cloud - init
issue_majo r =$( cat / etc / os - release | grep  VERSION_ID
| awk - F ' "' '{ printf $ 2 }' | awk - F '.' '{ printf $ 1
}')
bash ./ cloud - init -*/ tools / deploy . sh ubuntu "$
issue_majo r "
```

- Debian 8/9

```
# 檢查安裝 python - pip
if ! python - c ' import  setuptools ' >& / dev / null ;
then
    apt - get  - y  install  python - pip
fi
# 備份舊版 cloud - init 配置
test - d / etc / cloud && mv / etc / cloud / etc / cloud -
old
# 下載並解壓阿里雲版本 cloud - init
wget http :// ecs - image - utils . oss - cn - hangzhou .
aliyuncs . com / cloudinit / ali - cloud - init - latest . tgz
tar - zxvf ./ ali - cloud - init - latest . tgz
# 安裝 cloud - init
```

```

issue_major=$(cat /etc/os-release | grep VERSION_ID
| awk -F '"' '{printf $2}' | awk -F '.' '{printf $1
}')
bash ./cloud-init-*/tools/deploy.sh debian "$
issue_major"

```

- SUSE 11/12

```

# 檢查安裝 python - pip
if ! python -c 'import setuptools' >& /dev/null; then
    zypper -n install python - pip
fi
# 備份舊版 cloud-init 配置
test -d /etc/cloud && mv /etc/cloud /etc/cloud -
old
# 下載並解壓阿里雲版本 cloud-init
wget http://ecs-image-utils.oss-cn-hangzhou.
aliyuncs.com/cloudinit/ali-cloud-init-latest.tgz
tar -zxvf ./ali-cloud-init-latest.tgz
# 安裝 cloud-init
issue_major=$(cat /etc/os-release | grep VERSION_ID
| awk -F '"' '{printf $2}' | awk -F '.' '{printf $1
}')
bash ./cloud-init-*/tools/deploy.sh sles "$
issue_major"

```

- OpenSUSE 13/42

```

# 檢查安裝 python - pip
if ! python -c 'import setuptools' >& /dev/null; then
    zypper -n install python - pip
fi
# 備份舊版 cloud-init 配置
test -d /etc/cloud && mv /etc/cloud /etc/cloud -
old
# 下載並解壓阿里雲版本 cloud-init
wget http://ecs-image-utils.oss-cn-hangzhou.
aliyuncs.com/cloudinit/ali-cloud-init-latest.tgz
tar -zxvf ./ali-cloud-init-latest.tgz
# 安裝 cloud-init
issue_major=$(cat /etc/os-release | grep VERSION_ID
| awk -F '"' '{printf $2}' | awk -F '.' '{printf $1
}')
bash ./cloud-init-*/tools/deploy.sh opensuse "$
issue_major"

```

### (可選) 安裝社區版cloud-init

#### 前提條件

安裝cloud-init之前，原始伺服器必須安裝git、python 2.7和python-pip依賴軟體，以下是安裝命令供您參考：

- CentOS/RedHat:

```
yum -y install git python python - pip
```

- Ubuntu/Debian:

```
apt - get - y install git python python - pip
```

- OpenSUSE/SUSE:

```
zypper - n install git python python - pip
```

## 操作步驟

1. 登入原始伺服器。
2. 使用git下載cloud-init源碼包，並進入cloud-init目錄：

```
git clone https://git.launchpad.net/cloud-init
cd ./cloud-init
```

3. 安裝所有依賴庫：

```
pip install -r requirements.txt
```

4. 安裝cloud-init:

```
python setup.py install
```

5. 運行 `vi /etc/cloud/cloud.cfg` 修改設定檔cloud.cfg。

將 `cloud_init _modules` :之前的配置修改為以下內容:

```
# Example datasource config
# The top level settings are used as module
# and system configuration.
# A set of users which may be applied and / or
# used by various modules
# when a 'default' entry is found it will reference
# the 'default_user'
# from the distro configuration specified below
users :
  - default
user :
  name : root
  lock_passwd : False
# If this is set, 'root' will not be able to
# ssh in and they
# will get a message to login instead as the
# above $ user
disable_root : false
# This will cause the set + update hostname module to
# not operate (if true)
preserve_hostname : false
syslog_fix_perms : root : root
datasource_list : [ AliYun ]
```

```
# Example datasource config
datasource :
  AliYun :
    support_xen : false
    timeout : 5 # ( defaults to 50 seconds )
    max_wait : 60 # ( defaults to 120 seconds )
#   metadata_urls : [ ' blah . com ' ]
# The modules that run in the ' init ' stage
cloud_init _modules :
```

## 下一步

- 對於準備遷移上雲的Linux伺服器：您可以[使用遷雲工具遷移](#)或[匯入自訂鏡像](#)。
- 對於已在阿里雲上啟動並執行Linux自訂鏡像ECS執行個體：您可以重啟系統驗證結果。如果系統自動設定了主機名稱、軟體源和NTP等配置，則表示已成功安裝cloud-init。

## 排查故障



### 說明：

- 不同鏡像缺少的庫可能不同，您都可以通過pip安裝，之後再次安裝cloud-init。
- 如果系統預設軟體包管理器（例如 yum ）和 pip 管理器分別安裝過不同版本的依賴庫，可能造成庫版本衝突，導致cloud-init運行異常。建議您根據報錯資訊按需下載依賴庫。

### 報錯：no setuptools module in python

報錯資訊提示 `no setuptools module in python`，表示您需要安裝python setuptools。通過以下方式解決：

- CentOS/RedHat: `yum -y install python - pip`
- Ubuntu/Debian: `apt - get - y install python - pip`
- OpenSUSE/SUSE: `zypper - n install python - pip`

### 報錯：No module named six

使用pip安裝six庫 `pip install six` 可以解決以下報錯。

```
File "/ root / cloud - init / cloudinit / log . py ", line 19 ,
in < module >
    import six
ImportError : No module named six )
```

### 報錯：No module named oauthlib.oauth1

使用pip安裝oauthlib庫 `pip install oauthlib` 可以解決以下報錯。

```
File "/ root / cloud - init / cloudinit / url_helper . py ", line
20 , in < module >
    import oauthlib . oauth1 as oauth1
```

```
ImportError: No module named oauthlib.oauth1 )
```

報錯時沒有明確缺少的依賴庫

如果安裝報錯時，沒有明確提示缺少哪些依賴庫時，您可以根據cloud-init的requirements.txt檔案裡顯示的庫，安裝所有依賴庫：

```
pip install -r requirements.txt
```

參考連結

[cloud-init官網 - 阿里雲資料來源說明文檔](#)

### 4.3.4 安裝virtio驅動

為避免部分伺服器、虛擬機器或者雲主機的作業系統在匯入自訂鏡像後，建立的ECS執行個體無法啟動，您需要在匯入鏡像前檢查是否需要在原始伺服器中安裝virtio驅動。

無需安裝virtio驅動的鏡像

從本地匯入自訂鏡像時，阿里雲會自動處理匯入的自訂鏡像的virtio驅動的作業系統有：

- Windows Server 2008
- Windows Server 2012
- Windows Server 2016
- CentOS 6/7
- Ubuntu 12/14/16
- Debian 7/8/9
- SUSE 11/12

以上列表的鏡像，預設已安裝virtio驅動的系統，需要注意修復臨時檔案系統。

需要安裝virtio驅動的鏡像

其他不在以上列表的作業系統，您需要在匯入鏡像之前，為原始伺服器安裝virtio驅動。

檢查伺服器核心是否支援virtio驅動

1. 運行 `grep -i virtio /boot/config-$(uname -r)` 檢查當前作業系統的核心是否支援virtio驅動。



说明:

- 如果在輸出資訊中沒有找到 `VIRTIO_BLK` 及 `VIRTIO_NET` 的資訊，表示該作業系統沒有安裝virtio相關驅動，暫時不能直接匯入阿里雲雲平台。您需要為您的伺服器編譯安裝virtio驅動。
- 如果參數 `CONFIG_VIRTIO_BLK` 及 `CONFIG_VIRTIO_NET` 取值為 `y`，表示包含了virtio驅動，您可以參閱 [匯入鏡像必讀](#) 直接 [匯入自訂鏡像](#) 到阿里雲。
- 如果參數 `CONFIG_VIRTIO_BLK` 及 `CONFIG_VIRTIO_NET` 取值為 `m`，需要進入第2步。

2. 執行命令 `lsinitrd / boot / initramfs -$(uname -r).img | grep virtio` 確認virtio驅動是否包含在臨時檔案系統initramfs或者initrd中。



说明:

- 截圖表明，initramfs已經包含了virtio\_blk驅動，以及其所依賴的virtio.ko、virtio\_pci.ko和virtio\_ring.ko，您可以參閱 [匯入鏡像必讀](#) 直接 [匯入自訂鏡像](#) 到阿里雲。
- 如果臨時檔案系統initramfs沒有包含virtio驅動，則需要修復臨時檔案系統。

### 修復臨時檔案系統

通過檢查，發現原始伺服器核心支援virtio驅動，但是臨時檔案系統initramfs或者initrd中沒有包含virtio驅動時，需要修復臨時檔案系統。以CentOS等為例。

- CentOS/RedHat 5

```
mkinitrd -f --allow-missing \
  --with=xen-vbd --preload=xen-vbd \
  --with=xen-platform-pci --preload=xen-
platform-pci \
  --with=virtio_blk --preload=virtio_blk \
  --with=virtio_pci --preload=virtio_pci \
  --with=virtio_con sole --preload=virtio_con
sole \
```

- CentOS/RedHat 6/7

```
mkinitrd -f --allow-missing \
  --with=xen-blkfront --preload=xen-blkfront \
  --with=virtio_blk --preload=virtio_blk \
  --with=virtio_pci --preload=virtio_pci \
  --with=virtio_con sole --preload=virtio_con
sole \
```



```
) / boot / initramfs -$(uname -r).img $(uname -r)
```

#### • Debian/Ubuntu

```
echo -e 'xen-blkfront \ nvirtio_blk \ nvirtio_pci \
nvirtio_console' >> \
/etc/initramfs-tools/modules
mkinitramfs -o / boot / initrd .img -$(uname -r)"
```

### 編譯安裝virtio驅動

此處以Redhat伺服器為例，為您示範如何編譯安裝virtio驅動。

#### 下載核心安裝包

1. 運行 `yum install -y ncurses-devel gcc make wget` 安裝編譯核心的必要組件。
2. 運行 `uname -r` 查詢當前系統使用的核心版本，如樣本中的4.4.24-2.a17.x86\_64。
3. 前往 [Linux核心列表頁面](#) 下載對應的核心版本源碼，如樣本中的4.4.24開頭的 `linux-4.4.24.tar.gz`的網址為 <https://www.kernel.org/pub/linux/kernel/v4.x/linux-4.4.24.tar.gz>。
4. 運行 `cd /usr/src` 切換目錄。
5. 運行 `wget https://www.kernel.org/pub/linux/kernel/v4.x/linux-4.4.24.tar.gz` 下載安裝包。
6. 運行 `tar -xzf linux-4.4.24.tar.gz` 解壓安裝包。
7. 運行 `ln -s linux-4.4.24 linux` 建立連結。
8. 運行 `cd /usr/src/linux` 切換目錄。

#### 編譯核心

1. 依次運行以下命令編譯核心。

```
make mrproper
symvers_path=$(find /usr/src -name "Module.symvers")
test -f $symvers_path && cp $symvers_path .
cp /boot/config-$(uname -r) ./config
make menuconfig
```

2. 出現以下介面時，開始開啟virtio相關配置：



说明：

選 \* 配置表示編譯到核心，選 m 配置表示編譯為模組。

- a. 使用空格勾選Virtualization項。

確認是否勾選了KVM（Kernel-based Virtual Machine）選項。

```
Processor type and features --->
  [*] Paravirtualized guest support --->
    --- Paravirtualized guest support
    ( 128 ) Maximum allowed size of a domain in
    gigabytes
  [*] KVM paravirtualized clock
  [*] KVM Guest support
```

```
Device Drivers --->
  [*] Block devices --->
  < M > Virtio block driver ( EXPERIMENTAL )
  -* Network device support --->
    < M > Virtio network driver ( EXPERIMENTAL )
```

- b. 按下Esc鍵退出核心配置介面並根據彈窗提示儲存.config檔案。
- c. [檢查](#) virtio相關配置是否已經正確配置。
- d. 若檢查後發現暫未設定virtio相關配置，運行以下命令手動編輯.config檔案。

```
make oldconfig
make prepare
make scripts
make
make install
```

- e. 運行以下命令查看virtio驅動的安裝情況。

```
find /lib/modules/"$(uname -r)"/ -name "virtio.*" |
grep -E "virtio.*"
grep -E "virtio.*" < /lib/modules/"$(uname -r)"/
modules.builtin
```



说明:

如果任一命令輸出virtio\_blk、virtio\_pci.virtio\_console等檔案清單，表明您已經正確安裝了virtio驅動。

下一步

檢查virtio驅動後，您可以：

- [#unique\\_74](#)。

- [匯入自訂鏡像](#)。

### 4.3.5 配置 Customized Linux 自訂鏡像

當您的鏡像作業系統不在阿里雲所支援的已有平台類型中時，您可以在 [ECS 控制台](#) 匯入鏡像頁面選擇 Customized Linux（定製版鏡像）的平台類型，如下圖所示。

對於定製版鏡像，阿里雲會將它當作未知的作業系統類型，無法識別作業系統各種資訊，只能利用約定好的方法把執行個體運行所需要的必要的配置資訊傳入到執行個體中，然後執行個體啟動時運行預定義的指令碼處理這些配置資訊。

#### 配置方法及限制條件

##### 限制條件

- 定製版鏡像的第一個分區必須可以被寫入。
- 定製版鏡像的第一個分區類型支援 FAT32/EXT2/EXT3/EXT4/UFS。
- 鏡像的虛擬檔案大小必須大於 5 GB。

##### 配置方法

1. 在鏡像的第一個分區的根目錄下建立目錄 `aliyun_cus tom_image`。
2. 使用該鏡像所建立的執行個體在啟動時，阿里雲會在 `aliyun_cus tom_image` 目錄的 `os.conf` 檔案中寫入執行個體相關配置資訊。如果不存在這個檔案，則會自動建立。
3. 鏡像中需要有一個預定義的啟動指令碼，該指令碼用來解析 `os.conf` 檔案的各項配置，並進行相關配置，詳見 [配置解析指令碼 - 注意事項](#) 和 [指令碼樣本](#)。

#### 安全要求

定製版鏡像的基本安全要求如下：

- 不得存在可被遠程利用的高危漏洞。
- 使用控制台的 [管理終端](#) 功能登入時如果存在初始預設密碼，則必須在首次登入時修改，修改之前不允許進入系統做任何操作。
- SSH 不允許有初始預設密碼，初始密碼必須隨機產生（通過阿里雲 ECS 控制系統下發）。

#### os.conf 設定檔樣本

傳統網路執行個體的 os.conf 設定檔樣本

```
hostname = iZ23r29djm jZ
password = cXdlcjEyMz QK
eth0_ip_ad dr = 10 . 171 . 254 . 123
eth0_mac_a ddr = 00 : 8c : fa : 5e : 14 : 23
eth0_netma sk = 255 . 255 . 255 . 0
eth0_gatew ay = 10 . 171 . 254 . 1
```

```
eth0_route = " 10 . 0 . 0 . 0 / 8 10 . 171 . 254 . 1 ; 172 . 16 . 0
. 0 / 12 10 . 171 . 254 . 1 "
eth1_ip_ad dr = 42 . 120 . 74 . 105
eth1_mac_a ddr = 00 : 8c : fa : 5e : 14 : 24
eth1_netma sk = 255 . 255 . 255 . 0
eth1_gatew ay = 42 . 120 . 74 . 1
eth1_route = " 0 . 0 . 0 . 0 / 0 42 . 120 . 74 . 1 "
dns_namese rver = " 7 . 7 . 7 . 7 8 . 8 . 8 . 8 "
```

其中各個參數的說明如下表所示。

參數名稱	參數說明
hostname	主機名稱參數。
password	密碼參數，Base 64 編碼的字串。
eth0_ip_addr	eth0 網卡 IP 位址。
eth0_mac_addr	eth0 網卡 MAC 位址。
eth0_netmask	eth0 網卡掩碼。
eth0_gateway	eth0 網卡預設閘道。
eth0_route	eth0 路由列表（內網路由列表），預設用半形分號分隔。
eth1_ip_addr	eth1 網卡 IP 位址。
eth1_mac_addr	eth1 網卡 MAC 位址。
eth1_netmask	eth1 網卡掩碼。
eth1_gateway	eth1 網卡預設閘道。
eth1_route	eth1 路由（預設外網路由）列表，預設用半形分號分隔。
dns_nameserver	DNS 地址清單，預設用空格分隔。

VPC 網路的執行個體設定檔樣本

```
hostname = iZ23r29djm jZ
password = cXdlcjEyMz QK
eth0_ip_ad dr = 10 . 171 . 254 . 123
eth0_mac_a ddr = 00 : 8c : fa : 5e : 14 : 23
eth0_netma sk = 255 . 255 . 255 . 0
eth0_gatew ay = 10 . 171 . 254 . 1
eth0_route = " 0 . 0 . 0 . 0 / 0 10 . 171 . 254 . 1 "
dns_namese rver = " 7 . 7 . 7 . 7 8 . 8 . 8 . 8 "
```

其中各個參數的說明如下表所示。

參數名稱	參數說明
hostname	主機名稱參數。

參數名稱	參數說明
password	密碼參數，Base 64 編碼的字串。
eth0_ip_addr	eth0 網卡 IP 位址。
eth0_mac_addr	eth0 網卡 MAC 位址。
eth0_netmask	eth0 網卡掩碼。
eth0_gateway	eth0 網卡預設閘道。
eth0_route	eth0 路由列表，預設用半形分號分隔。
dns_nameserver	DNS 地址清單，預設用空格分隔。

### 配置解析指令碼 - 注意事項

建立執行個體時阿里雲會將配置項的相關資訊寫入第一個分區的 `aliyun_cus tom_image` 目錄的 `os.conf` 檔案中。配置定製版鏡像的關鍵是要在鏡像中預定義指令碼，該指令碼將從 `os.conf` 中讀取相關配置資訊並執行這些配置。下面是設計該指令碼時的一些注意事項。

- 各配置項設值規則：如 [os.conf 設定檔樣本](#) 的配置項所述，VPC 與傳統網路執行個體的配置項數量和部分配置項的取值規則均有所不同。
- 開機啟動：該指令碼需要設定成開機自動運行。
- 設定檔讀取路徑：鏡像在建立 I/O 最佳化執行個體或非 I/O 最佳化執行個體時，為第一個分區所分配的裝置名稱預設不一樣；所以在指令碼中最好可以以 `uuid` 或 `label` 來識別第一個分區的裝置。使用者密碼為 Base64 編碼的字串，所以在設定密碼時需要進行相關處理。
- 對於 VPC 及傳統網路的判斷：需要在指令碼中判斷該執行個體是傳統網路還是 VPC 網路。目前最簡單的方法是判斷配置項中是否存在 `eth1_route` 或其他 `eth1` 相關的配置項。
- VPC 及傳統網路的配置項不同之處：
  - 對於 VPC 執行個體，會在 `os.conf` 的 `eth0_route` 參數中配置預設外網路由；
  - 對於傳統網路執行個體，會在 `os.conf` 的 `eth1_route` 參數中配置預設路由，內網路由配置在 `eth0_route` 中。

所以需要在指令碼中在判斷出當前執行個體的網路類型後再有針對性地進行解析和處理。

- 配置最佳化：`os.conf` 中的配置在執行個體的整個生命週期中執行一次即可，所以在指令碼執行成功後最好刪掉 `os.conf` 設定檔，同時指令碼如果沒有讀取到 `os.conf` 配置，則不執行檔案中的配置。
- 自訂鏡像的處理：根據 Customized Linux 執行個體所建立的自訂鏡像，在鏡像中也會包含這個開機啟動指令碼。使用該自訂鏡像建立執行個體時，阿里雲會在執行個體第一次啟動時寫入 `os.conf` 配置，指令碼在檢測到該配置時即可執行相關配置。

- 修改相關配置時的處理：當執行個體的配置資訊通過阿里雲的控制台或 API 發生變更時，阿里雲會把相關資訊寫入到 `os.conf` 中，指令碼將被再次執行從而下發這些更改。

### 指令碼樣本

以下指令碼是以 CentOS 鏡像為例的樣本指令碼，需要注意如下事項：

- 該指令碼僅供參考，您需要根據實際的作業系統類型進行調整從而得到具體的指令碼。
- 在使用指令碼前，請先在鏡像中調試指令碼，並保證調試通過。
- 該指令碼必須配置為開機自動執行，比如將這個指令碼放到 `/etc/init.d/` 目錄下。

```
#!/ bin / bash
### BEGIN INIT INFO
# Provides : os - conf
# Required - Start : $ local_fs $ network $ named $ remote_fs
# Required - Stop :
# Should - Stop :
# Default - Start : 2 3 4 5
# Default - Stop : 0 1 6
# Short - Descriptio n : The initial os - conf job , config
the system .
### END INIT INFO
first_part ition_dir ='/ boot /'
os_conf_di r =${ first_part ition_dir }/ aliyun_cus tom_image
os_conf_fi le =${ os_conf_di r }/ os . conf
load_os_co nf () {
if [[ - f $ os_conf_fi le ]]; then
. $ os_conf_fi le
return 0
else
return 1
fi
}
cleanup () {
# ensure $ os_conf_fi le is deleted , to avoid repeating
config system
rm $ os_conf_fi le >& / dev / null
# ensure $ os_conf_di r is existst
mkdir - p $ os_conf_di r
}
config_pas sword () {
if [[ - n $ password ]]; then
password =$( echo $ password | base64 - d )
if [[ $? == 0 && - n $ password ]]; then
echo " root :$ password " | chpasswd
fi
fi
}
config_hos tname () {
if [[ - n $ hostname ]]; then
sed - i " s / ^ HOSTNAME = . * / HOSTNAME = $ hostname / " / etc /
sysconfig / network
hostname $ hostname
fi
}
config_dns () {
if [[ - n $ dns_namese rver ]]; then
```

```

dns_conf =/ etc / resolv . conf
sed - i '/^ nameserver .*/ d ' $ dns_conf
for i in $ dns_name server ; do
echo " nameserver $ i " >> $ dns_conf
done
fi
}
is_classic _network () {
# vpc : eth0
# classic : eth0 eth1
grep - q ' eth1 ' $ os_conf_file
}
config_network () {
/ etc / init . d / network stop
config_interface eth0 ${ eth0_ip_address } ${ eth0_netmask } ${
eth0_mac_address }
config_route eth0 ${ eth0_route }
if is_classic_network ; then
config_interface eth1 ${ eth1_ip_address } ${ eth1_netmask } ${
eth1_mac_address }
config_route eth1 ${ eth1_route }
fi
/ etc / init . d / network start
}
config_interface () {
local interface = $ 1
local ip = $ 2
local netmask = $ 3
local mac = $ 4
interface_cfg = "/ etc / sysconfig / network - scripts / ifcfg - ${
interface }"
cat << EOF > $ interface_cfg
DEVICE = $ interface
IPADDR = $ ip
NETMASK = $ netmask
HWADDR = $ mac
ONBOOT = yes
BOOTPROTO = static
EOF
}
config_default_gateway () {
local gateway = $ 1
sed - i " s / ^ GATEWAY = .*/ GATEWAY = $ gateway / " / etc / sysconfig
/ network
}
config_route () {
local interface = $ 1
local route = $ 2
route_conf = / etc / sysconfig / network - scripts / route - ${
interface }
> $ route_conf
echo $ route | sed ' s / ; / \ n / ' | \
while read line ; do
dst = $( echo $ line | awk '{ print $ 1 }')
gw = $( echo $ line | awk '{ print $ 2 }')
if ! grep - q "$ dst " $ route_conf 2 > / dev / null ; then
echo "$ dst via $ gw dev $ interface " >> $ route_conf
fi
if [ [ "$ dst " == " 0 . 0 . 0 . 0 / 0 " ] ] ; then
config_default_gateway $ gw
fi
done
}
##### sysvinit service portal #####

```

```
start () {
  if load_os_conf ; then
    config_password
    config_network
    config_hostname
    config_dns
    cleanup
    return 0
  else
    echo "not load $ os_conf_file "
    return 0
  fi
}
RETVAL = 0
case "$1" in
  start )
    start
    RETVAL = $?
;;
*)
  echo " Usage : $0 { start }"
  RETVAL = 3
;;
esac
exit $ RETVAL
```

### 4.3.6 轉換鏡像格式

ECS只支援匯入RAW、VHD和qcow2格式的鏡像檔案。其他鏡像檔案，需要轉換格式後再匯入。本文介紹如何利用qemu-img工具將其它格式的鏡像檔案轉換成VHD或RAW格式。qemu-img能將RAW、qcow2、VMDK、VDI、VHD (vpc)、VHDX、qcow1或QED格式的鏡像轉換成VHD格式，也可以實現RAW和VHD格式的互相轉換。

本地為Windows作業系統

按以下步驟安裝qemu-img並轉換鏡像檔案格式：

1. [下載qemu-img](#)並完成安裝。本樣本安裝路徑為 `C:\Program Files\qemu`。



## 2. 為qemu-img配置環境變數：

- a. 選擇 開始 > 電腦，按右鍵 屬性。
- b. 在左側導覽列裡，單擊 進階系統設定。
- c. 在系統屬性對話方塊裡，單擊 進階 頁籤，並單擊 環境變數。
- d. 在環境變數對話方塊裡，在系統變數部分找到 Path，並單擊 編輯。如果 Path 變數不存在，單擊 建立。
- e. 添加系統變數值：
  - 如果是 編輯系統變數，在變數值裡，添加 C :\ Program Files \ qemu ，不同的變數值之間以半形分號 (;) 分隔。
  - 如果是 建立系統變數，在變數名處輸入 Path ，在變數值處輸入 C :\ Program Files \ qemu 。

3. 測試是否成功配置環境變數：開啟Windows 命令提示字元，運行 `qemu - img -- help`，如果顯示正常即表示已配置環境變數。

4. 在命令提示字元裡，使用 `cd [源鏡像檔案所在的目錄]` 切換檔案目錄，例如 `cd D :\ ConvertImage`。

5. 運行 `qemu - img convert - f qcow2 - O raw centos . qcow2 centos . raw` 命令轉換鏡像檔案格式，其中：

- - f 的參數值為源鏡像檔案的格式。
- - O (必須是大寫) 的參數值為目標鏡像格式、源鏡像檔案名稱和目標檔案名稱。

轉換完成後，目標檔案會出現在源鏡像檔案所在的目錄下。

本地為Linux作業系統

按以下步驟安裝qemu-img並轉換鏡像檔案格式：

### 1. 安裝qemu-img，樣本如下：

- 如果本地為Ubuntu作業系統，運行 `apt install qemu - img`。
- 如果本地為CentOS作業系統，運行 `yum install qemu - img`。

2. 運行 `qemu - img convert - f qcow2 - O raw centos . qcow2 centos . raw` 命令轉換鏡像檔案格式，其中：

- - f 的參數值為源鏡像檔案的格式。
- - O (必須是大寫) 的參數值為目標鏡像格式、源鏡像檔案名稱和目標檔案名稱。

轉換完成後，目標檔案會出現在源鏡像檔案所在的目錄下。

## 常見問題

如果安裝qemu-img報錯時，而且沒有明確提示缺少哪些依賴庫時，您可以根據cloud-init的requirements.txt檔案裡顯示的庫，並運行 `pip install -r requirements.txt` 安裝所有依賴庫。

## 下一步

[匯入自訂鏡像](#)

### 4.3.7 匯入自訂鏡像

將本地物理鏡像檔案匯入ECS適用於在雲上部署業務的情境。匯入的自訂鏡像會出現在相應地區下您的自訂鏡像列表裡，您可以使用這份鏡像建立ECS執行個體和更換系統硬碟。



#### 說明：

- 匯入自訂鏡像是相對耗時的任務，完成的時間取決於鏡像檔案的大小和當前置入任務並發數，需要您耐心等待。
- 匯入自訂鏡像時，系統會自動產生一份快照，您可以在快照列表裡看到快照資訊。匯入鏡像任務未完成前，快照的狀態會顯示為失敗。任務完成後，狀態會自動更新為可用。這份快照的容量為匯入的自訂鏡像檔案大小，與匯入自訂鏡像時設定的系統硬碟大小無關。

## 前提條件

在匯入鏡像前，您應該已經完成如下工作：

- 參閱[匯入鏡像必讀](#)、[定製Linux自訂鏡像](#)和[轉換鏡像格式](#)瞭解鏡像限制和要求。
- [開通OSS服務](#)。
- （可選）如果您使用的是RAM子帳號，應預先聯絡主帳號授權 `AliyunECSI` `mageImport` `DefaultRole` 角色策略獲得操作許可。

## 操作步驟

在阿里雲管理主控台上匯入自訂鏡像的步驟如下：

1. 使用OSS的第三方工具用戶端、API或者OSS SDK上傳製作好的自訂鏡像。關於如何上傳大於5 GiB的檔案，請參閱[OSS斷點續傳](#)。
2. 登入 [ECS管理主控台](#)。
3. 在左側導覽列裡，選擇快照和鏡像 > 鏡像。
4. 在鏡像列表頁，單擊匯入鏡像。
5. 在匯入鏡像對話方塊裡，單擊提示資訊第3步裡的確認地址。

6. 在雲資源訪問授權視窗，勾選 `AliyunECSI mageImport DefaultRole` 和 `AliyunECSE xportDefaultRole`，單擊同意授權授權ECS服務訪問您的OSS資源。
7. 在鏡像列表頁，再次單擊匯入鏡像。
8. 在匯入鏡像對話方塊裡，完成以下設定：
  - 鏡像所在地區：上傳鏡像檔案的OSS Bucket所在的地區。
  - OSS Object 地址：從OSS控制台擷取的鏡像檔案Object地址。更多詳情，請參閱OSS[獲取檔案訪問地址](#)。
  - 鏡像名稱：指定自訂鏡像鏡像檔案匯入後顯示的名稱。長度為2–128個字元，以大小寫字母或中文開頭，可包含數字、點號 (.)、底線 (\_)、半形冒號 (:) 和連字號 (-)。
  - 作業系統：根據您的鏡像操作形同類型選擇Windows或Linux。如果您要匯入的是非標準平台自訂鏡像鏡像，應選擇Linux。
  - 系統硬碟大小：系統硬碟大小範圍為40 GiB–500 GiB。
  - 系統架構：64位作業系統選擇x86\_64，32位作業系統選擇i386。
  - 系統平台：下拉式清單裡提供的選項由作業系統的選項決定，選擇適用於您的鏡像的系統平台。
    - Windows：支援Windows Server 2003、Windows Server 2008和Windows Server 2012。
    - Linux：支援CentOS、SUSE、Ubuntu、Debian、FreeBSD、CoreOS、Aliyun、Customized Linux和Others Linux（請[提交工單](#) 確認是否支援）。
    - 如果您的鏡像的作業系統是根據Linux核心定製開發的，請[提交工單](#) 聯絡我們。
  - 鏡像格式：僅支援RAW，qcow2和VHD格式，推薦使用傳輸內容更小的qcow2或者VHD格式。
  - 鏡像描述：填寫鏡像描述資訊，便於後續管理。
  - 添加資料盤鏡像：同時匯入包含資料盤的自訂鏡像鏡像，支援的資料盤容量範圍為5 GiB–2000 GiB。
9. 資訊確認無誤後，單擊確定，建立一個匯入自訂鏡像鏡像的任務。
10. (可選) 您可以在您匯入地區的鏡像列表中查看匯入進度。在任務完成前，您都可以通過[任務管理](#)，找到該匯入的自訂鏡像鏡像，取消匯入鏡像任務。

您也可以使用ECS API `ImportImage`匯入自訂鏡像。

下一步

[#unique\\_41](#)

## 參考連結

- [自訂鏡像FAQ](#)
- [使用Packer建立並匯入本地鏡像](#)

## 4.4 複製鏡像

複製鏡像適用於跨地區部署應用，您可以在不同地區的 ECS 執行個體中運行同一鏡像環境。複製鏡像將某一地區中的一份自訂鏡像複製到另一個地區，任務完成時間取決於網路傳輸速度和並發任務的排隊數量。

### 注意事項

- 複製鏡像時，會同時在目標地區產生對應的快照，隨後在目標地區根據快照產生自訂鏡像。因此，會產生不同地區間的資料轉送流量費。目前未對這部分流量收費，具體收費時間以阿里雲公告為準。
- 目前支援在華北 1、華北 2、華北 3、華北 5、華東 1、華東 2 和華南 1 地區之間複製鏡像。涉及其他國家和地區地區時，可以 [提交工單](#) 申請，並在工單頁面註明複製鏡像時的源地區、目標地區、系統硬碟已使用空間、是否有資料盤、系統硬碟已使用空間和鏡像 ID。
- 複製鏡像後，您會在另一地區獲得同樣配置的一份自訂鏡像。但是相關角色授權和服務授權資訊會丟失，也不會包含原來設定過的 `#unique_84`。

### 操作步驟

在管理主控台上複製鏡像的步驟如下：

1. 登入 [ECS 管理主控台](#)。
2. 選擇地區。
3. 在左側導覽列中，選擇 [快照和鏡像](#) > [鏡像](#)。
4. 選中需要複製的鏡像，鏡像類型 必須是 自訂鏡像，在 操作 列中，單擊 複製鏡像。



#### 說明：

如果您要複製的自訂鏡像大於 200 GiB，需要提交工單處理。當您單擊 複製鏡像 時，系統會引導您提交工單。

5. 在 複製鏡像 對話方塊中，您可以看到選中的自訂鏡像的 ID，您需要完成以下設定：
  - a. 選擇 目標地區。
  - b. 填寫鏡像在目標地區顯示的 自訂鏡像名稱 和 自訂鏡像描述。
  - c. 單擊 確定。

6. 切換到目標地區，查看自訂鏡像的複製進度。當進度顯示為 100% 時，說明任務完成。



说明:

如果進度沒到 100%，狀態為建立中時，您可以隨時點擊取消複製。取消任務後，目標地區的鏡像列表中不再顯示該自訂鏡像。

您也可以使用 ECS API [CopyImage](#) 和 [CancelCopyImage](#) 完成複製鏡像任務。

下一步

當鏡像的狀態變為可用時，您可以使用複製的鏡像 [创建ECS实例](#) 或 [#unique\\_38](#)。

您也可以的在目標地區中查看從源地區複製過來的快照資訊。

FAQ

[複製鏡像 FAQ](#)

## 4.5 共用鏡像

製作了自訂鏡像後，您可以將鏡像共用給其他阿里雲使用者使用。共用鏡像能明顯縮短 ECS 新使用者的適應時間，使得新使用者能基於您的自訂鏡像，快速建立 ECS 執行個體並建立業務環境。並且共用鏡像不會佔用被共用者的鏡像使用額度。

注意事項

您只能共用自己的自訂鏡像，無法共用其他使用者共用給您的鏡像。每份自訂鏡像最多可以共用給 50 個使用者，並且只能在同一阿里雲地區中共用鏡像，不支援跨地區共用鏡像。

共用鏡像之前，請確認自訂鏡像已清除敏感性資料和檔案。



说明:

ECS 無法保證共用鏡像的完整性和安全性，使用共用鏡像時請確認鏡像來自於可信任的共用者或共用帳號，並需要自行承擔風險。根據共用鏡像建立執行個體後，請務必 [串連執行個體](#) 檢查鏡像的完整性和安全性。

刪除共用鏡像造成的影響

如果您的自訂鏡像已經共用給其他帳號，您需要預先解除該自訂鏡像的全部共用關係後才能刪除鏡像。刪除被共用的自訂鏡像後：

- 被共用者無法通過管理主控台或者 ECS API 查詢到該鏡像，也無法使用該鏡像建立 ECS 執行個體和更換系統硬碟。

- 被共用者使用共用鏡像建立的 ECS 執行個體無法重新初始化系統硬碟。

## 共用鏡像

在管理主控台上共用鏡像的步驟如下：

1. 登入 [ECS 管理主控台](#)。
2. 選擇地區。
3. 在左側導覽列中，選擇 [快照和鏡像 > 鏡像](#)。
4. 選擇一份自訂鏡像，在操作列中，單擊 [共用鏡像](#)。
5. 在共用鏡像頁面，帳號類型選擇 [aliyunUid](#)，在帳號對話方塊中輸入待共用的使用者識別碼。您可以參閱 [附錄. 如何擷取帳號 ID 擷取 aliyunUid](#)。

共享鏡像

請勿在鏡像中包含不適合公開的數據，目前您已將本鏡像共享給了 1 個帳號。

帳號類型：[aliyunUid](#) \*帳號：[15519\[redacted\]](#) [共享鏡像](#)

<input type="checkbox"/>	aliyunUid	操作
<input type="checkbox"/>	15519[redacted]	<a href="#">取消共享</a>

[取消共享](#) 共有1條, 每頁顯示：[\[dropdown\]](#) 條 [«](#) [<](#) [1](#) [>](#) [»](#)



### 說明：

您也可以點擊 [不共用](#) 解除與該使用者的鏡像共用關係。不共用後，該使用者將無法查詢和使用該鏡像；如果該使用者已經使用這份共用鏡像建立了執行個體，執行個體將無法[重新初始化系統硬碟](#)。

6. (可選) 被共用者可以在ECS管理主控台同一地區中的 [快照和鏡像 > 鏡像 > 共用鏡像](#) 頁面查看共用鏡像。

您也可以使用 ECS API [ModifyImageSharePermission](#) 和 [DescribeImageSharePermission](#) 完成共用鏡像。

## 下一步

共用鏡像後，被共用者可以使用該鏡像建立一台或多台實執行個體。

1. 登入 [ECS 管理主控台](#)。
2. 參閱 [#####創建ECS实例](#) 建立一台或多台執行個體，並在選擇鏡像時選擇 共用鏡像。

被共用者還可以使用該鏡像為執行個體 [#unique\\_38](#)。

#### 附錄. 如何擷取帳號 ID

您可以通過如下步驟查詢您的賬戶ID。

1. 登入阿里雲管理主控台。
2. 滑鼠移至右上方帳戶圖片，如 `funCustomer*****@aliyun.com`，在彈出的帳號菜單中單擊安全設定。
3. 找到帳號ID。

## 4.6 匯出鏡像

匯出ECS自訂鏡像適用於本地測試或私人雲端環境情境。



说明:

- 匯出自訂鏡像是相對耗時的任務，需要您耐心等待。完成的時間取決於自訂鏡像檔案的大小和當前置出任務並發數。
- 匯出自訂鏡像需要使用 [Object Storage Service](#)，所以會產生一定的OSS儲存和下載的流量費用。更多詳情，請參閱 [OSS计量项和计费项](#)。

#### 限制條件

您需要注意如下限制條件：

- 不支援匯出通過 [鏡像市场](#) 的系統硬碟快照建立的自訂鏡像。
- 匯出的自訂鏡像中可以包含資料盤快照，但資料盤不能超過4塊，單塊資料盤容量最大不能超過 500 GiB。
- 使用匯出的全鏡像 [#unique\\_95](#) 時，您需要自行確認/etc/fstab記錄的檔案裝置與匯出的資料盤快照資訊互相對應。

#### 前提條件

在匯出自訂鏡像前，您應該已經完成如下工作：

- [提交工單](#) 申請匯出許可權，並說明匯出自訂鏡像的使用情境。

- 請確保您已經開通OSS服務，而且自訂鏡像所在地區裡有可用的OSS Bucket。更多詳情，請參閱 [OSS创建存储空间](#)。

## 操作步驟

在管理主控台上匯出一份自訂鏡像的步驟如下：

1. 登入 [ECS管理主控台](#)。
2. 在左側導覽列裡，選擇 [快照和鏡像](#) > [鏡像](#)。
3. 選擇地區。
4. 找到需要匯出的自訂鏡像，在 [操作](#) 列中，單擊 [匯出鏡像](#)。
  - a. 在 [匯出鏡像](#) 對話方塊裡，單擊提示資訊第3步裡的 [確認地址](#)。
  - b. 在 [雲資源訪問授權](#) 視窗，單擊 [同意授權](#) [授權ECS服務訪問您的OSS資源](#)。
5. 回到ECS管理主控台首頁，在自訂鏡像列表頁 [操作](#) 列中，再次單擊 [匯出鏡像](#)。
6. 在 [匯出鏡像](#) 對話方塊中：
  - 選擇一個自訂鏡像所在地區的OSS Bucket。
  - 為自訂鏡像的Object名稱設定一個前綴。例如，您可以將Demo設為前綴，則匯出的自訂鏡像檔案，在OSS Bucket中的名稱即為Demo-[系統自動產生的檔案名稱]。
7. 單擊 [確定](#) 開始匯出自訂鏡像。
8. （可選）取消匯出自訂鏡像任務。在任務完成前，您都可以通過 [任務管理](#)，在相應地區下找到匯出自訂鏡像的任務，取消匯出自訂鏡像。

您也可以使用ECS API [ExportImage](#) 和 [CancelTask](#) 匯出自訂鏡像。

## 下一步

1. 登入 [OSS管理主控台](#) 查詢匯出自訂鏡像的結果。



### 说明:

匯出的自訂鏡像包含資料盤快照時，您的OSS中會出現多個檔案。檔案名稱帶有 `system` 的表示系統硬碟快照，檔案名稱帶有 `data` 的表示資料盤快照。資料盤快照會有與資料盤對應的標識，即磁碟的掛載點，如xvdb或者xvdc。

2. 自訂鏡像匯出成功後，[下載文件](#) 後下載自訂鏡像檔案，鏡像檔案的格式預設為RAW。



## 5 雲市場鏡像

鏡像，相當於Elastic Compute Service的裝機盤。您可以選擇自行配置環境和安裝軟體，也可以購買雲市場鏡像快速獲得預裝的雲端服務器運行環境或軟體應用，滿足建站、應用開發、可視化管理等個人化需求，讓ECS即開即用，省時方便。

### 雲市場鏡像的優點

不管是建立執行個體時，還是建立執行個體後，您都可以切換使用雲市場鏡像。雲市場鏡像部署相比自訂部署，有如下優點：

### 建立執行個體時選擇雲市場鏡像

如果您是新購雲端服務器的使用者，推薦您使用雲市場鏡像。

1. 前往[ECS購買頁](#)。
2. 參閱 [建立ECS執行個體](#) 完成選型配置，在 鏡像 配置項處，選擇 鏡像市場 > 從鏡像市場選擇（含作業系統）。

### 在雲市場選購鏡像後建立執行個體

1. 前往 [雲市場](#)。
2. 選擇您需要的鏡像環境並單擊 [立即購買](#)。
3. 如果您未登入阿里雲控制台，需要登入後才能成功跳轉。
4. 參閱 [建立ECS執行個體](#) 完成選型配置。

### 使用雲市場鏡像更換作業系統

若您已購買了雲端服務器，需要使用鏡像來部署運行環境，或者安裝軟體應用，操作如下：



说明：

更換鏡像會導致系統硬碟資料丟失，建議您在更換作業系統前備份資料。更多詳情，請參閱 [#unique\\_104](#)。

1. 登入 [ECS管理主控台](#) 。
2. 停止目標執行個體。

3. 參閱 [#unique\\_105](#)，並在更換系統硬碟 頁面的 鏡像類型 處選擇 鏡像市場，即可使用所需的鏡像。

## 6 鏡像相關開源工具

---

## 7 更換作業系統

---

在控制台中，您可以更換作業系統，比如把Linux系統更換成Windows系統，或把Ubuntu更換為CentOS。

更換作業系統，是通過更換執行個體的系統硬碟實現的。

- 如果要使用您的自訂鏡像，請參見 [#unique\\_105](#)。
- 如果要更換為公用鏡像，請參考 [#unique\\_108](#)。



说明:

非中國大陸的地區暫不支援Linux和Windows系統的互換，僅支援Linux和Linux、Windows和Windows同類型系統的更換。

## 8 常見問題

---

### 8.1 Windows Server半年渠道鏡像與執行個體管理

ECS支援Windows Server半年渠道鏡像，本文介紹如何管理由該鏡像建立的Windows Server半年渠道執行個體。

#### 鏡像簡介

ECS現已支援Windows Server半年渠道鏡像，建立執行個體時您可以在Windows Server公用鏡像列表中看到Version 1809資料中心版鏡像。Windows Server半年渠道鏡像是一款純Server Core模式啟動並執行作業系統，不提供圖形化使用者介面。Windows Server半年渠道鏡像對硬體要求寬鬆許多，降低了更新頻率並且支援遠端管理。本文適用於下列鏡像：

- Windows Server Version 1809 資料中心版
- Windows Server Version 1709 資料中心版

#### 執行個體管理工具簡介

Windows Server半年渠道執行個體不再包含資源管理員、控制台、Windows Explorer、不支援\*.msc 功能如devmgmt.msc 等。Windows Server半年渠道執行個體支援使用Sconfig、Server Manager、PowerShell和Windows Admin Center等工具管理伺服器。

由於Windows Server半年渠道執行個體使用Server Core模式，本文推薦您使用功能更為完善的PowerShell和Windows Admin Center。更多詳情，請參見微軟文檔[如何管理 Server Core 模式伺服器](#)。

#### PowerShell遠端管理

PowerShell依賴於.NET Framework實現了強大的物件導向的指令碼，可以做到SSH功能一樣遠端管理Windows執行個體。我們假設您的執行個體公網IP為172.16.1XX.183，您可以按以下步驟實現PowerShell遠端管理。

1. 遠端連線Windows執行個體。詳情請參見[#unique\\_111](#)。
2. 在命令列裡輸入 PowerShell 開啟PowerShell。
3. 在執行個體PowerShell中運行以下命令：

```
Enable - PSRemoting - Force
```

```
Set - NetFirewal lRule - Name " WINRM - HTTP - In - TCP - PUBLIC
" - RemoteAddr ess Any
```

4. 在執行個體所在安全性群組中添加規則允許存取HTTP 5985連接埠和HTTPS 5986連接埠。詳情請參見[#unique\\_112](#)。
5. 在用戶端電腦命令列裡輸入 `PowerShell` 開啟PowerShell。
6. 在用戶端PowerShell中運行以下命令：

```
Set - Item WSMAN : localhost \ client \ trustedhos ts - value
172 . 16 . 1XX . 183 - Force
```



说明：

172 . 16 . 1XX . 183 代表只授信您的執行個體，您也可以使用\*表示授信所有電腦。

7. 在用戶端PowerShell中運行 `Enter - PSSession ' 172 . 16 . 1XX . 183 ' - Credential : ' administra tor '` 並按提示輸入執行個體密碼。

現在您可以在用戶端電腦管理您的Windows執行個體了。

## Windows Admin Center

Windows Admin Center是一個基於瀏覽器的圖形管理工具，可以在Server Core運行環境中取代伺服器管理和MMC。我們假設您的執行個體公網IP為172.16.1XX.183，您可以按以下任一方法安裝Windows Admin Center。

### · 通過命令列安裝Windows Admin Center

1. 遠端連線Windows執行個體。詳情請參見[#unique\\_111](#)。
2. 在執行個體所在安全性群組中添加規則允許存取HTTP 5985連接埠和HTTPS 5986連接埠。詳情請參見[#unique\\_112](#)。
3. 在命令列裡輸入 `PowerShell` 開啟PowerShell。
4. 在執行個體PowerShell中運行以下命令：

```
Enable - PSRemoting - Force
Set - NetFirewal lRule - Name " WINRM - HTTP - In - TCP -
PUBLIC " - RemoteAddr ess Any
```

5. 運行以下命令下載Windows Admin Center。

```
wget - Uri http :// download . microsoft . com / download
/ E / 8 / A / E8A26016 - 25A4 - 49EE - 8200 - E4BCBF292C 4A /
HonoluluTe chnicalPre view1802 . msi - UseBasicPa rsing -
OutFile c :\ HonoluluTe chnicalPre view1802 . msi
```

```
msiexec /i c:\HonoluluTechnicalPreview1802.msi /qn
/L *v log.txt SME_PORT = 443 SSL_CERTIFICATE_OPTION
= generate
```

6. 運行 `cat log.txt` 命令查看下載進度，當記錄檔出現下列資訊，說明Windows Admin Center已經成功安裝。

```
MSI (s) (14:44) [09:48:37:885]: Product:
Project 'Honolulu '(技術預覽版) -- Installation completed
successfully.
MSI (s) (14:44) [09:48:37:885]: Windows
Installer 已安裝產品。產品名稱: Project 'Honolulu '(技術預覽版)。
產品版本: 1.1.10326.0。產品語言: 1033。製造商: Microsoft
Corporation。安裝成功或錯誤狀態: 0。
```

- 通過瀏覽器安裝Windows Admin Center

#### 前提條件

通過瀏覽器安裝Windows Admin Center需要在用戶端計算中完成，請確保您已經通過配置PowerShell管理執行個體。更多詳情，請參見[PowerShell 遠端管理](#)。

#### 操作步驟

1. 下載並安裝Windows Admin Center。
2. 完成安裝後，開啟<https://localhost/>。
3. 單擊添加，在彈窗中添加執行個體的IP地址。

現在，您可以通過Microsoft Edge或者Chrome使用Windows Admin Center的用戶端電腦管理執行個體。

## 常見問題

### 如何複製檔案到Windows Server半年渠道執行個體？

假設需要複製的檔案在您的用戶端電腦上，並且您已經配置了PowerShell遠端管理或者已安裝Windows Admin Center。

- 通過RDP應用

1. 遠端連線Windows執行個體。詳情請參見[#unique\\_111](#)。
2. 在用戶端電腦上，複製目標檔案。
3. 在執行個體CMD環境中輸入 `notepad`。
4. 單擊檔案 > 開啟，在開啟對話方塊裡，選擇檔案要複製的目標目錄，按右鍵選擇粘貼。

- 通過PowerShell遠程

1. 啟動目標Windows執行個體。
2. 在用戶端電腦上開啟CMD，輸入 `PowerShell` 進入PowerShell。
3. 通過PowerShell遠端管理目標執行個體。詳情請參見[PowerShell遠端管理](#)。
4. 在用戶端電腦上運行以下命令：

```
$ session = New - PSSession - ComputerName 172.16.1XX.183  
Copy - Item - ToSession $ session - Path C:\1.txt -  
Destination c:\2.txt
```



说明:

`C:\1.txt` 是用戶端電腦的檔案位置，`C:\2.txt` 是要拷貝到的Windows執行個體目錄。

- 通過Windows Admin Center

1. 啟動目標Windows 執行個體。
2. 配置Windows Admin Center工具。詳情請參見[Windows Admin Center](#)。
3. 開啟Windows Admin Center，點擊被管理的執行個體，單擊檔案，選中檔案後單擊上傳。

如何從內部關閉或者重啟Windows Server半年渠道執行個體？

- 通過RDP應用

1. 遠端連線Windows執行個體。詳情請參見[#unique\\_111](#)。
2. 在CMD中輸入 `sconfig`，根據需要選擇 `13` 重啟執行個體或者 `14` 停止執行個體並斷行符號。

- 通過PowerShell

1. 遠端連線Windows執行個體。詳情請參見[#unique\\_111](#)。
2. 在CMD中輸入 `PowerShell` 進入PowerShell。
3. 選擇並輸入以下命令列重啟或者停止執行個體。

```
shutdown -r -t 00 ::命令列 在 0 秒後重啟  
shutdown -s -t 00 ::命令列 在 0 秒後關機  
Stop - Computer - Force # Powershell 立即關機
```



```
Restart - Computer - Force # Powershell 立即重啟
```

- 通過PowerShell遠端管理

1. 啟動目標 Windows 執行個體。
2. 在用戶端電腦上開啟CMD，輸入 `PowerShell` 進入PowerShell。
3. 通過PowerShell遠端管理目標執行個體。詳情請參見[PowerShell遠端管理](#)。
4. 在用戶端電腦上選擇性運行以下PowerShell命令：

```
Enter - PsSession - ComputerName 172 . 16 . 1XX . 183  
Restart - Computer - Force #重啟  
Stop - Computer - Force #關機
```

- 通過Windows Admin Center

1. 啟動目標Windows執行個體。
2. 配置Windows Admin Center工具。詳情請參見[Windows Admin Center](#)。
3. 開啟Windows Admin Center，點擊被管理的執行個體，單擊概述，選擇性單擊重啟或者關機。

#### 如何安裝 IIS 服務？

- 通過RDP應用

1. 遠端連線Windows執行個體。詳情請參見[#unique\\_111](#)。
2. 在CMD中輸入 `PowerShell` 進入PowerShell。
3. 運行以下命令安裝IIS：

```
Import - Module ServerManager  
Add - WindowsFeature Web - Server , Web - CGI , Web - Mgmt  
- Console
```

- 通過PowerShell遠端管理

1. 啟動目標 Windows 執行個體。
2. 在用戶端電腦上開啟CMD，輸入 `PowerShell` 進入 PowerShell。
3. 通過PowerShell遠端管理目標執行個體，詳情請參見[PowerShell遠端管理](#)。
4. 在用戶端電腦上運行以下PowerShell命令：

```
Enter - PsSession - ComputerName 172 . 16 . 1XX . 183  
Import - Module ServerManager
```

```
Add - WindowsFeature Web - Server , Web - CGI , Web - Mgmt - Console
```

- 通過Windows Admin Center

1. 啟動目標Windows執行個體。
2. 配置Windows Admin Center工具，詳情請參見[Windows Admin Center](#)。
3. 開啟Windows Admin Center，選中被管理的執行個體，單擊角色和功能，單擊Web 服務器，選擇您需要的功能後單擊是。

如何重建立不小心在 RDP 會話中關閉了的命令列視窗？

如果在遠端工作階段中不小心關閉了命令列視窗，遠程應用將變成純黑介面，無法操作。這時您可以：

1. 通過mstsc串連的情況下按Ctrl + Alt + End按鍵組合，其他情況按Ctrl + Alt + Del按鍵組合。
2. 在出現的介面選擇工作管理員並斷行符號。
3. 在工作管理員中，依次單擊檔案 > 建立任務檔案，輸入 cmd 後單擊確定。

#### 參考連結

- 微軟Windows Server半年渠道概述：[Windows Server 半年渠道概述](#)。
- 微軟引入Windows Server Version 1709：[Introducing Windows Server, version 1709](#)
- [Windows Admin Center](#)
- 微軟遠端連線排錯：[About Remote Troubleshooting](#)

## 8.2 Linux鏡像如何開啟或關閉Meltdown與Spectre安全性漏洞補丁

本文描述了阿里雲ECS如何應對和預防Meltdown與Spectre損害，您可以根據本文描述瞭解阿里雲ECS側的安全措施。

#### 背景資訊

漏洞Meltdown與Spectre（中文名稱為熔斷和幽靈）是存在於英特爾（Intel）處理器晶片的安全性漏洞。該漏洞事件來源於晶片硬體層面的設計缺陷，可導致作業系統核心資訊泄露、應用程式越權訪問系統核心資料等問題。漏洞編號請在 CVE 網站搜尋查看：分別為：

- [CVE-2017-5753](#)
- [CVE-2017-5715](#)
- [CVE-2017-5754](#)

阿里雲於2018年1月20日發布了[安全性漏洞公告](#)公示漏洞風險詳情及影響範圍。

本工具體介紹阿里雲提供的公用鏡像（GuestOS）安全補丁設定情況，及您如何根據具體情況配置作業系統。預設安全性原則如下所示：

- 對Meltdown漏洞預設採取pti on（Page Table Isolation開啟）措施，開啟Meltdown安全性漏洞補丁。
- 對Spectre漏洞預設採取noibrs（No Indirect Branch Restricted Speculation，禁止間接分支限制推測）措施，結合了reptpoline和ibpb（Indirect Branch Prediction Barriers）的限制策略。

如何開啟或者關閉Meltdown安全性漏洞補丁

已開啟Meltdown安全性漏洞補丁（採取pti on措施）的公用鏡像包括：

- CentOS 7.5/7.6
- Debian 9.6/8.10
- Red Hat 7.5/7.6
- SUSE Linux 15
- Ubuntu 18.04
- CoreOS 1911.3.0
- FreeBSD 11.2
- OpenSUSE 15

以上列表將隨阿里雲公用鏡像的更新而發生變化。

如果您認為開啟PTI對執行個體效能有影響，或者您有其他防護措施，可以根據以下步驟關閉PTI：

1. 遠端連線執行個體。

2. 選擇您要做的操作：

- CentOS、Debian、OpenSUSE、Red Hat、SUSE Linux和Ubuntu：添加核心參數 `nopti`。
- CoreOS：運行 `vi /usr/share/oem/grub.cfg`，在GRUB設定檔中加入 `pti = off`。
- FreeBSD：運行 `vi /boot/loader.conf`，在設定檔中加入 `vm.pmap.pti = 0`。

3. 重啟執行個體。

## 如何開啟或者關閉Spectre安全性漏洞補丁

阿里雲目前支援ibrs和ibpb能力輸出。公用鏡像預設通過retpoline和ibpb（Indirect Branch Prediction Barriers）的限制策略對Spectre漏洞進行防護，同時通過noibrs參數禁用了ibrs。

涉及的公用鏡像包括：

- CentOS 7.5/7.6
- Debian 9.6/8.10
- Red Hat 7.5/7.6
- SUSE Linux 15
- Ubuntu 18.04
- CoreOS 1911.3.0
- FreeBSD 11.2
- OpenSUSE 15

以上列表將隨阿里雲公用鏡像的更新而發生變化。

如果您需要恢復作業系統預設設定，或者認為當前設定對效能有影響或您有其他防護措施需要關閉Spectre安全性漏洞補丁，請按照以下步驟做相應修改：

1. 遠端連線執行個體。
2. 根據下表中的說明選擇您要做的操作：

發行平台	如何恢復阿里雲預設設定	如何恢復作業系統預設設定	如何關閉Spectre漏洞補丁
CentOS	添加核心參數noibrs	刪除核心參數noibrs	添加核心參數 spectre_v2=off
Red Hat			
CoreOS	運行 <code>vi /usr/oem/share/grub.cfg</code> 在GRUB設定檔中添加核心參數 spectre_v2=off	刪除核心參數spectre_v2=off	
OpenSUSE	添加核心參數spectre_v2=off		
Debian	已預設開啟retpoline與ibpb防護措施	無需修改	
Ubuntu			
SUSE Linux	已預設開啟retpoline防護措施		

發行平台	如何恢復阿里雲預設設定	如何恢復作業系統預設設定	如何關閉Spectre漏洞補丁
FreeBSD	添加核心參數hw. ibrs_disable	刪除核心參數hw. ibrs_disable	添加核心參數hw. ibrs_disable



说明:

核心參數 `noibrs` 對OpenSUSE和CoreOS作業系統無效，您需要通過設定 `spectre_v2 = off` 添加防護措施。

3. 重啟執行個體。

如何檢測是否開啟防護

1. 遠端連線執行個體。
2. 在[GitHub spectre-meltdown-checker Repo](#)擷取spectre-meltdown-checker.sh檢測指令碼。
3. 在執行個體內部運行：

```
chmod +x spectre-meltdown-checker.sh
sudo bash spectre-meltdown-checker.sh
```

4. 根據指令碼提示判斷是否已開啟Meltdown或Spectre防護。

相關連結

若您使用的是下列作業系統，可以前往官網瞭解更多詳情：

- [Red Hat](#)
- [SUSE Linux](#)
- [Ubuntu](#)