ALIBABA CLOUD

Alibaba Cloud

云服务器ECS

最佳實務

Document Version: 20200817

Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.

2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.

3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	Danger: Resetting will result in the loss of user configuration data.
Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	Warning: Restarting will cause business interruption. About 10 minutes are required to restart an instance.
Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	Notice: If the weight is set to 0, the server no longer receives new requests.
Note	A note indicates supplemental instructions, best practices, tips, and other content.	Note: You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.安全

1.1. ECS安全性群組實踐（一）

本文主要介紹如何配置安全性群組的入網規則。

在雲端安全性群組提供類似虛擬防火牆功能，用於設定單個或多個 ECS 執行個體的網路存取控制，是重要的安全隔離手段。建立 ECS 執行個體時，您必須選擇一個安全性群組。您還可以添加安全性群組規則，對某個安全性群組下的所有 ECS 執行個體的出方向和入方向進行網路控制。

在配置安全性群組的入網規則之前，您應已經瞭解以下安全性群組相關的資訊：

- 安全性群組限制
- 安全性群組預設規則
- 設定安全性群組 In 方向的存取權限
- 設定安全性群組 Out 方向的存取權限

安全性群組實踐的基本建議

在開始安全性群組的實踐之前，下面有一些基本的建議：

- 最重要的規則：安全性群組應作為白名單使用。
- 開放應用出入規則時應遵循“最小授權”原則，例如，您可以選擇開放具體的連接埠（如 80 連接埠）。
- 不應使用一個安全性群組管理所有應用，因為不同的分層一定有不同的需求。
- 對於分布式應用來說，不同的應用類型應該使用不同的安全性群組，例如，您應對 Web、Service、Database、Cache 層使用不同的安全性群組，暴露不同的出入規則和許可權。
- 沒有必要為每個執行個體單獨設定一個安全性群組，控制管理成本。
- 優先考慮 VPC 網路。
- 不需要公網訪問的資源不應提供公網 IP。
- 儘可能保持單個安全性群組的規則簡潔。因為一個執行個體最多可以加入 5 個安全性群組，一個安全性群組最多可以包括 100 個安全性群組規則，所以一個執行個體可能同時應用數百條安全性群組規則。您可以彙總所有分配的安全規則以判斷是否允許流入或留出，但是，如果單個安全性群組規則很複雜，就會增加管理的複雜度。所以，應儘可能地保持單個安全性群組的規則簡潔。
- 阿里雲的控制台提供了複製安全性群組和安全性群組規則的功能。如果您想要修改線上的安全性群組和規則，您應先複製一個安全性群組，再在複製的安全性群組上進行調試，從而避免直接影響線上應用。

> **說明** 調整線上的安全性群組的出入規則是比較危險的動作。如果您無法確定，不應隨意更新安全性群組出入規則的設定。

設定安全性群組的入網規則

以下是安全性群組的入網規則的實踐建議。

不要使用 0.0.0.0/0 的入網規則

允許全部入網訪問是經常犯的錯誤。使用 0.0.0.0/0 意味著所有的連接埠都對外暴露了存取權限。這是非常不安全的。正確的做法是，先拒絕所有的連接埠對外開放。安全性群組應該是白名單訪問。例如，如果您需要暴露 Web 服務，預設情況下可以只開放 80、8080 和 443 之類的常用TCP連接埠，其它的連接埠都應關閉。

```
{ "IpProtocol" : "tcp", "FromPort" : "80", "ToPort" : "80", "SourceCidrIp" : "0.0.0.0/0", "Policy": "accept"},
{ "IpProtocol" : "tcp", "FromPort" : "8080", "ToPort" : "8080", "SourceCidrIp" : "0.0.0.0/0", "Policy": "accept
"},
{ "IpProtocol" : "tcp", "FromPort" : "443", "ToPort" : "443", "SourceCidrIp" : "0.0.0.0/0", "Policy": "accept"},
```

關閉不需要的入網規則

如果您當前使用的入規則已經包含了 0.0.0.0/0，您需要重新審視自己的應用需要對外暴露的連接埠和服務。如果確定不想讓某些連接埠直接對外提供服務，您可以加一條拒絕的規則。比如，如果您的伺服器上安裝了 MySQL 資料庫服務，預設情況下您不應該將 3306 連接埠暴露到公網，此時，您可以添加一條拒絕規則，如下所示，並將其優先順序設為100，即優先順序最低。

```
{ "IpProtocol" : "tcp", "FromPort" : "3306", "ToPort" : "3306", "SourceCidrIp" : "0.0.0.0/0", "Policy": "drop",
Priority: 100},
```

上面的調整會導致所有的連接埠都不能訪問 3306 連接埠，極有可能會阻止您正常的業務需求。此時，您可以通過授權另外一個安全性群組的資源進行入規則訪問。

授權另外一個安全性群組入網訪問

不同的安全性群組按照最小原則開放相應的出入規則。對於不同的應用分層應該使用不同的安全性群組，不同的安全性群組應有相應的出入規則。

例如，如果是分布式應用，您會區分不同的安全性群組，但是，不同的安全性群組可能網路不通，此時您不應該直接授權 IP 或者 CIDR 網段，而是直接授權另外一個安全性群組 ID 的所有的資源都可以直接存取。比如，您的應用對 Web、Database 分別建立了不同的安全性群組：sg-web 和 sg-database。在sg-database 中，您可以添加如下規則，授權所有的 sg-web 安全性群組的資源訪問您的 3306 連接埠。

```
{ "IpProtocol" : "tcp", "FromPort" : "3306", "ToPort" : "3306", "SourceGroupId" : "sg-web", "Policy": "accep
t", Priority: 2},
```

授權另外一個 CIDR 可以入網訪問

傳統網路中，因為網段不太可控，建議您使用安全性群組 ID 來授信入網規則。

VPC 網路中，您可以自己通過不同的 VSwitch 設定不同的 IP 域，規劃 IP 位址。所以，在 VPC 網路中，您可以預設拒絕所有的訪問，再授信自己的專用網路的網段訪問，直接授信可以相信的 CIDR 網段。

```
{ "IpProtocol" : "icmp", "FromPort" : "-1", "ToPort" : "-1", "SourceCidrIp" : "10.0.0.0/24", Priority: 2},
{ "IpProtocol" : "tcp", "FromPort" : "0", "ToPort" : "65535", "SourceCidrIp" : "10.0.0.0/24", Priority: 2},
{ "IpProtocol" : "udp", "FromPort" : "0", "ToPort" : "65535", "SourceCidrIp" : "10.0.0.0/24", Priority: 2},
```

變更安全性群組規則步驟和說明

變更安全性群組規則可能會影響您的執行個體間的網路通訊。為了保證必要的網路通訊不受影響，您應先嘗試以下方法允許存取必要的執行個體，再執行安全性群組策略收緊變更。

> 說明 執行收緊變更後，應觀察一段時間，確認業務應用無異常後再執行其它必要的變更。

- 建立一個安全性群組，將需要互連訪問的執行個體加入這個安全性群組，再執行變更操作。
- 如果授與類型為 **安全性群組訪問**，則將需要互連訪問的對端執行個體所綁定的安全性群組 ID 添加為授權對象；
- 如果授與類型為 **位址區段訪問**，則將需要互連訪問的對端執行個體內網 IP 添加為授權對象。

具體操作指引請參見 傳統網路內網執行個體互連設定方法。

1.2. ECS安全性群組實踐（二）

本文從授權和撤銷安全性群組規則、加入和移出安全性群組講解Elastic Compute Service的安全性群組最佳實務。

網路類型

阿里雲的網路類型分為傳統網路和Virtual Private Cloud，對安全性群組支援不同的設定規則：

- 如果是傳統網路，您可以設定內網入方向、內網出方向、公網入方向和公網出方向的安全性群組規則。
- 如果是Virtual Private Cloud，您可以設定內網入方向和內網出方向的安全性群組規則。

安全性群組是區分網路類型的，一台傳統網路類型的ECS執行個體只能加入傳統網路的安全性群組。一台Virtual Private Cloud類型的ECS執行個體只能加入本VPC的安全性群組。

安全性群組內網通訊的概念

本文開始之前，您應知道以下幾個安全性群組內網通訊的概念：

- 預設只有同一個安全性群組的ECS執行個體可以網路互連。即使是同一個賬戶下的ECS執行個體，如果分屬不同安全性群組，內網網路也是不通的。這個對於傳統網路和Virtual Private Cloud都適用。所以，傳統網路類型的ECS執行個體也是內網安全的。
- 如果您有兩台ECS執行個體，不在同一個安全性群組，您希望它們內網不互連，但實際上它們卻內網互連，那麼，您需要檢查您的安全性群組內網規則設定。如果內網協議存在下面的協議，建議您重新設定。
  - 允許所有連接埠。
  - 授權對象為CIDR網段（SourceCidrIp）：*0.0.0.0/0*或者*10.0.0.0/8*的規則。如果是傳統網路，上述協議會造成您的內網暴露給其它的訪問。
- 如果您想實現在不同安全性群組的資源之間的網路互連，您應使用安全性群組方式授權。對於內網訪問，您應使用源安全性群組授權，而不是CIDR網段授權。

安全規則的屬性

安全規則主要是描述不同的存取權限，包括如下屬性：

- Policy：授權策略，參數值可以是*accept*（接受）或*drop*（拒絕）。
- Priority：優先順序，根據安全性群組規則的建立時間降序排序匹配。規則優先順序可選範圍為1-100，預設值為1，即最高優先順序。數字越大，代表優先順序越低。
- NicType：網路類型。如果只指定了SourceGroupId而沒有指定SourceCidrIp，表示通過安全性群組方式授權，此時，NicType必須指定為*intranet*。
- 規則描述：
  - IpProtocol：IP協議，取值：*tcp*、*udp*、*icmp*、*gre*或*all*。all表示所有的協議。

- PortRange：IP協議相關的連接埠號碼範圍：
  - IpProtocol取值為*tcp*或*udp*時，連接埠號碼取值範圍為1~65535，格式必須是“開始端點口號/終止連接埠號碼”，如“1/200”表示連接埠號碼範圍為1~200。如果輸入值為“200/1”，介面調用將報錯。
  - IpProtocol取值為*icmp*、*gre*或*all*時，連接埠號碼範圍值為-1/-1，表示不限制連接埠。
- 如果通過安全性群組授權，應指定SourceGroupId，即源安全性群組ID。此時，根據是否跨帳號授權，您可以選擇設定源安全性群組所屬的帳號SourceGroupOwnerAccount。
- 如果通過CIDR授權，應指定SourceCidrIp，即源IP位址區段，必須使用CIDR格式。

授權一條入網請求規則

在控制台或者通過API建立一個安全性群組時，入網方向預設*deny all*，即預設情況下您拒絕所有入網請求。這並不適用於所有的情況，所以您要適度地配置您的入網規則。

比如，如果您需要開啟公網的80連接埠對外提供HTTP服務，因為是公網訪問，您希望入網儘可能多訪問，所以在IP網段上不應做限制，可以設定為*0.0.0.0/0*，具體設定可以參考以下描述，其中，括弧外為控制台參數，括弧內為OpenAPI參數，兩者相同就不做區分。

- 網卡類型（NicType）：公網（internet）。如果是Virtual Private Cloud類型的只需要填寫intranet，通過EIP實現公網訪問。
- 授權策略（Policy）：允許（accept）。
- 規則方向（NicType）：入網。
- 協議類型（IpProtocol）：TCP（tcp）。
- 連接埠範圍（PortRange）：80/80。
- 授權對象（SourceCidrIp）：0.0.0.0/0。
- 優先順序（Priority）：1。

> 說明　上面的建議僅對公網有效。內網請求不建議使用CIDR網段，請參見傳統網路的內網安全性群組規則不要使用 CIDR 或者 IP 授權。

禁止一個入網請求規則

禁止一條規則時，您只需要配置一條拒絕策略，並設定較低的優先順序即可。這樣，當有需要時，您可以配置其它高優先順序的規則覆蓋這條規則。例如，您可以採用以下設定拒絕6379連接埠被訪問。

- 網卡類型（NicType）：內網（intranet）。
- 授權策略（Policy）：拒絕（drop）。
- 規則方向（NicType）：入網。
- 協議類型（IpProtocol）：TCP（tcp）。
- 連接埠範圍（PortRange）：6379/6379。
- 授權對象（SourceCidrIp）：0.0.0.0/0。
- 優先順序（Priority）：100。

傳統網路的內網安全性群組規則不要使用CIDR或者IP授權

對於傳統網路類型的ECS執行個體，阿里雲預設不開啟任何內網的入規則。內網的授權一定要謹慎。

> 說明　為了安全考慮，不建議開啟任何基於CIDR網段的授權。

對於彈性計算來說，內網的IP經常變化，另外，這個IP的網段是沒有規律的，所以，建議您通過安全性群組授權對傳統網路內網的訪問。

例如，您在安全性群組sg-redis上構建了一個redis的叢集，為了只允許特定的機器（如sg-web）訪問這個redis的伺服器編組，您不需要配置任何CIDR，只需要添加一條入規則：指定相關的安全性群組ID即可。

- 網卡類型（NicType）：內網（intranet）。
- 授權策略（Policy）：允許（accept）。
- 規則方向（NicType）：入網。
- 協議類型（IpProtocol）：TCP（tcp）。
- 連接埠範圍（PortRange）：6379/6379。
- 授權對象（SourceGroupId）：sg-web。
- 優先順序（Priority）：1。

對於Virtual Private Cloud類型的執行個體，如果您已經通過多個VSwitch規劃好自己的IP範圍，您可以使用CIDR設定作為安全性群組入規則。但是，如果您的Virtual Private Cloud網段不夠清晰，建議您優先考慮使用安全性群組作為入規則。

將需要互相通訊的ECS執行個體加入同一個安全性群組

一個ECS執行個體最多可以加入5個安全性群組，而同一安全性群組內的ECS執行個體之間是網路互連的。如果您在規劃時已經有多個安全性群組，而且，直接設定多個安全規則過於複雜的話，您可以建立一個安全性群組，然後將需要內網通訊的ECS執行個體加入這個新的安全性群組。

這裡也不建議您將所有的ECS執行個體都加入一個安全性群組，這將會使得您的安全性群組規則設定變成夢魘。對於一個中大型應用來說，每個伺服器編組的角色不同，合理地規劃每個伺服器的入方向請求和出方向請求是非常有必要的。

在控制台上，您可以根據文檔加入安全性群組的描述將一台執行個體加入安全性群組。

如果您對阿里雲的OpenAPI非常熟悉，您可以參見彈性管理ECS執行個體，通過OpenAPI進行大量操作。對應的Python片段如下。

```
def join_sg(sg_id, instance_id):
request = JoinSecurityGroupRequest()
request.set_InstanceId(instance_id)
request.set_SecurityGroupId(sg_id)
response = _send_request(request)
return response
# 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)
```

將ECS執行個體移除安全性群組

如果ECS執行個體加入不合適的安全性群組，將會暴露或者Block您的服務，這時您可以選擇將ECS執行個體從這個安全性群組中移除。但是在移除安全性群組之前必須保證您的ECS執行個體已經加入其它安全性群組。

> 說明 將ECS執行個體從安全性群組移出，將會導致這台ECS執行個體和當前安全性群組內的網路不通，建議您在移出之前做好充分的測試。

對應的Python片段如下。

```
def leave_sg(sg_id, instance_id):
request = LeaveSecurityGroupRequest()
request.set_InstanceId(instance_id)
request.set_SecurityGroupId(sg_id)
response = _send_request(request)
return response
# 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)
```

定義合理的安全性群組名稱和標籤

合理的安全性群組名稱和描述有助於您快速識別當前複雜的規則群組合。您可以通過修改名稱和描述來協助自己識別安全性群組。

您也可以通過為安全性群組設定標籤分組管理自己的安全性群組。您可以在控制台直接設定標籤，也可以通過API設定標籤。

刪除不需要的安全性群組

安全性群組中的安全規則類似於一條條白名單和黑名單。所以，請不要保留不需要的安全性群組，以免因為錯誤加入某台ECS執行個體而造成不必要的麻煩。

1.3. ECS安全性群組實踐（三）

在安全性群組的使用過程中，通常會將所有的雲端服務器放置在同一個安全性群組中，從而可以減少初期配置的工作量。但從長遠來看，業務系統網路的互動將變得複雜和不可控。在執行安全性群組變更時，您將無法明確添加和刪除規則的影響範圍。

合理規劃和區分不同的安全性群組將使得您的系統更加便於調整，梳理應用提供的服務並對不同應用進行分層。這裡推薦您對不同的業務規劃不同的安全性群組，並設定不同的安全性群組規則。

區分不同的安全性群組

- **公網服務的雲端服務器和內網伺服器盡量屬於不同的安全性群組**

  是否對外提供公網服務，包括主動暴露某些連接埠對外訪問（例如 80、443 等），被動地提供連接埠轉寄規則（例如雲端服務器具有公網 IP、EIP、NAT 連接埠轉寄規則等），都會導致自己的應用可能被公網訪問到。

  2 種情境的雲端服務器所屬的安全性群組規則要採用最嚴格的規則，建議拒絕優先，預設情況下應當關閉所有的連接埠和協議，僅僅暴露對外提供需要服務的連接埠，例如 80、443。由於僅對屬於對外公網訪問的伺服器編組，調整安全性群組規則時也比較容易控制。

  對於對外提供伺服器編組的職責應該比較明晰和簡單，避免在同樣的伺服器上對外提供其它的服務。例如 MySQL、Redis 等，建議將這些服務安裝在沒有公網存取權限的雲端服務器上，然後通過安全性群組的組組授權來訪問。

  如果當前有公網雲端服務器已經和其它的應用在同一個安全性群組 SG_CURRENT。您可以通過下面的方法來進行變更。

  i. 梳理當前提供的公網服務暴露的連接埠和協議，例如 80、443。

  ii. 新建立一個安全性群組，例如 SG_WEB， 然後添加相應的連接埠和規則。

  > 說明 授權策略：允許，協議類型：ALL， 連接埠：80/80，授權對象：0.0.0.0/0， 授權策略：允許，協議類型：ALL，連接埠：443/443，授權對象：0.0.0.0/0。

  iii. 選擇安全性群組 SG_CURRENT， 然後添加一條安全性群組規則，組組授權，允許 SG_WEB 中的資源訪問SG_CURRENT。

  > 說明 授權策略：允許，協議類型：ALL，連接埠：-1/-1，授權對象：SG_WEB，優先順序：按照實際情況自訂[1-100]。

  iv. 將一台需要切換安全性群組的執行個體 ECS_WEB_1 添加到新的安全性群組中。

  a. 在 ECS 控制台中，選擇 **安全性群組管理**。

  b. 選擇 **SG_WEB > 管理執行個體 > 添加執行個體**，選擇執行個體 ECS_WEB_1 加入到新的安全性群組 SG_WEB 中，確認 ECS_WEB_1 執行個體的流量和網路工作正常。

  v. 將 ECS_WEB_1 從原來的安全性群組中移出。

  a. 在 ECS 控制台中，選擇 **安全性群組管理**。

  b. 選擇 **SG_WEB > 管理執行個體 > 添加執行個體**，選擇 ECS_WEB_1 ，從 SG_CURRENT 移除，測試網路連通性，確認流量和網路工作正常。

  c. 如果工作不正常，將 ECS_WEB_1 仍然加回到安全性群組 SG_CURRENT 中，檢查設定的 SG_WEB 暴露的連接埠是否符合預期，然後繼續變更。

  vi. 執行其它的伺服器安全性群組變更。

- **不同的應用使用不同的安全性群組**

  在生產環境中，不同的作業系統大多情況下不會屬於同一個應用分組來提供負載平衡服務。提供不同的服務意味著需要暴露的連接埠和拒絕的連接埠是不同的，建議不同的作業系統盡量歸屬於不同的安全性群組。

例如，對於 Linux 作業系統，可能需要暴露 TCP（22）連接埠來實現 SSH，對 Windows 可能需要開通 TCP(3389) 遠端桌面連線。

除了不同的作業系統歸屬不同的安全性群組，即便同一個鏡像類型，提供不同的服務，如果之間不需要通過內網進行訪問的話，最好也劃歸不同的安全性群組。這樣方便解耦，並對未來的安全性群組規則進行變更，做到職責單一。

在規劃和新增應用時，除了考慮劃分不同的虛擬交換器配置子網，也應該同時合理的規劃安全性群組。使用網段+安全性群組約束自己作為服務提供者和消費者的邊界。

具體的變更流程參見上面的操作步驟。

- **生產環境和測試環境使用不同的安全性群組**

  為了更好的做系統的隔離，在實際開發過程中，您可能會構建多套的測試環境和一套線上環境。為了更合理的做網路隔離，您需要對不同的環境配置使用不同的安全性原則，避免因為測試環境的變更重新整理到了線上影響線上的穩定性。

  通過建立不同的安全性群組，限制應用的訪問域，避免生產環境和測試環境聯通。同時也可以對不同的測試環境分配不同的安全性群組，避免多套測試環境之間互相干擾，提升開發效率。

僅對需要公網訪問子網或者雲端服務器分配公網 IP

不論是傳統網路還是Virtual Private Cloud 中，合理的分配公網 IP 可以讓系統更加方便地進行公網管理，同時減少系統受攻擊的風險。在專用網路的情境下，建立虛擬交換器時，建議您盡量將需要公網訪問的服務區的 IP 區間放在固定的幾個交換器(子網 CIDR)中，方便審計和區分，避免不小心暴露公網訪問。

在分布式應用中，大多數應用都有不同的分層和分組，對於不提供公網訪問的雲端服務器盡量不提供公網 IP，如果是有多台伺服器提供公網訪問，建議您配置公網流量分發的負載平衡服務來公網服務，提升系統的可用性，避免單點。

對於不需要公網訪問的雲端服務器盡量不要分配公網 IP。專用網路中當您的雲端服務器需要訪問公網的時候，優先建議您使用 NAT Gateway，用於為 VPC 內無公網 IP 的 ECS 執行個體提供訪問互連網的代理服務，您只需要配置相應的 SNAT 規則即可為具體的 CIDR 網段或者子網提供公網訪問能力，具體配置參見SNAT。避免因為只需要訪問公網的能力而在分配了公網 IP(EIP) 之後也向公網暴露了服務。

最小原則

安全性群組應該是白名單性質的，所以需盡量開放和暴露最少的連接埠，同時儘可能少地分配公網 IP。若想訪問線上機器進行任務日誌或錯誤排查的時候直接分配公網 IP，掛載 EIP 雖然簡便，但是畢竟會將整個機器暴露在公網之上，更安全的策略是通過跳板機來管理。

使用跳板機

跳板機由於其自身的許可權巨大，除了通過工具做好審計記錄。在專用網路中，建議將跳板機分配在專有的虛擬交換器之中，對其提供相應的 EIP 或者 NAT 連接埠轉寄表。

首先建立專有的安全性群組 SG_BRIDGE，例如開放相應的連接埠，例如 Linux TCP(22) 或者 Windows RDP(3389)。為了限制安全性群組的入網規則，可以限制能登入的授權對象為企業的公網出口範圍，減少被登入和掃描的機率。

然後將作為跳板機的雲端服務器加入到該安全性群組中。為了讓該機器能訪問相應的雲端服務器，可以配置相應的組授權。例如在 SG_CURRENT 添加一條規則允許 SG_BRIDGE 訪問某些連接埠和協議。

使用跳板機 SSH 時，建議您優先使用 SSH 金鑰對 而不是密碼登入。

總之，合理的安全性群組規劃使您在擴容應用時更加遊刃有餘，同時讓您的系統更加安全。

1.4. 傳統網路內網執行個體互連設定方法

安全性群組是執行個體層級防火牆，為保障執行個體安全，設定安全性群組規則時要遵循最小授權原則，下面介紹四種安全的內網執行個體互連設定方法。

方法 1. 使用單 IP 位址授權

- 適用情境：適用於小規模執行個體間內網互連情境。
- 優點：以IP地址方式授權，安全性群組規則清晰，容易理解。
- 缺點：內網互連執行個體數量較多時，會受到安全性群組規則條數 100 條的限制，另外後期維護工作量比較大。
- 設定方法：
  i. 選擇需要互連的執行個體，進入**本執行個體安全性群組**。
  ii. 選擇需要配置安全性群組，單擊**配置規則**。
  iii. 單擊**內網入方向**，並單擊**添加安全性群組規則**。
  iv. 按以下描述添加安全性群組規則：
    - **授權策略**：允許。
    - **協議類型**：根據實際需要選擇協議類型。
    - **連接埠範圍**：根據您的實際需要設定連接埠範圍，格式為*開始端點口號/終止連接埠號碼*。
    - **授與類型**：位址區段訪問。
    - **授權對象**：輸入想要內網互連的執行個體的內網 IP 位址，格式必須是 *a.b.c.d/32*。其中，子網路遮罩必須是 */32*。

方法 2. 加入同一安全性群組

- 適用情境：如果您的應用架構比較簡單，可以為所有的執行個體選擇相同的安全性群組，綁定同一安全性群組的執行個體之間不用設定特殊規則，預設網路互連。
- 優點：安全性群組規則清晰。
- 缺點：僅適用於簡單的應用網路架構，網路架構調整時授權方法要隨之進行修改。
- 設定方法：請參見加入、移出安全性群組。

方法 3. 綁定互連安全性群組

- 適用情境：為需要互連的執行個體增加綁定一個專門用於互連的安全性群組，適用於多層應用網路架構情境。
- 優點：操作簡單，可以迅速建立執行個體間互連，可應用於複雜網路架構。
- 缺點：執行個體需綁定多個安全性群組，安全性群組規則閱讀性較差。
- 設定方法：
  i. 建立一個安全性群組並命名，例如：互連安全性群組，不需要給建立的安全性群組添加任何規則。
  ii. 將需要互連的執行個體都添加綁定建立的互連安全性群組，利用同一安全性群組的執行個體之間預設互連的特性，達到內網執行個體互連的效果。

方法 4. 安全性群組互信授權

- 適用情境：如果您的網路架構比較複雜，各執行個體上部署的應用都有不同的業務角色，您就可以選擇使用安全性群組互相授權方式。
- 優點：安全性群組規則結構清晰、閱讀性強、可跨賬戶互連。

- 缺點：安全性群組規則配置工作量較大。
- 設定方法：
  i. 選擇需要建立互信的執行個體，進入**本執行個體安全性群組**。
  ii. 選擇需要配置安全性群組，單擊**配置規則**。
  iii. 單擊**內網入方向**，並單擊**添加安全性群組規則**。
  iv. 按以下描述添加安全性群組規則：
    - **授權策略**：允許。
    - **協議類型**：根據您的實際需要選擇協議類型。
    - **連接埠範圍**：根據實際需求設定。
    - **授與類型**：安全性群組訪問。
    - **授權對象**：
      - 如果您選擇**本帳號授權**：按照您的組網要求，將有內網互連需求的對端執行個體的安全性群組 ID 填入**授權對象**即可。
      - 如果您選擇**跨帳號授權**：**授權對象**應填入對端執行個體的安全性群組 ID，**帳號 ID**是對端帳號 ID（可以在**帳號管理 > 安全設定**裡查到）。

建議

如果前期安全性群組授權過大，建議採用以下流程收緊授權範圍。

圖中的**刪除0.0.0.0**是指刪除原來的允許*0.0.0.0/0*位址區段的安全性群組規則。

如果安全性群組規則變更操作不當，可能會導致您的執行個體間通訊受到影響，請在修改設定前備份您要操作的安全性群組規則，以便出現互連問題時及時恢複。

安全性群組映射了執行個體在整個應用架構中的角色，推薦按照應用架構規劃防火牆規則。例如：常見的三層 Web 應用程式架構就可以規劃三個安全性群組，將部署了相應應用或資料庫的執行個體綁定對應的安全性群組：

- Web 層安全性群組：開放 80 連接埠。
- APP 層安全性群組：開放 8080 連接埠。
- DB 層安全性群組：開放 3306 連接埠。

1.5. 安全性群組內網路隔離

安全性群組是一種虛擬防火牆，具備狀態檢測和包過濾功能。安全性群組由同一個地區內具有相同安全保護需求並相互信任的執行個體組成。為了滿足同安全性群組內執行個體之間網路隔離的需求，阿里雲豐富了安全性群組網路連通策略，支援安全性群組內實現網路隔離。

安全性群組內的網路隔離規則

- 安全性群組內網路隔離是網卡之間的隔離，而不是ECS執行個體之間的隔離。若執行個體上綁定了多張彈性網卡，需要在每個網卡上設定安全性群組隔離規則。
- 不會改變預設的網路連通策略。

安全性群組內網路隔離是一種自訂的網路連通策略，對於預設安全性群組和建立的安全性群組無效。安全性群組預設的網路連通策略是：同一安全性群組內的執行個體之間私網互連，不同安全性群組的執行個體之間預設私網不通。

- 安全性群組內網路隔離的優先順序最低。

  設定了組內網路隔離的安全性群組，僅在安全性群組內沒有任何自訂規則的情況下保證安全性群組內執行個體之間網路隔離。以下情況設定了組內網路隔離但執行個體仍然互連：

  - 安全性群組內既設定了組內隔離，又設定了讓組內執行個體之間可以互相訪問的ACL。
  - 安全性群組內既設定了組內隔離，又設定了組內互連。
- 網路隔離只對當前安全性群組內的執行個體有效。

修改策略

您可以使用ModifySecurityGroupPolicy介面來修改安全性群組內的網路連通策略。

案例分析

執行個體和執行個體所屬的安全性群組的關係如下：

本樣本中，Group1、Group2、Group3分別為3個不同的安全性群組，ECS1、ECS2、ECS3分別為3個不同的ECS執行個體。ECS1和ECS2同屬安全性群組Group1和Group2，ECS2和ECS3同屬安全性群組Group3。

3個安全性群組內的網路連通原則設定如下：

安全性群組	內網連通策略	包含的執行個體
Group1	隔離	ECS1、ECS2
Group2	互連	ECS1、ECS2
Group3	互連	ECS2、ECS3

各執行個體間的網路連通情況如下：

執行個體	網路互連／隔離	原因
ECS1和ECS2	互連	ECS1、ECS2同時屬於Group1和Group2。Group1的策略是隔離，Group2的策略是互連，由於網路隔離的優先順序最低，所以ECS1和ECS2互連。
ECS2和ECS3	互連	ECS2和ECS3同時屬於Group3。Group3的策略是互連，所以ECS2和ECS3互連。
ECS1和ECS3	隔離	ECS1和ECS3分屬不同的安全性群組，不同安全性群組的執行個體之間預設網路不通。如果兩個安全性群組之間需要互相訪問，可以通過安全性群組規則授權。

1.6. 安全性群組五元組規則

安全性群組用於設定單台或多台ECS執行個體的網路存取控制，它是重要的網路安全隔離手段，用於在雲端劃分安全域。安全性群組五元組規則能精確控制源IP、源連接埠、目的IP、目的連接埠以及傳輸層協議。

背景資訊

在最初涉及安全性群組規則時，

- 安全性群組入規則只支援：源IP地址、目的連接埠、傳輸層協議。
- 安全性群組出規則只支援：目的IP地址、目的連接埠、傳輸層協議。

在多數應用情境下，該安全性群組規則簡化了設定，但存在如下弊端：

- 無法限定入規則的源連接埠範圍，預設允許存取所有源連接埠。
- 無法限定入規則的目的IP地址，預設允許存取安全性群組下的所有IP地址。
- 無法限定出規則的源連接埠範圍，預設允許存取所有源連接埠。
- 無法限定出規則的源IP地址，預設允許存取安全性群組下的所有IP地址。

五元組規則定義

五元組規則包含：源IP地址、源連接埠、目的IP地址、目的連接埠、傳輸層協議。

五元組規則完全相容原有的安全性群組規則，能更精確的控制源IP地址、源連接埠、目的IP地址、目的連接埠以及傳輸層協議。

五元組出規則樣本如下：

```
源IP地址：172.16.1.0/32
源連接埠：22
目的IP：10.0.0.1/32
目的連接埠：不限制
傳輸層協議：TCP
授權策略：Drop
```

樣本中的出規則表示禁止172.16.1.0/32通過22連接埠對10.0.0.1/32發起TCP訪問。

應用情境

- 某些平台類網路產品接入第三方廠商的解決方案為使用者提供網路服務，為了防範這些產品對使用者的ECS執行個體發起非法訪問，則需要在安全性群組內設定五元組規則，更精確的控制出流量和入流量。
- 設定了組內網路隔離的安全性群組，如果您想精確控制組內若干ECS執行個體之間可以互相訪問，則需要在安全性群組內設定五元組規則。

配置五元組規則

您可以使用OpenAPI設定五元組規則。

- 增加安全性群組入規則，請參見 AuthorizeSecurityGroup。
- 增加安全性群組出規則，請參見 AuthorizeSecurityGroupEgress。
- 刪除安全性群組入規則，請參見 RevokeSecurityGroup。
- 刪除安全性群組出規則，請參見 RevokeSecurityGroupEgress。

參數說明

在授權或解除授權時，各參數的含義如下表所示。

參數	入規則中各參數含義	出規則中各參數含義
SecurityGroupId	當前入規則所屬的安全性群組ID，即目的安全性群組ID。	當前出規則所屬的安全性群組ID，即源安全性群組ID。
DestCidrIp	目的IP範圍，選擇性參數。 • 如果指定DestCidrIp，則可以更精細地控制入規則生效的目的IP範圍； • 如果不指定DestCidrIp，則入規則生效的IP範圍就是SecurityGroupId這個安全性群組下的所有IP。	目的IP，DestGroupId與DestCidrIp二者必選其一，如果二者都指定，則DestCidrIp優先順序高。
PortRange	目的連接埠範圍，必選參數	目的連接埠範圍，必選參數。
DestGroupId	不允許輸入。目的安全性群組ID一定是SecurityGroupId。	目的安全性群組ID。DestGroupId與DestCidrIp二者必選其一，如果二者都指定，則DestCidrIp優先順序高。
SourceGroupId	源安全性群組ID，SourceGroupId與SourceCidrIp二者必選其一，如果二者都指定，則SourceCidrIp優先順序高。	不允許輸入，出規則的源安全性群組ID一定是SecurityGroupId。
SourceCidrIp	源IP範圍，SourceGroupId與SourceCidrIp二者必選其一，如果二者都指定，則SourceCidrIp優先順序高。	源IP範圍，選擇性參數。 • 如果指定SourceCidrIp，則會更精細地限定出規則生效的源IP。 • 如果不指定SourceCidrIp，則生效的源IP就是SecurityGroupId這個安全性群組下的所有IP。
SourcePortRange	源連接埠範圍，選擇性參數，不填則不限制源連接埠。	源連接埠範圍，選擇性參數，不填則不限制源連接埠。

1.7. 通過API允許不同帳號下的ECS執行個體內網通訊

若您需要實現同一地區下不同帳號的ECS執行個體內網通訊，可以參考本文描述授權安全性群組間互訪。

前提條件

本文調用API的工具為阿里雲CLI，請確保您已安裝並配置了阿里雲CLI。具體操作，請參見安裝CLI和配置CLI。

背景信息

目前授權安全性群組內網通訊有以下兩種，請根據您的實際需求選擇方式。

- ECS執行個體間通訊：授權同一帳號兩台ECS執行個體間的內網通訊。
- 帳號間內網通訊：授權同一帳號同一地區下兩個安全性群組內所有的ECS執行個體的內網通訊，包括授權以後購買的同一安全性群組內的ECS執行個體。

> **說明** 帳號間內網通訊實際上是安全性群組間授權，即授權處於這兩個安全性群組內的ECS執行個體後就可以實現內網通訊。修改安全性群組配置會影響到安全性群組內所有的ECS執行個體，請根據實際需要進行操作，避免影響到ECS執行個體網路下啟動並執行業務。

安全性群組是ECS執行個體的虛擬防火牆，安全性群組本身不提供通訊能力和組網能力。授權不同安全性群組內的執行個體內網通訊後，請同時確保執行個體可以建立內網互連的能力。

- 若執行個體均是傳統網路類型，必須位於同一地區下。
- 若執行個體均是VPC類型，不同VPC間預設內網不通。建議通過公網訪問的方式通訊，或者通過Express Connect、VPN網關和雲企業網等方式提供訪問能力。詳情請參見Express Connect、VPN網關和雲企業網。
- 若執行個體網路類型不同，請設定ClassicLink允許執行個體通訊。具體操作，請參見。
- 若執行個體位於不同地區，建議通過公網訪問的方式通訊，或者通過Express Connect、VPN網關和雲企業網等方式提供訪問能力。詳情請參見Express Connect、VPN網關和雲企業網。

ECS執行個體間通訊

1. 查詢兩台ECS執行個體的內網IP地址和兩台ECS執行個體所處的安全性群組ID。 您可以通過控制台或調用DescribeInstances介面獲得ECS執行個體所屬的安全性群組ID。假設兩台ECS執行個體的資訊如下表所示。

執行個體	IP地址	所屬安全性群組	安全性群組ID
執行個體A	10.0.0.1	sg1	sg-bp1azkttqpldxgtedXXX
執行個體B	10.0.0.2	sg2	sg-bp15ed6xe1yxeycg7XXX

2. 在sg1安全性群組中添加允許存取10.0.0.2的入方向的規則。

```
aliyun ecs AuthorizeSecurityGroup --SecurityGroupId sg-bp1azkttqpldxgtedXXX --RegionId cn-qingdao --IpProtocol all --PortRange=-1/-1. --SourceCidrIp 10.0.0.2 --NicType intranet
```

3. 在sg2安全性群組中添加允許存取10.0.0.1的入方向的規則。

```
aliyun ecs AuthorizeSecurityGroup --SecurityGroupId sg-bp15ed6xe1yxeycg7XXX --RegionId cn-qingdao --IpProtocol all --PortRange=-1/-1. --SourceCidrIp 10.0.0.1 --NicType intranet
```

> **說明**
> - 以上命令中，地區取值為華北 1（青島）*cn-qingdao*，請您根據實際情況修改。
> - 以上命令中，調用AuthorizeSecurityGroup介面添加安全性群組入方向的允許存取規則，主要關注的參數為SecurityGroupId和SourceCidrIp。

4. 等待一分鐘後，使用ping命令測試兩台ECS執行個體之間是否內網互連。

帳號間內網通訊

1. 查詢兩個帳號的帳號名和兩個帳號下對應的安全性群組ID。 您可以通過控制台或調用DescribeInstances介面獲得ECS執行個體所屬的安全性群組ID。假設兩個帳號的資訊如下表所示。

帳號	帳號ID	安全性群組	安全性群組ID
帳號A	a@aliyun.com	sg1	sg-bp1azkttqpldxgtedXXX
帳號B	b@aliyun.com	sg2	sg-bp15ed6xe1yxeycg7XXX

2. 在sg1安全性群組中添加允許存取sg2安全性群組入方向的規則。

```
aliyun ecs AuthorizeSecurityGroup --SecurityGroupId sg-bp1azkttqpldxgtedXXX --RegionId cn-qingdao --IpProtocol all --PortRange=-1/-1. --SourceGroupId sg-bp15ed6xe1yxeycg7XXX --SourceGroupOwnerAccount b@aliyun.com --NicType intranet
```

3. 在sg2安全性群組中添加允許存取sg1安全性群組入方向的規則。

```
aliyun ecs AuthorizeSecurityGroup --SecurityGroupId sg-bp15ed6xe1yxeycg7XXX --RegionId cn-qingdao --IpProtocol all --PortRange=-1/-1. --SourceGroupId sg-bp1azkttqpldxgtedXXX --SourceGroupOwnerAccount a@aliyun.com --NicType intranet
```

> 說明
>
> - 以上命令中，地區取值為華北 1（青島）*cn-qingdao*，請您根據實際情況修改。
> - 以上命令中，調用AuthorizeSecurityGroup介面添加安全性群組入方向的允許存取規則時，主要關注的參數為SecurityGroupId、SourceGroupId和SourceGroupOwnerAccount。

4. 等待一分鐘後，使用ping命令測試查看兩台ECS執行個體之間是否內網互連。

2.資料恢複

2.1. 誤刪檔案後如何恢複資料

本文檔主要以CentOS7作業系統為例，介紹如何使用開源工具Extundelete快速恢複被誤刪除掉的資料。

簡介

在日常使用中有時難免會出現資料被誤刪除的情況，在這個時候該如何快速、有效地恢複資料呢？在阿里雲上恢複資料有多種方式，例如：

- 通過阿里雲控制台復原備份好的快照，自訂鏡像恢複等方式。
- 購買多台ECS，實現業務的負載平衡，高可用。
- 利用Object Storage Service（Object Storage Service），儲存靜態網頁和海量圖片、視頻等重要資料。

在Linux下，基於開源的資料恢複工具有很多，常見的有debugfs、R-Linux、ext3grep、extundelete等，比較常用的有ext3grep和extundelete，這兩個工具的恢複原理基本一樣，只是extundelete功能更加強大。

Extundelete是基於linux的開來源資料恢複軟體。在使用阿里雲的雲端服務器時，如果您不小心誤刪除資料，並且Linux系統也沒有與Windows系統下資源回收筒類似的功能，您可以方便快速安裝此工具。

Extundelete能夠利用inode資訊結合日誌去查詢該inode所在的block位置，以次來尋找和恢複所需的資料，該工具最給力的一點就是支援ext3/ext4雙格式分區恢複，基於整個磁碟的恢複功能較為強大。

在資料被誤刪除後，第一時間要做的是卸載被刪除資料所在的磁碟或磁碟分割。因為將檔案刪除後，僅僅是將檔案的inode結點中的扇區指標清零，實際檔案還儲存在磁碟上，如果磁碟以讀寫入模式掛載，這些已刪除的檔案的資料區塊就可能被作業系統重新分配出去，在這些資料區塊被新的資料覆蓋後，這些資料就真的丟失了，恢複工具也回力無天。所以，以唯讀模式掛載磁碟可以盡量降低資料區塊中資料被覆蓋的風險，以提高恢複資料成功的幾率。

> 說明 在實際線上恢複過程中，切勿將extundelete安裝到您誤刪的檔案所在硬碟，這樣會有一定幾率將需要恢複的資料徹底覆蓋，切記操作前做好快照備份。

適用對象

- 磁碟中檔案誤刪除的使用者，且未對磁碟進行過寫入等操作
- 網站訪問量小、少量 ECS 執行個體的使用者

使用方法

需安裝的軟體及版本：e2fsprogs-devel e2fsprogs gcc-c++ make（編譯器等）Extundelete-0.2.4。

> 說明 extundelete需要libext2fs版本1.39或更高版本來運行，但是對於ext4支援，請確保您有e2fsprogs版本1.41或更新版本（可以通過運行命令“dumpe2fs”並記錄其輸出的版本）。

以上版本是寫文檔時的軟體版本。您下載的版本可能與此不同。

- 部署extundelete工具

```
wget http://zy-res.oss-cn-hangzhou.aliyuncs.com/server/extundelete-0.2.4.tar.bz2
yum -y install bzip2 e2fsprogs-devel e2fsprogs gcc-c++ make #安裝相關依賴和庫
tar -xvjf extundelete-0.2.4.tar.bz2
cd extundelete-0.2.4 #進入程式目錄
./configure #如下圖表示安裝成功
```

```
make && make install
```

這個時候會出現src目錄，下面有個extundelete可執行檔以及相應路徑，如下圖，其實預設檔案安裝在*usr/local/bin*下面，下面示範就在*usr/local/bin*目錄下。

- **使用extundelete，類比資料誤刪除然後恢複的過程**
  i. 檢查ECS現有的磁碟和可用分區，並對/dev/vdb進行分區，格式化，此處不在介紹磁碟分割格式化方式，如果不會的話可以點擊此文檔查看操作方式格式化和掛載資料盤。

  ```
  fdisk -l
  ```

  ii. 將分區後的磁碟掛載到/zhuyun目錄下，然後在/zhuyun下面建立測試檔案hello,寫入test。

  ```
  mkdir /zhuyun #建立zhuyun目錄
  mount /dev/vdb1 /zhuyun #將磁碟掛載到zhuyun目錄下
  echo test > hello #寫入測試檔案
  ```

  iii. 記錄檔案MD5值，md5sum命令用於產生和校正刪除前和恢複後倆個檔案的md5值。

  ```
  md5sum hello
  ```

  iv. 類比刪除hello檔案。

  ```
  rm -rf hello
  cd ~
  fuser -k /zhuyun #結束使用某分區的進程樹（確認沒有資源佔用的話，可以跳過此步）
  ```

  v. 卸載資料盤。

  ```
  umount /dev/vdb1 #任何的檔案恢複工具，在使用前，均要將要恢複的分區卸載或掛載為唯讀，防止資料被覆蓋使用
  ```

  vi. 使用Extundelete工具恢複檔案。

  ```
  extundelete --inode 2 /dev/vdb1 #為尋找某i節點中的內容，使用2則說明為整個分區搜尋，如果需要進入目錄搜尋，只須要指定目錄I節點即可。這是可以看到刪除的檔案名稱和inode
  ```

```
/usr/local/bin/extundelete --restore-inode 12 /dev/vdb1 #恢複刪除的檔案
```

這個時候會在執行命令的同級目錄下出現RECOVERED_FILES目錄，查看是否恢複。

通過md5值查看，前後倆個檔案，一樣說明恢複成功。

```
--restore-inode 12 # --restore-inode 按指定的I節點恢複
--extundelete --restore-all # --restore-all 全部恢複
```

3.執行個體配置

3.1. ECS執行個體資料轉送的實現方式

在資訊化高速發展的今天，伺服器每天都會與其它單機交換大量檔案資料，檔案傳輸對大家來說是家常便飯。因此，其重要性就不言而喻了。檔案傳輸方式各有不同，選擇一款合適自己的檔案傳輸工具，在工作中能起到事半功倍的效果。節省資源、方便傳輸、提升工作效率、加密保護等等。因此，很多檔案傳輸工具應運而生，例如：NC、FTP、SCP、NFS、SAMBA、RSYNC/SERVERSYNC等等，每種方式都有自己的特點。本文將首先簡單介紹一下檔案傳輸的基本原理，然後，詳細介紹類Unix/Linux、Windows平台上熱門檔案傳輸方式，並針對它們各自的特點進行比較，讓讀者對檔案傳輸方式有比較詳盡地瞭解，從而能夠根據不同的需要選擇合適的檔案傳輸方式。

檔案傳輸原理

檔案傳輸是資訊傳輸的一種形式，它是在資料來源和資料宿之間傳送檔案資料的過程，也稱檔案資料通訊。作業系統把檔案資料提取到記憶體中做暫存，再複製到目的地，加密就是在檔案外加了一個殼，檔案本身還是一個整體，複製只是把這個整體轉移到其它地方，不需要解密，只有開啟壓縮包時才需解密。一個大檔案作為一個資料整體，是不可能瞬間從一台主機轉移到其它的主機，傳輸是一個持續的過程，但不是把檔案分割了，因此，如果在傳輸的過程中意外中斷，目標路徑中是不會有傳輸的檔案，另外，如果傳輸的是多個檔案，那麼，這些檔案是按順序分別傳輸，如果中間中斷，則正在傳輸的檔案會傳輸失敗，但是，之前已經傳完的檔案傳輸成功（如果傳輸的是檔案壓縮包，那麼，不管裡面有幾個檔案，它本身被視為一個檔案）。

通常我們看到的 NC、FTP、SCP、NFS 等等，都是可以用來傳輸檔案資料的工具，下面我們將詳細介紹主要檔案傳輸工具的特點以及用法。

NETCAT

在網路工具中有“瑞士軍刀”的美譽，它功能強大，作為網路工具的同時，它傳輸檔案的能力也不容小覷。

常用參數

參數	說明
-g <網關>	設定路由器躍程通訊網關，最多可設定8個
-G <指向器數目>	設定來源路由指向器，其數值為4的倍數
-i <延遲秒數>	設定時間間隔，以便傳送資訊及掃描通訊連接埠
-l	使用監聽模式，管控傳入的資料
-o <輸出檔案>	指定檔案名稱，把往來傳輸的資料以16進位字碼傾倒成該檔案儲存
-p <通訊連接埠>	設定本地主機使用的通訊連接埠
-r	指定本地與遠端主機的通訊連接埠
-u	使用UDP傳輸協議
-v	顯示指令執行過程
-w <逾時秒數>	設定等待連線的時間
-z	使用0輸入/輸出模式，只在掃描通訊連接埠時使用

參數	說明
-n	直接使用IP地址，而不通過網域名稱伺服器

用法舉例

1.連接埠掃描21-24(以IP192.168.2.34為例)。

```
nc -v -w 2 192.168.2.34 -z 21-24
```

返回樣本：

```
nc: connect to 192.168.2.34 port 21 (tcp) failed: Connection refused
Connection to 192.168.2.34 22 port [tcp/ssh] succeeded!
nc: connect to 192.168.2.34 port 23 (tcp) failed: Connection refused
nc: connect to 192.168.2.34 port 24 (tcp) failed: Connection refused
```

2. 從192.168.2.33拷貝檔案到192.168.2.34。

- 在192.168.2.34上：`nc-l 1234 > test.txt`
- 在192.168.2.33上：`nc192.168.2.34 < test.txt`

3.用nc命令操作memcached。

- 儲存資料：`printf “set key 0 10 6rnresultrn” |nc 192.168.2.34 11211`
- 擷取資料：`printf “get keyrn” |nc 192.168.2.34 11211`
- 刪除資料：`printf “delete keyrn” |nc 192.168.2.34 11211`
- 查看狀態：`printf “statsrn” |nc 192.168.2.34 11211`
- 類比top命令查看狀態：`watch “echo stats” |nc 192.168.2.34 11211`
- 清空緩衝：

```
printf “flush_allrn” |nc 192.168.2.34 11211 #謹慎操作，清空了緩衝就沒了
```

SCP 安全拷貝

SCP（ Secure Copy）命令的用法和 RCP 命令格式非常類似，區別就是 SCP 提供更安全保障，SCP 在需要進行驗證時會要求你輸入密碼或口令，一般推薦使用 SCP 命令，因為它比 RCP 更安全。SCP 命令使用 SSH 來傳輸資料，並使用與 SSH 相同的認證模式，提供同樣的安全保障，SSH 是目前較可靠得，為遠程登入工作階段和其他網路服務提供安全性的協議，利用 SSH 協議可以有效防止遠端管理過程中的資訊泄露問題。SCP 是基於 SSH 的應用，所以進行資料轉送的機器上必須支援 SSH 服務。

特點

SCP 類似於RCP, 它能夠保留一個特定檔案系統上的檔案屬性，能夠保留檔案屬性或者需要遞迴的拷貝子目錄。

SCP它具備更好檔案傳輸保密性。與此同時，付出的代價就是檔案傳輸時需要輸入密碼而且涉及到 SSH 的一些配置問題，這些都影響其使用的方便性，對於有特定需求的使用者，是比較合適的傳輸工具。

常用樣本

使用 SCP 命令，需要輸入密碼，如果不想每次都輸入，可以通過配置 SSH，這樣在兩台機器間拷貝檔案時不需要每次都輸入使用者名稱和密碼：

產生 RSA 類型的密鑰：

返回樣本

上述命令產生 RSA 類型的密鑰。在提示密鑰的儲存路徑和密碼時，可以直接斷行符號使用預設路徑和空密碼。這樣，產生的公用密鑰儲存/.ssh/id_rsa.pub，私人密鑰儲存在 /.ssh/id_rsa 。然後把這個金鑰組中的公用密鑰的內容複寫到要訪問的機器上的 /.ssh/authorized_keys 檔案中。這樣，下次再訪問那台機器時，就不用輸入密碼了。

在兩台Linux主機間複製檔案

命令基本格式：

```
scp [選擇性參數] file_source file_target
```

從本地複製到遠程（如下四種方式）：

```
scp local_file remote_username@remote_ip:remote_folder
scp local_file remote_username@remote_ip:remote_file
scp local_file remote_ip:remote_folder
scp local_file remote_ip:remote_file
```

> **說明** 第1,2個指定了使用者名稱，命令執行後需要再輸入密碼，第1個僅指定了遠端目錄，檔案名稱字不變，第2個指定了檔案名稱。

第3,4個沒有指定使用者名稱，命令執行後需要輸入使用者名稱和密碼，第3個僅指定了遠端目錄，檔案名稱字不變，第4個指定了檔案名稱。

從遠程複製到本地：

```
scp root@www.cumt.edu.cn:/home/root/others/music /home/space/music/i.mp3
scp -r www.cumt.edu.cn:/home/root/others/ /home/space/music/
```

> **說明** 從遠程複製到本地，只要將從本地複製到遠端命令的後2個參數調換順序即可。

Rsync

Rsync是linux/Unix檔案同步和傳送工具。用於替代rcp的一個工具，rsync可以通過rsh或ssh使用，也能以daemon模式去運行，在以daemon方式運行時rsync server會開一個873連接埠，等待用戶端去串連。串連時rsync server會檢查口令是否相符，若通過口令查核，則可以通過進行檔案傳輸，第一次連通完成時，會把整份檔案傳輸一次，以後則就只需進行增量備份。

安裝方式

> 說明 可以使用每個發行版本內建的安裝包管理器安裝。

```
sudo apt-get install rsync #在debian、ubuntu 等線上安裝方法;
slackpkg install rsync #Slackware 軟體包線上安裝;
yum install rsync #Fedora、Redhat 等系統安裝方法;
```

源碼編譯安裝:

```
wget http://rsync.samba.org/ftp/rsync/src/rsync-3.0.9.tar.gz
tar xf rsync-3.0.9.tar.gz
cd rsync-3.0.9
./configure && make && make install
```

參數介紹:

參數	說明
-v	詳細模式輸出
-a	歸檔模式，表示以遞迴的方式傳輸檔案，並保持所有檔案屬性不變，相當於使用了組合參數-rlptgoD
-r	對子目錄以遞迴模式處理
-l	保留軟連結
-p	保持檔案許可權
-t	保持檔案時間資訊
-g	保持檔案屬組資訊
-o	保持檔案屬主資訊
-D	保持裝置檔案資訊
-H	保留硬鏈結
-S	對疏鬆檔案進行特殊處理以節省DST的空間
-z	對備份的檔案在傳輸時進行壓縮處理

rsync六種不同的工作模式

- 拷貝本地檔案，將/home/coremail目錄下的檔案拷貝到/cmbak目錄下。

  ```
  rsync -avSH /home/coremail/ /cmbak/
  ```

- 拷貝本地機器的內容到遠程機器。

```
rsync -av /home/coremail/ 192.168.11.12:/home/coremail/
```

- 拷貝遠程機器的內容到本地機器。

```
rsync -av 192.168.11.11:/home/coremail/ /home/coremail/
```

- 拷貝遠程rsync伺服器（daemon形式運行rsync）的檔案到本地機。

```
rsync -av root@172.16.78.192::www /databack
```

- 拷貝本地機器檔案到遠程rsync伺服器（daemon形式運行rsync）中。當DST路徑資訊包含”::”分隔字元時啟動該模式。

```
rsync -av /databack root@172.16.78.192::www
```

- 顯示遠程機的檔案清單。這類似於rsync傳輸，不過只要在命令中省略掉本地機資訊即可。

```
rsync -v rsync://192.168.11.11/data
```

rsync設定檔說明

```
cat/etc/rsyncd.conf #內容如下
port = 873 #連接埠號碼
uid = nobody #指定當模組傳輸檔案的守護進程UID
gid = nobody #指定當模組傳輸檔案的守護進程GID
use chroot = no #使用chroot到檔案系統中的目錄中
max connections = 10 #最大並發串連數
strict modes = yes #指定是否檢查口令檔案的許可權
pid file = /usr/local/rsyncd/rsyncd.pid #指定PID檔案
lock file = /usr/local/rsyncd/rsyncd.lock #指定支援max connection的鎖檔案，預設為/var/run/rsyncd.lock
motd file = /usr/local/rsyncd/rsyncd.motd #定義伺服器資訊的，自己寫 rsyncd.motd 檔案內容
log file = /usr/local/rsyncd/rsync.log #rsync 伺服器的日誌
log format = %t %a %m %f %b
syslog facility = local3
timeout = 300
[conf] #自訂模組
path = /usr/local/nginx/conf #用來指定要備份的目錄
comment = Nginx conf
ignore errors #可以忽略一些IO錯誤
read only = no #設定no，用戶端可以上傳檔案，yes是唯讀
write only = no #no為用戶端可以下載，yes不能下載
hosts allow = 192.168.2.0/24 #可以串連的IP
hosts deny = * #禁止串連的IP
list = false #客戶請求時，使用模組列表
uid = root
gid = root
auth users = backup #串連使用者名稱，和linux系統使用者名稱無關係
secrets file = /etc/rsyncd.pass #驗證密碼檔案
```

3.2. 通過讀寫分離提升資料吞吐效能

一般情況下，對資料庫的讀和寫都在同一個資料庫伺服器中操作時，業務系統效能會降低。為了提升業務系統效能，最佳化使用者體驗，可以通過讀寫分離來減輕主要資料庫的負載。本文分別從應用程式層和系統層來介紹讀寫分離的實現方法。

應用程式層實現方法

應用程式層中直接使用代碼實現，在進入Service之前，使用AOP來做出判斷，是使用寫庫還是讀庫，判斷依據可以根據方法名判斷，比如說以query、find、get等開頭的就走讀庫，其他的走寫庫。

優點：

- 多資料來源切換方便，由程式自動完成。
- 不需要引入中介軟體。

- 理論上支援任何資料庫。

缺點：

- 由程式員完成，營運參與不到。
- 不能做到動態增加資料來源。

系統層實現方法

系統層的實現方法包括以下兩種：

- 使用Distributed Relational Database Service實現讀寫分離。
- 使用中介軟體MySQL-proxy實現讀寫分離。

本教程介紹如何使用中介軟體MySQL-proxy實現讀寫分離。

MySQL proxy

MySQL Proxy是一個處於Client端和MySQL server端之間的簡單程式，它可以監測、分析或改變它們的通訊。它使用靈活，沒有限制，常見的用途包括：Server Load Balancer，故障、查詢分析，查詢過濾和修改等等。

MySQL-proxy原理

MySQL Proxy是一個中介層代理，簡單的說，MySQL Proxy就是一個串連池，負責將前台應用的串連請求轉寄給背景資料庫，並且通過使用lua指令碼，可以實現複雜的串連控制和過濾，從而實現讀寫分離和Server Load Balancer。對於應用來說，MySQL Proxy是完全透明的，應用則只需要串連到MySQL Proxy的監聽連接埠即可。當然，這樣proxy機器可能成為單點失效，但完全可以使用多個proxy機器做為冗餘，在應用伺服器的串連池配置中配置到多個proxy的串連參數即可。

優點：

- 來源程式不需要做任何改動就可以實現讀寫分離。
- 動態添加資料來源不需要重啟程式。

缺點：

- 序依賴於中介軟體，會導致切換資料庫變得困難。
- 由中介軟體做了中轉代理，效能有所下降。

操作步驟

環境說明：

- 主庫IP：121.40.18.26
- 從庫IP：101.37.36.20
- MySQL-proxy代理IP：116.62.101.76

前期準備：

- 1、建立3台ECS，並安裝mysql。
- 2、搭建主從，必須保證主從資料庫資料一致。

主環境

1. 修改mysql設定檔。

```
vim /etc/my.cnf
[mysqld]
server-id=202 #設定伺服器唯一的id，預設是1
log-bin=mysql-bin # 啟用二進位日誌
```

從環境

```
[mysqld]
server-id=203
```

2. 重啟主從伺服器中的MySQL服務。

```
/etc/init.d/mysqld restart
```

3. 在主伺服器上建立帳戶並授權slave。

```
mysql -uroot -p95c7586783
grant replication slave on *.* to 'syncms'@'填寫slave-IP' identified by '123456';
flush privileges;
```

4. 查看主要資料庫狀態。

```
mysql> show master status;
```

5. 配置從資料庫。

```
change master to master_host='填寫master-IP', master_user='syncms', master_password='123456', mas
ter_log_file='mysql-bin.000005', master_log_pos=602;
```

6. 啟動slave同步進程並查看狀態。

```
start slave;
show slave status\G
```

7. 驗證主從同步。

```
mysql> create database testproxy;
mysql> create table testproxy.test1(ID int primary key,name char(10) not null);
mysql> insert into testproxy.test1 values(1,'one');
mysql> insert into testproxy.test1 values(2,'two');
mysql> select * from testproxy.test1;
```

從庫操作

從庫中尋找testproxy.test1表的資料，與主庫一致，主從同步成功

```
select * from testproxy.test1;
```

讀寫分離配置

1.安裝MySQL-Proxy。

```
wget https://cdn.mysql.com/archives/mysql-proxy/mysql-proxy-0.8.5-linux-glibc2.3-x86-64bit.tar.gz
mkdir /alidata
tar xvf mysql-proxy-0.8.5-linux-glibc2.3-x86-64bit.tar.gz
mv mysql-proxy-0.8.5-linux-glibc2.3-x86-64bit/ /alidata/mysql-proxy-0.8.5
```

2.環境變數設定。

```
vim /etc/profile #加入以下內容
PATH=$PATH:/alidata/mysql-proxy-0.8.5/bin
export $PATH
source /etc/profile #使變數立即生效
mysql-proxy -V
```

3.讀寫分離設定。

```
cd /alidata/mysql-proxy-0.8.5/share/doc/mysql-proxy/
vim rw-splitting.lua
```

MySQL Proxy會檢測用戶端串連，當串連沒有超過min_idle_connections預設值時，不會進行讀寫分離預設最小4個(最大8個)以上的用戶端串連才會實現讀寫分離，現改為最小1個最大2個，便於讀寫分離的測試，生產環境中，可以根據實際情況進行調整。

調整前：

調整後：

4.將lua管理指令碼（admin.lua）複製到讀寫分離指令碼(rw-splitting.lua)所在目錄。

```
cp /alidata/mysql-proxy-0.8.5/lib/mysql-proxy/lua/admin.lua /alidata/mysql-proxy-0.8.5/share/doc/mysql-proxy/
```

授權

1.主庫中操作授權，因主從同步的原因，從庫也會執行。

```
mysql -uroot -p95c7586783
grant all on *.* to 'mysql-proxy'@'填寫MySQL Proxy IP' identified by '123456';
flush privileges;
```

2.開啟MySQL-Proxy。

```
mysql-proxy --daemon --log-level=debug --log-file=/var/log/mysql-proxy.log --plugins=proxy -b 填寫master-IP:3306 -r 填寫slave-IP:3306 --proxy-lua-script="/alidata/mysql-proxy-0.8.5/share/doc/mysql-proxy/rw-splitting.lua" --plugins=admin --admin-username="admin" --admin-password="admin" --admin-lua-script="/alidata/mysql-proxy-0.8.5/share/doc/mysql-proxy/admin.lua"
```

3.啟動MySQL-Proxy之後，查看連接埠和相關進程。

```
netstat -tpln
```

```
ps -ef | grep mysql
```

測試讀寫分離

1.關閉從複製

```
stop slave;
```

2.MySQL-Proxy上操作，登入mysql-proxy後台管理。

```
mysql -u admin -padmin -P 4041 -h MySQL-Proxy-IP
select * from backends; #查看狀態
```

第一次串連，會串連到主庫上。

```
mysql -umysql-proxy -p123456 -h 116.62.101.76 -P 4040
insert into testproxy.test1 values(3,'three'); #新增一條資料，由於測試需要，關閉了從複製，因此該資料在主庫中存在，在從庫中不存在
```

多開幾個串連進行測試，當查詢testproxy.test1表的資料顯示是從庫的資料時，讀寫分離成功。

```
mysql -umysql-proxy -p123456 -h 116.62.101.76 -P 4040
select * from testproxy.test1;
```

3.3. 設定Windows作業系統慣用語言

本文使用公用鏡像中的Windows Server 2016英語版作業系統為例，從Windows更新下載語言資源套件，為一台ECS執行個體重新設定慣用語言。

背景信息

Elastic Compute Service僅提供中文版和英文版的Windows Server公用鏡像。如果您需要使用其他語言版本，如阿拉伯語、德語、俄語或日語等，可以根據本文設定ECS執行個體的慣用語言。本文為德語為示範步驟，適用於Windows Server 2012及其以上的版本作業系統。建立使用德語和德語鍵盤設定的自訂鏡像後，您可以使用該自訂鏡像根據自身需求建立任意數量的執行個體。

操作步驟

1. 串連Windows執行個體。串連方式請參見串連方式導航。
2. 開啟PowerShell模組。
3. 運行以下命令臨時禁用WSUS（Windows Server Update Services）更新源。

   ```
   Set-ItemProperty -Path 'HKLM:\SOFTWARE\Policies\Microsoft\Windows\WindowsUpdate\AU' -Name UseWUServer -Value 0
   Restart-Service -Name wuauserv
   ```

4. 找到控制台，單擊Clock, Language, and Region > Language > Add a language。
5. 在Add languages對話方塊中，選擇一種語言，例如Deutsch (German) > Deutsch (Deutschland)，單擊Add。
6. 選擇語言，例如Deutsch (Deutschland)，單擊Move up更改語言優先順序。
7. 單擊所選語言右側的Options，線上檢查語言更新。
8. 等待執行個體檢查更新，大約三分鐘後更新會提示可供下載，單擊Download and install language pack。
9. 等待安裝完成。
10. 在ECS控制台重新啟動執行個體。

11. 再次串連Windows執行個體。 顯示語言會在重啟登入後更改為德語。
12. 開啟PowerShell ISE模組，運行以下命令重新啟用WSUS。

```
Set-ItemProperty -Path 'HKLM:\SOFTWARE\Policies\Microsoft\Windows\WindowsUpdate\AU' -Name UseWUServer -Value 1
Restart-Service -Name wuauserv
```

13. 開啟Windows Update，檢查安全更新，重新安裝配置語言設定之前已完成的所有安全更新。

後續步驟

您可以使用相同語言設定建立多台執行個體：

1. 登入 ECS管理主控台 。
2. 根據該Windows執行個體建立自訂鏡像。
3. 通過自訂鏡像建立指定數量的執行個體。

4.Block Storage

5.擴容資料盤_Linux

隨著業務的增長，您的資料盤容量可能無法滿足資料存放區的需要，這時您可以使用 **磁碟擴容** 功能擴容資料盤。

> **說明**
>
> - 掛載在執行個體上的資料盤，只有當執行個體處於 **運行中 (Running)** 或 **已停止(Stopped)** 狀態時才可以擴容。**擴容這種資料盤需要在控制台上重啟執行個體後才能使擴容後的容量生效，而重啟執行個體會停止執行個體，中斷您的業務，所以請您謹慎操作。**
> - 建議在擴容資料盤之前手動建立快照，以備份資料。
> - 無論資料盤的狀態是 **待掛載** 還是 **使用中**，都可以執行磁碟擴容操作。
> - 訂用帳戶執行個體如果做過 **續費降配** 操作，當前計費周期的剩餘時間內，執行個體上的訂用帳戶雲端硬碟不支援擴容磁碟操作。
> - 如果資料盤正在建立快照，則不允許執行擴容資料盤的操作。
> - 磁碟擴容功能只能擴容資料盤，不能擴容系統硬碟或本地碟（本地 SSD 盤等）。

本文以一個高效雲端硬碟的資料盤和一個運行CentOS 7.3 64位的 ECS 執行個體為例，說明如何擴容資料盤並使擴容後的容量可用。

您可以按以下步驟完成擴容操作：

步驟 1. 在控制台上擴容資料盤的磁碟空間

步驟 2. 登入執行個體擴容檔案系統

步驟 1. 在控制台上擴容資料盤的磁碟空間

按以下步驟在控制台上擴容資料盤的磁碟空間：

1. 登入 ECS管理主控台。
2. 在左側導覽列裡，選擇 **儲存 > 雲端硬碟** 。

   > **說明** 如果您需要擴容的資料盤已經掛載在某個執行個體上，您可以單擊 **執行個體**，找到相應執行個體後，進入執行個體詳情頁，並單擊 **本執行個體磁碟**。

3. 選擇地區。
4. 找到需要擴容的磁碟，並在 **操作** 列中，選擇 **更多 > 磁碟擴容** 。
5. 在 **磁碟擴容** 頁面上，設定 **擴容後容量**，在本樣本中為30 GiB。擴容後容量只能比當前容量大。
6. 待頁面上顯示費用資訊後，單擊 **確定擴容**。

   > **說明** 擴容成功後，磁碟列表裡即顯示擴容後的容量。但是，如果您的資料盤已經掛載到執行個體上，只有在控制台上 **重啟執行個體** 後，登入執行個體才能看到新的磁碟空間容量。

在控制台上擴容資料盤的磁碟空間後，

- 如果資料盤已經掛載到執行個體上，您必須執行 步驟 2. 登入執行個體擴容檔案系統。
- 如果資料盤未掛載到執行個體上，您必須先掛載資料盤（參見 掛載雲端碟），再根據資料盤的實際情況執行不同的操作：

- 如果這是一個未格式化的資料盤，您必須格式化資料盤。詳細資料，請參見 Linux 格式化和掛載資料盤。
- 如果這個資料盤之前已經格式化並分區，您必須 步驟 2. 登入執行個體擴容檔案系統。

步驟 2. 登入執行個體擴容檔案系統

在ECS控制台上完成磁碟擴容後，磁碟每個分區的檔案系統並未擴容。您需要登入執行個體擴容檔案系統。

在本樣本中，假設資料盤掛載在一台Linux執行個體上，執行個體的作業系統為CentOS 7.3 64位，未擴容前的資料盤只有一個主要磁碟分割（/dev/vdb1，ext4檔案系統），檔案系統的掛載點為 */resizetest*，檔案系統擴容完成後，資料盤仍然只有一個主要磁碟分割。

1. 使用使用者名密碼驗證串連 Linux 執行個體。
2. 運行 `umount` 命令卸載主要磁碟分割。

```
umount /dev/vdb1
```

> **說明** 使用 `df -h` 查看是否卸載成功，如果看不到 /dev/vdb1 的資訊表示卸載成功。以下為樣本輸出結果。

```
[root@iXXXXXX ~]# df -h
Filesystem Size Used Avail Use% Mounted on
/dev/vda1 40G 1.5G 36G 4% /
devtmpfs 487M 0 487M 0% /dev
tmpfs 497M 0 497M 0% /dev/shm
tmpfs 497M 312K 496M 1% /run
tmpfs 497M 0 497M 0% /sys/fs/cgroup
tmpfs 100M 0 100M 0% /run/user/0
```

3. 使用 `fdisk` 命令刪除原來的分區並建立新分區：

> **說明** 如果您使用 `parted` 工具操作分區，不能與 `fdisk` 交叉使用，否則會導致分區的起始扇區不一致。關於 `parted` 工具的使用說明可以參考這裡。

   i. 運行命令 `fdisk -l` 羅列分區資訊並記錄擴容前資料盤的最終容量、起始扇區（First sector）位置。
   ii. 運行命令 `fdisk [資料盤裝置名稱]` 進入 `fdisk` 介面。本樣本中，命令為 `fdisk /dev/vdb` 。
   iii. 輸入 `d` 並按斷行符號鍵，刪除原來的分區。

> **說明** 刪除分區不會造成資料盤內資料的丟失。

   iv. 輸入 `n` 並按斷行符號鍵，開始建立新的分區。

v. 輸入 `p` 並按斷行符號鍵，選擇建立主要磁碟分割。因為建立的是一個單分區資料盤，所以只需要建立主要磁碟分割。

> 說明 如果要建立4個以上的分區，您應該建立至少一個擴充分區，即選擇 `e` 。

vi. 輸入分區編號並按斷行符號鍵。因為這裡僅建立一個分區，所以輸入 *1*。

vii. 輸入第一個可用的扇區編號：為了保證資料的一致性，First sector需要與原來的分區保持一致。在本樣本中，按斷行符號鍵採用預設值。

> 說明 如果發現First sector顯示的位置和之前記錄的不一致，說明之前可能使用 `parted` 來分區，那麼就停止當前的 `fdisk` 操作，使用 `parted` 重新操作。

viii. 輸入最後一個扇區編號：因為這裡僅建立一個分區，所以按斷行符號鍵採用預設值。

ix. 輸入 wq 並按斷行符號鍵，開始分區。

```
[root@iXXXXXX ~]# fdisk /dev/vdb
Welcome to fdisk (util-linux 2.23.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
Command (m for help): d
Selected partition 1
Partition 1 is deleted
Command (m for help): n
Partition type:
p primary (0 primary, 0 extended, 4 free)
e extended
Select (default p):
Using default response p
Partition number (1-4, default 1):
First sector (2048-62914559, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-62914559, default 62914559):
Using default value 62914559
Partition 1 of type Linux and of size 30 GiB is set
Command (m for help): wq
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

> 說明 如果您使用的是 parted 工具，進入 parted 介面後，輸入 p 羅列當前的分區情況。如果有分區，則使用 rm+ 序號來刪除老的分區表，然後使用 unit s 定義起始位置，單位使用扇區個數計量，最後使用 mkpart 命令來建立即可，如下圖所示。

4. （可選）部分作業系統裡，修改分區後可能會重新自動掛載檔案系統。建議先執行 df -h 重新查看檔案系統空間和使用方式。如果檔案系統重新被掛載，執行 umount [檔案系統名稱] 再次卸載檔案系統。
5. 檢查檔案系統，並變更檔案系統大小。

```
e2fsck -f /dev/vdb1 # 檢查檔案系統
resize2fs /dev/vdb1 # 變更檔案系統大小
```

> 說明
> - 使用 `e2fsck` 時，由於系統需要檢查並訂本文件系統元資料，所以速度較慢、耗時較長，請耐心等待。
> - 正確使用 `e2fsck` 和 `resize2fs` 指令，不會造成原有資料丟失。

以下為樣本輸出結果。

```
[root@iXXXXXX ~]# e2fsck -f /dev/vdb1
e2fsck 1.42.9 (28-Dec-2013)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/vdb1: 11/1835008 files (0.0% non-contiguous), 159218/7339776 blocks
[root@iXXXXXX ~]# resize2fs /dev/vdb1
resize2fs 1.42.9 (28-Dec-2013)
Resizing the filesystem on /dev/vdb1 to 7864064 (4k) blocks.
The filesystem on /dev/vdb1 is now 7864064 blocks long.
```

6. 將擴容完成的檔案系統掛載到原來的掛載點（如本樣本中的 */resizetest*）。

```
mount /dev/vdb1 /resizetest
```

7. 查看檔案系統空間和使用方式：運行命令 `df -h` 。如果出現擴容後的檔案系統資訊，說明掛載成功，可以使用擴容後的檔案系統了。

> 說明 掛載操作完成後，不需要在控制台上重啟執行個體即可開始使用擴容後的檔案系統。

以下為樣本輸出結果。

```
[root@iXXXXXX ~]# df -h
Filesystem Size Used Avail Use% Mounted on
/dev/vda1 40G 1.5G 36G 4% /
devtmpfs 487M 0 487M 0% /dev
tmpfs 497M 0 497M 0% /dev/shm
tmpfs 497M 312K 496M 1% /run
tmpfs 497M 0 497M 0% /sys/fs/cgroup
tmpfs 100M 0 100M 0% /run/user/0
/dev/vdb1 30G 44M 28G 1% /resizetest
```

6.监控

7.GPU執行個體最佳實務

7.1. 在GPU執行個體上使用RAPIDS加速機器學習任務

本文介紹了如何在GPU執行個體上基於NGC環境使用RAPIDS加速庫，加速資料科學和機器學習任務，提高計算資源的使用效率。

背景資訊

RAPIDS，全稱Real-time Acceleration Platform for Integrated Data Science，是NVIDIA針對資料科學和機器學習推出的GPU加速庫。更多RAPIDS資訊請參見官方網站。

NGC，全稱NVIDIA GPU CLOUD，是NVIDIA推出的一套深度學習生態系統，供開發人員免費訪問深度學習和機器學習軟體堆棧，快速搭建相應的開發環境。NGC網站提供了RAPIDS的Docker鏡像，預裝了相關的開發環境。

JupyterLab是一套互動開發環境，協助您高效地瀏覽、編輯和執行伺服器上的代碼檔案。

Dask是一款輕量級大資料架構，可以提升並行計算效率。

本文提供了一套基於NVIDIA的RAPIDS Demo代碼及資料集修改的範例程式碼，示範了在GPU執行個體上使用RAPIDS加速一個從ETL到ML Training端到端任務的過程。其中，ETL時使用RAPIDS的cuDF，ML Training時使用XGBoost。本文範例程式碼基於輕量級大資料架構Dask運行，為一套單機啟動並執行代碼。

> 說明 NVIDIA官方RAPIDS Demo代碼請參見Mortgage Demo。

前提條件

- 註冊阿里雲帳號並完成實名認證，請參見阿里雲帳號註冊流程和 個人實名認證 。
- 在NGC註冊頁面註冊NGC帳號。
- 擷取NGC API Key。
  i. 登入NGC網站。
  ii. 前往CONFIGURATION，單擊Get API Key。
  iii. 單擊Generate API Key。
  iv. 在Generate a New API Key中，單擊Confirm。

  > 說明 新的NGC API Key會覆蓋舊的NGC API Key。如果您已持有NGC API Key，請確保不再需要舊的NGC API Key。

  v. 複製API Key並儲存到本地。

步驟一：擷取RAPIDS鏡像下載命令

1. 登入NGC網站。
2. 開啟MACHINE LEARNING頁面，單擊RAPIDS鏡像。
3. 擷取docker pull命令。

本文範例程式碼基於RAPIDS 0.6版本鏡像編寫，因此在運行本範例程式碼時，使用Tag為0.6版本的鏡像。實際操作時，請選擇您匹配的版本。

i. 選擇Tags頁簽。

ii. 找到並複製Tag資訊。本樣本中，選擇 `0.6-cuda10.0-runtime-ubuntu16.04-gcc5-py3.6` 。

iii. 返回頁面頂部，複製Pull Command中的命令到文字編輯器，將鏡像版本替換為對應的Tag資訊，並儲存。本樣本中，將 `cuda9.2-runtime-ubuntu16.04` 替換為 `0.6-cuda10.0-runtime-ubuntu16.04-gcc5-py3.6` 。

儲存的docker pull命令用於在步驟二中下載RAPIDS鏡像。

步驟二：部署RAPIDS環境

1. 建立一台GPU執行個體。

   詳細步驟請參見使用嚮導建立執行個體。

   - **執行個體**：RAPIDS僅適用於特定的GPU型號（採用NVIDIA Pascal及以上架構），因此您需要選擇GPU型號符合要求的執行個體規格，目前有gn6i、gn6v、gn5和gn5i，詳細的GPU型號請參見執行個體規格類型系列。建議您選擇顯存更大的gn6i、gn6v或gn5執行個體。本樣本中，選用了顯存為16 GB的GPU執行個體。
   - **鏡像**：在鏡像市場中搜尋並使用 `NVIDIA GPU Cloud VM Image` 。
   - **公網頻寬**：選擇**分配公網IPv4地址**或者在執行個體建立成功後綁定EIP地址。
   - **安全性群組**：選擇的安全性群組需要開放以下連接埠：
     - TCP 22 連接埠，用於SSH登入
     - TCP 8888連接埠，用於支援訪問JupyterLab服務
     - TCP 8787連接埠、TCP 8786連接埠，用於支援訪問Dask服務

2. 串連GPU執行個體。

   串連方式請參見串連Linux執行個體。

3. 輸入NGC API Key後按斷行符號鍵，登入NGC容器環境。

4. （可選）運行**nvidia-smi**查看GPU型號、GPU驅動版本等GPU資訊。

   建議您瞭解GPU資訊，預判規避潛在問題。例如，如果NGC的驅動版本太低，新Docker鏡像版本可能會不支援。

5. 運行在步驟一中擷取的docker pull命令下載RAPIDS鏡像。

   ```
   docker pull nvcr.io/nvidia/rapidsai/rapidsai:0.6-cuda10.0-runtime-ubuntu16.04-gcc5-py3.6
   ```

6. （可選）查看下載的鏡像。

   建議您查看Docker鏡像資訊，確保下載了正確的鏡像。

   ```
   docker images
   ```

7. 運行容器部署RAPIDS環境。

```
docker run --runtime=nvidia \
--rm -it \
-p 8888:8888 \
-p 8787:8787 \
-p 8786:8786 \
nvcr.io/nvidia/rapidsai/rapidsai:0.6-cuda10.0-runtime-ubuntu16.04-gcc5-py3.6
```

步驟三：運行RAPIDS Demo

1. 在GPU執行個體上下載資料集和Demo檔案。

```
# 擷取apt源地址並下載指令碼（指令碼功能：下載訓練資料、notebook、utils）
$ source_address=$(curl http://100.100.100.200/latest/meta-data/source-address|head -n 1)
$ source_address="${source_address}/opsx/ecs/linux/binary/machine_learning/"
$ wget $source_address/rapids_notebooks_v0.6/utils/download_v0.6.sh
# 執行下載指令碼
$ sh ./download_v0.6.sh
# 切換到下載目錄查看下載檔案
$ apt update
$ apt install tree
$ tree /rapids/rapids_notebooks_v0.6/
```

   下載成功後的檔案結構如下圖，共5個檔案夾、16個檔案：

2. 在GPU執行個體上啟動JupyterLab服務。

   推薦直接使用命令啟動。

```
# 切換到工作目錄
$ cd /rapids/rapids_notebooks_v0.6/xgboost
# 啟動jupyter-lab，直接使用命令啟動，並設定登入密碼
$ jupyter-lab --allow-root --ip=0.0.0.0 --no-browser --NotebookApp.token='登入密碼'
# 退出
$ sh ../utils/stop-jupyter.sh
```

   - 除使用命令外，您也可以執行指令碼 `$ sh ../utils/start-jupyter.sh` 啟動jupyter-lab，此時無法設定登入密碼。
   - 您也可以連續按兩次 `Ctrl+C` 退出。

3. 開啟瀏覽器，在地址欄輸入 `http://您的GPU執行個體IP地址:8888` 遠端存取JupyterLab 。

   > 說明 推薦使用Chrome瀏覽器。

   如果您在啟動JupyterLab服務時設定了登入密碼，會跳轉到密碼輸入介面。

4. 運行NoteBook代碼。

   該案例是一個抵押貸款迴歸的任務，詳細資料請參見代碼執行過程。登入成功後，可以看到NoteBook代碼的程式碼封裝括以下內容：

   - *mortgage_2000_1gb*檔案夾：儲存解壓後的訓練資料。該檔案夾下包含：acq檔案夾、perf檔案夾和names.csv檔案。
   - *xgboost_E2E.ipynb*檔案： XGBoost Demo檔案。雙擊檔案可以查看檔案詳情，單擊下圖中的執行按鈕可以逐步執行代碼，每次執行一個Cell。
   - *mortgage_2000_1gb.tgz*檔案： 2000年的抵押貸款迴歸訓練資料（1G分割的perf檔案夾下的檔案不會大於1G，使用1G分割的資料可以更有效利用GPU顯存）。

代碼執行過程

該案例基於XGBoost示範了資料預先處理到訓練的端到端的過程，主要分為三個階段：

- ETL（Extract-Transform-Load）：主要在GPU執行個體上進行。將業務系統的資料經過抽取、清洗轉換之後載入到資料倉儲。
- Data Conversion：在GPU執行個體上進行。將在ETL階段處理過的資料轉換為用於XGBoost訓練的DMatrix格式。
- ML-Training：預設在GPU執行個體上進行。使用XGBoost訓練梯度提升決策樹 。

NoteBook代碼的執行過程如下：

1. 準備資料集。

   本案例的Shell指令碼會預設下載2000年的抵押貸款迴歸訓練資料（*mortgage_2000_1gb.tgz*），並解壓到*mortgage_2000_1gb*檔案夾。

   如果您想擷取更多資料用於XGBoost模型訓練，可以設定參數download_url指定下載路徑，具體下載地址請參見Mortgage Data。

   樣本效果如下：

2. 設定相關參數。

參數名稱	說明
start_year	指定選擇訓練資料的起始時間，ETL時會處理start_year到end_year之間的資料。
end_year	指定選擇訓練資料的結束時間，ETL時會處理start_year到end_year之間的資料。
train_with_gpu	是否使用GPU進行XGBoost模型訓練，預設為*True*。
gpu_count	指定啟動worker的數量，預設為*1*。您可以按需要設定參數值，但不能超出GPU執行個體的GPU數量。
part_count	指定用於模型訓練的performance檔案的數量，預設為 *2 * gpu_count*。如果參數值過大，在Data Conversion階段會報錯超出GPU記憶體限制，錯誤資訊會在NoteBook後台輸出。

   樣本效果如下：

3. 啟動Dask服務。

代碼會啟動Dask Scheduler，並根據gpu_count參數啟動worker用於ETL和模型訓練。

樣本效果如下：

4. 啟動ETL。

   ETL階段會進行到表關聯、分組、彙總、切片等操作，資料格式採用cuDF庫的DataFrame格式（類似於pandas的DataFrame格式）。

   樣本效果如下：

5. 啟動Data Conversion。

   將DataFrame格式的資料轉換為用於XGBoost訓練的DMatrix格式，每個worker處理一個DMatrix對象。

   樣本效果如下：

6. 啟動ML Training。

   使用dask-xgboost啟動模型訓練，dask-xgboost負責多個dask worker間的通訊協同工作，底層仍然調用xgboost執行模型訓練。

   樣本效果如下：

相關函數

函數功能	函數名稱
下載檔案	def download_file_from_url(url, filename):
解壓檔案	def decompress_file(filename, path):
擷取當前機器的GPU個數	def get_gpu_nums():
管理GPU記憶體	• def initialize_rmm_pool(): • def initialize_rmm_no_pool(): • def run_dask_task(func, **kwargs):
提交DASK任務	• def process_quarter_gpu(year=2000, quarter=1, perf_file=""): • def run_gpu_workflow(quarter=1, year=2000, perf_file="", **kwargs):

函數功能	函數名稱
使用cuDF從CSV中載入資料	• def gpu_load_performance_csv(performance_path, **kwargs): • def gpu_load_acquisition_csv(acquisition_path, **kwargs): • def gpu_load_names(**kwargs):
處理和提取訓練資料的特徵	• def null_workaround(df, **kwargs): • def create_ever_features(gdf, **kwargs): • def join_ever_delinq_features(everdf_tmp, delinq_merge, **kwargs): • def create_joined_df(gdf, everdf, **kwargs): • def create_12_mon_features(joined_df, **kwargs): • def combine_joined_12_mon(joined_df, testdf, **kwargs): • def final_performance_delinquency(gdf, joined_df, **kwargs): • def join_perf_acq_gdfs(perf, acq, **kwargs): • def last_mile_cleaning(df, **kwargs):

8.FaaS執行個體最佳實務

8.1. faascmd工具

8.1.1. faascmd工具概述

faascmd是阿里雲FPGA雲端服務器（FaaS）提供的一個命令列工具，是基於python SDK開發的指令碼。

您可以使用faascmd工具：

- 進行授權及相關操作
- 管理和操作FPGA鏡像
- 查看和上傳objects
- 擷取FPGA執行個體資訊

8.1.2. 配置faascmd

在使用faascmd之前，您需要配置相關環境變數和RAM使用者的AccessKey。

操作步驟

1. 登入您的執行個體後，運行以下命令配置PATH環境變數。

```
export PATH=$PATH:<faascmd工具所在路徑>
```

2. 運行下列命令配置AccessKey ID和AccessKey Secret。

```
faascmd config --id=<yourAccessKeyID> --key=<yourAccessKeySecret>
```

8.1.3. 使用faascmd

您可以通過本主題瞭解faascmd命令的用法。

前提條件

使用faascmd工具之前，您需要先 配置faascmd 。

文法說明

- faascmd工具提供的所有命令和參數都嚴格區分大小寫。
- faascmd命令中各參數“=”前後不能有多餘空格。

授權

`faascmd auth` 命令用於授權faas admin訪問使用者的OSS bucket。

前提條件

1. 為FaaS建立一個OSSbucket，用於上傳原始編譯的DCP檔案。

2. 在該FaaSOSSbucket中，建立一個名為compiling_logs的檔案夾。

命令格式

```
faascmd auth --bucket=<yourFaasOSSBucketName>
```

範例程式碼

> 說明 如果同一主賬戶下有多個子賬戶，建議子賬戶間共用一個OSS bucket，以避免重複修改或覆蓋授權策略。

查看授權策略

`faascmd list_policy` 命令用來查看指定的OSS bucket是否已添加到相應的授權策略（faasPolicy）裡。

命令格式

```
faascmd list_policy
```

範例程式碼

> 說明 請關注您的OSS Bucket和OSS Bucket/compiling_logs是否出現在列出的策略資訊中。

刪除授權策略

`faascmd delete_policy` 命令用於刪除授權策略（faasPolicy）。

命令格式

```
faascmd delete_policy
```

範例程式碼

> 說明 如果同一主賬戶下有多個子賬戶，建議您去RAM控制台操作，以避免誤刪授權策略。

查看OSS Bucket下所有的objects

`faascmd list_objects` 命令用於查看使用者OSS Bucket下所有的objects。

命令格式

```
faascmd list_objects
```

範例程式碼

> 說明 您可以配合grep命令篩選出您想要的檔案。例如：`faascmd list_objects | grep "xxx"` 。

上傳原始編譯檔案

`faascmd upload_object` 命令用於將本地編譯的原始檔案上傳到使用者指定的OSS bucket中。

命令格式

```
faascmd upload_object --object=<newFileNameinOSSBucket> --file= <your_file_path>/fileNameYouWa
ntToUpload
```

範例程式碼

> 說明
> - 如果需上傳的檔案在目前的目錄下，則無需提供路徑。
> - intel fpga的本地編譯原始檔案為.gbs格式；xilinx fpga的本地編譯原始檔案為指令碼處理後得到的tar包。

下載OSS Bucket中的object

`faascmd get_object` 命令用來下載OSS Bucket中指定的object。

命令格式

```
faascmd get_object --obejct=<yourObjectName> --file=<your_local_path>/<yourFileName>
```

範例程式碼

> 說明 如果您不提供路徑，則預設下載到當前檔案夾。

建立fpga鏡像

`faascmd create_image` 命令用來提交製作fpga鏡像的請求。請求成功時，返回fpga imageuuid。

命令格式

```
faascmd create_image --object=<yourObjectName>
--fpgatype=<intel/xilinx> --encrypted=<true/false>
--kmskey=<key/如果encrypted為true，必須；否則可選>
--shell=<Shell Version/必選> --name=<name/可選>
--description=<description/可選> --tags=<tags/可選>
```

範例程式碼

查看fpga鏡像

`faascmd list_images` 命令用於查看使用者製作的所有fpga鏡像的資訊。

命令格式

```
faascmd list_images
```

範例程式碼

> **說明** 每個子賬戶最多允許保留10個fpga鏡像。

刪除fpga鏡像

`faascmd delete_image` 命令用於刪除fpga鏡像。

命令格式

```
faascmd delete_image --imageuuid=<yourImageuuid>
```

範例程式碼

下載fpga鏡像

`faascmd download_image` 命令用於提交下載fpga鏡像的請求。

命令格式

```
faascmd download_image --instanceId=<yourInstanceId>
--fpgauuid=<yourfpgauuid> --fpgatype=<intel/xilinx>
--imageuuid=<yourImageuuid> --imagetype=<afu>
--shell=<yourImageShellVersion>
```

範例程式碼

```
faascmd download_image --instanceId=XXXXX --fpgauuid=XXXX --fpgatype=intel --imageuuid=XXXX
```

查看fpga鏡像下載狀態

`faascmd fpga_status` 命令用於查看當前fpga板卡狀態或fpga鏡像的下載進度。

命令格式

```
faascmd fpga_status --fpgauuid=<fpgauuid> --instanceId=<instanceId>
```

範例程式碼

發布fpga鏡像

`faascmd publish_image` 命令用來提交發布fpga鏡像的請求。

命令格式

```
faascmd publish_image --imageuuid=<yourImageuuid> --imageid=<yourFPGAImageid>
```

> 說明
>
> - imageuuid 是您要發布到雲市場的鏡像id。您可以通過 `faascmd list_images` 命令查看。
> - imageid 是fpga鏡像id。您可以通過ECS控制台的執行個體詳情頁查看。

查看fpga執行個體的資訊

`faascmd list_instances` 命令用於擷取fpga執行個體的基本資料，包括執行個體id、fpga板卡資訊和shell版本。

命令格式

```
faascmd list_instances --instanceId=<yourInstanceId>
```

範例程式碼

8.1.4. faascmd工具FAQ

本文介紹使用faascmd工具時常見的問題與解決辦法。

常見問題

- **Name Error:global name'ID' is not defined.**

  原因：faascmd沒有擷取到您的AccessKeyID或AccessKeySecret資訊。

  解決辦法：執行 `faascmd config` 命令，此命令執行後，會將您輸入的AccessKeyID和AccessKeySecret資訊儲存在檔案 `/root/.faascredentials` 中。

- **HTTP Status:403 Error:RoleAccessError. You have no right to assume this role.**

  原因：faascmd沒有擷取到roleArn資訊，或者roleArn資訊與當前的AccessKeyID和AccessKeySecret資訊不屬於同一個賬戶。

  解決辦法：檢查 `/root/.faascredentials` 檔案是否包含以下資訊。

  ```
  [FaaSCredentials]
  accessid=xxxxxxxxxx
  accesskey=xxxxxxxxxxxxxxxxxxxxxxxx
  [Role]
  role=acs:ram::1234567890123456:role/xxxxxx
  [OSS]
  bucket=xxxx
  ```

  > 說明
  >
  > - 如果上述資訊存在，確認該role資訊與AccessKeyID/AccessKeySecret的雲ID是否一致。
  > - 如果上述資訊不存在，執行 `faascmd auth bucket=xxxx` 命令授權。

- **HTTP Status: 404 Error: EntityNotExist. Role Error. The specified Role not exists .**

  原因：您的雲賬戶下的faasrole角色不存在。

  解決辦法： 登陸RAM控制台查看faasrole角色是否存在。

  - 如果faasrole角色不存在，您需要執行 `faascmd config` 和 `faascmd auth` 命令建立該角色並為其授權。
  - 如果faasrole角色存在，請提交工單處理。

- **SDK.InvalidRegionId. Can not find endpoint to access.**

  原因：擷取不到faas服務的endpoint地址。

  解決辦法：您需要逐項檢查是否滿足以下配置。

  - 運行 `python -V` 命令檢查python版本是否為2.7.x。
  - 運行 `which python` 命令檢查python的預設安裝路徑是否為 `/usr/bin/python` 。
  - 運行 `cat /usr/lib/python2.7/site-packages/aliyunsdkcore/__init__.py` 命令檢查aliyunsdkcore版本是否為2.11.0及以上。

    > 說明 如果aliyunsdkcore版本號碼低於2.11.0，您需要運行 `pip install --upgrade aliyun-python-sdk-core` 命令升級至最新版本。

- **下載鏡像時返回 HTTP Status:404 Error:SHELL NOT MATCH. The image Shell is not match with fpga Shell!Request ID:D7D1AB1E-8682-4091-8129-C17D54FD10D4**

  原因：要下載的fpgaImage和指定fpga上的shell版本不匹配。

  解決辦法：您需要按下列步驟逐項檢查。

  - 運行 `faascmd list_instances --instance=xxx` 命令檢查當前fpga的shell版本號碼。
  - 運行 `faascmd list_images` 命令檢查指定的fpgaImage的shell版本號碼。

    > 說明
    > - 如果以上兩個shell版本號碼不同，您需要重新製作一個與fpga的shell版本號碼相同的fpgaImage，然後下載。
    > - 如果確定兩個shell版本一致，請提交工單。

- **下載鏡像時返回HTTP Status:503 Error:ANOTHER TASK RUNNING . Another task is running,user is allowed to take this task half an hour Request ID: 5FCB6F75-8572-4840-9BDC-87C57174F26D**

  原因：您之前提交的下載請求異常失敗或中斷導致fpga的狀態還停留在operating狀態。

  解決辦法：建議您等待10分鐘，直至下載任務自動結束，然後再次提交下載鏡像請求。

  > 說明 如果問題仍舊沒有解決，請提交工單。

- **運行faascmd list_images命令時，發現鏡像狀態是failed。**

  解決方案：您可以通過以下方式擷取編譯日誌，以定位相關錯誤。

```
faascmd list_objects|grep vivado
faascmd get_object --obejct=<yourObjectName> --file=<your_local_path>/vivado.log #路徑選填，預設下載到當前檔案夾。
```

常見錯誤碼

faascmd命令	API名字	錯誤資訊	錯誤描述	錯誤碼
適用所有命令	適用所有API	PARAMETER INVALIDATE	輸入參數有誤。	400
適用所有命令	適用所有API	InternalError	未知錯誤，提交工單。	500
auth	auth	NoPermisson	沒有訪問某個openAPI的許可權。	403
create_image	CreateFpgaImage	IMAGE NUMBER EXCEED	鏡像列表不能超過10個鏡像，刪除不需要的鏡像即可。	401
		FREQUENCY ERROR	目前提交鏡像請求的時間間隔為30min一次。	503
		SHELL NOT SUPPORT	輸入的shell版本不支援，請檢查shell版本是否正確。	404
		EntityNotExist.RoleError	使用者賬戶沒有建立faasRole。	404
		RoleAccessError	使用者輸入的roleArn為空白，或者roleArn資訊與AccessKey ID/AccessKey Secret不屬於同一個雲帳號。	403
		InvalidAccessKeyIdError	AccessKey ID/AccessKey Secret不合法。	401
		Forbidden.KeyNotFoundError	找不到指定的KMS key，請登陸KMS控制台檢查輸入的keyId是否存在。	503
		AccessDeniedError	faas admin 賬戶沒有訪問當前bucket的許可權。	
		OSS OBJECT NOT FOUND	指定的oss bucket/object不存在，或者不具備存取權限。	404
delete_image	DeleteFpgaImage	IMAGE NOT FOUND	指定的fpgaImage找不到。	400
		NOT AUTHORIZED	指定的instance不存在或者不屬於當前的雲賬戶。	401

faascmd命令	API名字	錯誤資訊	錯誤描述	錯誤碼
list_instances	DescribeFpgaInstances	RoleAccessError	使用者輸入的roleArn為空白，或者roleArn資訊與AccessKey ID/AccessKey Secret不屬於同一個雲帳號。	403
		INSTANCE INVALIDATE	指定的instance不屬於fpga執行個體。如果確定是fpga執行個體，請提交工單。	404
fpga_status	DescribeLoadTaskStatus	NOT AUTHORIZED	找不到指定的instanceId，請檢查輸入參數。	401
		FPGA NOT FOUND	找不到指定fpgauuid，請檢查輸入參數。	404
download_image	LoadFpgaImage	ANOTHER TASK RUNNING	之前提交的下載鏡像任務還在operating狀態。	503
		IMAGE ACCESS ERROR	指定的image不屬於當前雲賬戶。	401
		YOU HAVE NO ACCESS TO THIS INSTANCE	指定的instance不屬於當前的雲賬戶。	401
		IMAGE NOT FOUND	指定的fpgaImage找不到。	404
		FPGA NOT FOUND	指定的fpga找不到。	404
		SHELL NOT MATCH	鏡像的shell版本和指定的fpga上的shell版本不匹配。	404
		RoleAccessError	使用者輸入的roleArn為空白，或者roleArn資訊與AccessKey ID/AccessKey Secret不屬於同一個雲帳號。	403
		Image not in success state	指定的image不是success狀態，只有狀態為success的image才可以下載。	404
publish_image	PublishFpgaImage	FPGA IMAGE STATE ERROR	指定的image不是success狀態。	404
		FPGA IMAGE NOT FOUND	指定的image沒有找到或者不屬於目前使用者。	404

9.災備方案

保障企業業務穩定、IT系統功能正常、資料安全十分重要，可以同時保障資料備份與系統、應用容災的災備解決方案應勢而生，且發展迅速。ECS可使用快照、鏡像進行備份。

災備設計

- **快照備份**

  阿里雲ECS可使用快照進行系統硬碟、資料盤的備份。目前，阿里雲提供快照2.0服務，提供了更高的快照額度、更靈活的自動任務策略，並進一步降低了對業務I/O的影響。快照備份實行增量原理，第一次備份為全量備份，後續為增量備份。增量快照具有快速建立以及儲存容量小的優點。備份所需時間與待備份的增量資料體積有關。

  > **說明** 快照建立遵循增量原理，為了提高您的備份速度，建議您在建立完畢新快照後，再刪除最新的曆史快照。

  例如，快照1、快照2和快照3分別是磁碟的第一份、第二份和第三份快照。檔案系統對磁碟的資料進行分塊檢查，當建立快照時，只有變化了的資料區塊，才會被複製到快照中。阿里雲ECS的快照備份可配置為手動備份，也可配置為自動備份。配置為自動備份後可以指定磁碟自動建立快照的時間（24個整點）、重複日期（周一到周日）和保留時間（可自訂，範圍是1-65536天，或選擇持續保留）。

- **快照復原**

  當系統出現問題，需要將一塊磁碟的資料復原到之前的某一時刻，可以通過快照復原實現，前提是該磁碟已經建立了快照。注意：

  - 復原磁碟是無法復原操作，一旦復原完成，原有的資料將無法恢複，請謹慎操作。
  - 復原磁碟後，從所使用的快照的建立日期到目前時間這段時間內的資料都會丟失。

- **鏡像備份**

  鏡像檔案相當於副本檔案，該副本檔案包含了一塊或多塊磁碟中的所有資料，對於ECS而言，這些磁碟可以是單個系統硬碟，也可以是系統硬碟加資料盤的組合。使用鏡像備份時，均是全量備份，且只能手動觸發。

- **鏡像恢複**

  阿里雲ECS支援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。ECS的快照與鏡像配置請參考快照與鏡像。

  > **說明** 建立的自訂鏡像不能跨地區使用。

技術指標

RTO和RPO：與資料量大小有關，通常而言是小時層級。

應用情境

- **備份恢複**

阿里雲ECS可通過快照與鏡像對系統硬碟、資料盤進行備份。如果儲存在磁碟上的資料本身就是錯誤的資料，比如由於應用錯誤導致的資料錯誤，或者駭客利用應用漏洞進行惡意讀寫，此時就可以使用快照服務將磁碟上的資料恢複到期望的狀態。另外ECS可通過鏡像重新初始化磁碟或使用自訂鏡像新購ECS執行個體。

- **容災應用**

  ECS可以從架構上實現容災情境下的應用。例如，在應用前端購買SLB產品，後端相同應用部署至少兩台ECS伺服器，或者是使用阿里雲的彈性伸縮技術，根據自訂ECS自身資源的使用規則進行彈性擴容。這樣即便其中一台ECS伺服器故障或者資源利用超負荷，也不會使服務對外終止，從而實現容災情境下的應用。下圖以同城兩可用性區域機房部署ECS叢集為例，所有通訊均在阿里雲千兆內網中完成，響應快速並減少了公網流量費用：

  - Server Load Balancer：裝置側通過多可用性區域層級SLB做首層流量接入，使用者流量被分發至兩個及以上的可用性區域機房，機房內均部署ECS叢集。
  - ECS叢集：可用性區域機房部署的ECS節點是對等的，單節點故障不影響資料層應用和伺服器管控功能。發生故障後系統會自動熱遷移，另外的ECS節點可以持續提供業務訪問，防止可能的單點故障或者熱遷移失敗導致的業務訪問中斷。熱遷移失敗後通過系統事件獲知故障資訊，您可以及時部署新節點。
  - 資料層：在地區層級部署Object Storage Service，不同可用性區域機房的ECS節點可以直接讀取檔案資訊。若是資料庫應用，使用多可用性區域ApsaraDB for RDS服務做承載，主節點支援多可用性區域讀寫，與應用程式層流量來源無衝突關係。同時，備節點支援多可用性區域讀能力，防止主節點故障時，ECS無法讀取資料。