Pulse Connect Secure Virtual Appliance on Alibaba Cloud
Deployment Guide

Document Revision 1.0
Published Date January 2020
Pulse Secure, LLC
2700 Zanker Road,
Suite 200 San Jose
CA 95134

https://www.pulsesecure.net

Pulse Secure and the Pulse Secure logo are trademarks of Pulse Secure, LLC in the United States. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Pulse Secure, LLC assumes no responsibility for any inaccuracies in this document. Pulse Secure, LLC reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Products made or sold by Pulse Secure or components thereof might be covered by one or more of the following patents that are owned by or licensed to Pulse Secure: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

**END USER LICENSE AGREEMENT**

The Pulse Secure product that is the subject of this technical documentation consists of (or is intended for use with) Pulse Secure software. Use of such software is subject to the terms and conditions of the End User License Agreement (“EULA”) posted at https://support.pulsesecure.net/product-service-policies/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.
## Revision History

<table>
<thead>
<tr>
<th>Revision and Date</th>
<th>Added/Updated/Removed</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0, January 2020</td>
<td>NA</td>
<td>Initial publication for 9.1R4 release</td>
</tr>
</tbody>
</table>
Table of Contents

Revision History ........................................................................................................... 3
Overview ...................................................................................................................... 5
  About This Guide ........................................................................................................ 5
  Assumptions .............................................................................................................. 5
Pulse Connect Secure on Alibaba Cloud ........................................................................ 5
  Prerequisites and System Requirements on Alibaba Cloud ......................................... 5
  Deploying Pulse Connect Secure on Alibaba Cloud .................................................. 5
  Supported Platform Systems ..................................................................................... 6
Deploying Alibaba Cloud PCS using Alibaba Cloud Portal ........................................... 6
  Steps to Deploy Pulse Connect Secure on Alibaba Cloud .......................................... 6
Creating Alibaba Cloud PCS Image ............................................................................. 6
Creating Virtual Private Cloud .................................................................................... 11
Creating VSwitches ..................................................................................................... 13
Creating Security Groups .......................................................................................... 14
Creating PCS-VA Instance .......................................................................................... 16
Deploying Alibaba Cloud PCS using Terraform Template ........................................... 20
  Installing Terraform Template ................................................................................ 20
  Configuring Base Setup ........................................................................................... 20
  Deploying PCS with 2 NICs ...................................................................................... 21
  Deploying PCS with 3 NICs ...................................................................................... 21
Pulse Connect Secure Provisioning Parameters .......................................................... 21
Limitations .................................................................................................................. 23
Appendix A: Security Group (SG) .............................................................................. 24
Appendix B: Pulse Connect Secure Terraform Template ............................................. 27
  Base Setup .............................................................................................................. 27
  PCS with 2 NICs ..................................................................................................... 34
  PCS with 3 NICs ..................................................................................................... 37
  Variables ............................................................................................................... 40
  User Data .................................................................................................................. 41
References .................................................................................................................. 42
Requesting Technical Support ..................................................................................... 42
Overview

About This Guide

This guide helps in deploying the Pulse Connect Secure Virtual Appliance on Alibaba Cloud (Aliyun). In this release a Pulse Connect Secure administrator can manually upload the Pulse Connect Secure Virtual Appliance image (KVM) into Alibaba Cloud storage account. Once the package is available in the storage account, the Pulse Connect Secure administrator can deploy Pulse Connect Secure on Alibaba Cloud.

Assumptions

The basic understanding of deployment models of Pulse Connect Secure on a data center and basic experience in using Alibaba Cloud is needed for the better understanding of this guide.

Pulse Connect Secure on Alibaba Cloud

Prerequisites and System Requirements on Alibaba Cloud

To deploy the Pulse Connect Secure Virtual Appliance on Alibaba Cloud, you need the following:

- An Alibaba Cloud account
- Access to the Alibaba Cloud portal (https://account.alibabacloud.com/login/login.htm)*
- Pulse Connect Secure Virtual Appliance Image (file)
- Alibaba Cloud Terraform template
- Pulse Connect Secure (PSA-V) licenses
- Site-to-Site VPN between Alibaba Cloud and the corporate network (optional)

Note: This is needed only if the Pulse Connect Secure users need to access corporate resources

- Pulse Connect Secure configuration in XML format (optional)

Deploying Pulse Connect Secure on Alibaba Cloud

As depicted in the below diagram, a remote user can use Pulse Connect Secure to securely access cloud resources as well as corporate resources. To access corporate resources, the Pulse Connect Secure administrator needs to ensure that site-to-site VPN is already established between Alibaba Cloud and the corporate network.

Figure 1: Pulse Connect Secure on Alibaba Cloud
Supported Platform Systems

This section helps you in choosing the instance types that should be deployed with Pulse Connect Secure for Alibaba Cloud.

- PSA7000v is equivalent to ecs.g6.2xlarge in Beijing region.

<table>
<thead>
<tr>
<th>Model</th>
<th>Region</th>
<th>vCPU</th>
<th>Memory (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ecs.g6.2xlarge</td>
<td>Beijing</td>
<td>8</td>
<td>32</td>
</tr>
</tbody>
</table>

Deploying Alibaba Cloud PCS using Alibaba Cloud Portal

This section describes Alibaba Cloud PCS deployment with two NIC cards.

Steps to Deploy Pulse Connect Secure on Alibaba Cloud

- Creating Alibaba Cloud PCS Image
- Creating Virtual Private Cloud
- Creating Security Groups
- Creating PCS-VA Instance

Creating Alibaba Cloud PCS Image

To create Alibaba Cloud PCS image, do the following:

1. Download Alibaba Cloud PCS image, which is in the zip format, from the Pulse Secure Support site.
2. Unzip the file.
3. Log in to Alibaba Cloud with your account and credentials.

Figure 2: Alibaba Cloud Account Login Page
4. In the ECS Console displayed, select **Object Storage Service**. The Object Storage Service page allows OSS bucket management such as store and retrieve a variety of unstructured data files, including text files, images, audio files, and video files, over the network at any time.

**Figure 3: Object Storage Service Option**

5. Sign up for OSS and create one or more OSS buckets. For details, refer to [Alibaba Cloud Documentation](https://www.alibabacloud.com/).

**Figure 4: Bucket Management**

6. From the buckets list, click on the name of the created bucket.

7. In the window displayed, select the **Files** tab and then click **Create Folder** to create a folder for the Alibaba Cloud PCS-VA image.

**Figure 5: Create Folder**
8. After creating the folder, change to the newly created folder and then click **Upload**.
9. In the Upload section, drag and drop one or more Alibaba Cloud PCS-VA images.

**Figure 6: Upload Alibaba Cloud PCS VA Image**

10. Wait for the upload to complete.

**Figure 7: Upload Tasks**

11. Click on the **View Details** link corresponding to the uploaded file.

**Figure 8: View Details**
12. Make note of the URL of the image. You need to enter this URL when importing the image.

![Image URL](https://bucket-beijing-danumg.oss-cn-beijing.aliyuncs.com/pcs-va%20image/pcs_9.1R/v Alibaba Cloud)

**Figure 9: Image URL**

13. In the ECS Console, select Elastic Compute Service > Instances & Images > Images.

![Images Option](https://example.com)

**Figure 10: Images Option**

14. In the Images page displayed, select the region from the drop-down list located at the top-left corner of the page.

15. Click the Import image button located at the top-right corner of the page and select Manual Import.
16. In the Import Image page displayed, enter the following details and click **OK**.

**OSS Object Address**: Type the URL of the Alibaba Cloud PCS image that was uploaded to OSS bucket.
- **Image Name**: Type a unique name for the image.
- **Operating System**: Linux
- **System Disk Size (GB)**: 40
- **System Architecture**: Select an appropriate value from the drop-down list.
- **Platform**: CentOS
- **Image Format**: QCOW2

The imported image is listed in the Images page.

**Figure 13: Images List**

![Images List](image)

### Creating Virtual Private Cloud

To create Virtual Private Cloud (VPC), do the following:

1. Select Elastic Compute Service > Network & Security > VPC.

**Figure 14: VPC Option**

![VPC Option](image)
2. In the VPCs page, select the required region from the drop-down list located at the top-left corner of the page and click Create VPC.

**Figure 15: VPCs Page**

![Create VPC Page](image)

3. In the Create VPC page displayed, enter the VPC and VSwitch (Internal Port) details and click OK.

**Figure 16: Create VPC Page**

![Create VPC Page](image)
4. In the Create VPC - Details page, verify **Status** and click **Complete**. The created VPC is listed in the VPCs page.

**Figure 17: Create VPC – Details Page**

---

**Creating VSwitches**

While creating VPC, you created VSwitch for Internal Port. You need to create VSwitches for External and Management ports.

To create VSwitches, do the following:

1. Select **Elastic Compute Service > Network & Security > VPC > VSwitches**.
2. In the VSwitches page displayed, select the required region from the drop-down list located at the top-left corner of the page and click **Create VSwitch**.

**Figure 18: VSwitches Page**
3. In the Create VSwitch window displayed, do the following:
   a. Select VPC from the drop-down list.
   b. Enter a unique name for VSwitch for External port.
   c. Select Zone from the drop-down list.
   d. Click OK. The created VSwitch is listed in the VSwitches page.

![Create VSwitch Window](image)

4. Repeat the procedure to create VSwitch for Management Port.

Creating Security Groups

Security groups are groups of VPC instances that are located within the same region and share the same security requirements.

To create a security group, do the following:

2. In the Security Groups page, select the region from the drop-down list located at the top-right corner of the page and click Create Security Group.
3. In the Create Security Group window, select a template from the drop-down list.
4. Enter a name for the Security Group.
5. Select Security Group Type from the drop-down list.
6. Select Network Type as VPC.
7. Select VPC from the drop-down list.
8. Click OK.

Figure 21: Create Security Group Window
9. In the Notes dialog that is displayed, click Create Rules Now.

10. In the page that is displayed, click Add Security Group Rule to create the Inbound rules.

For details about Inbound rules, see Appendix A: Security Group (SG). This completes creation of Security Group.

Creating PCS-VA Instance

To create PCS-VA instance, do the following:

1. Select Elastic Compute Service > Instances and Images > Instances.
2. In the Instances page, select the region from the drop-down list located at the top-left corner of the page and click Create Instance.

For details about Inbound rules, see Appendix A: Security Group (SG). This completes creation of Security Group.

Creating PCS-VA Instance

To create PCS-VA instance, do the following:

1. Select Elastic Compute Service > Instances and Images > Instances.
2. In the Instances page, select the region from the drop-down list located at the top-left corner of the page and click Create Instance.

Figure 22: Instances Page – Create Instance Option

3. In the Basic Configurations page, select the Custom Launch tab.
4. Select the Billing Method as Pay-As-You-Go.
5. Select the appropriate zone in Region.
6. In Instance Type, type ecs.ic5.large.
   
   The Selected Instance Type displays the details of the instance type.
7. In the Image section, select the Custom Image tab.
8. From the drop-down list, select the required Alibaba Cloud PCS image that you want to deploy. For details about Alibaba Cloud PCS image, see “Creating Alibaba Cloud PCS Image”.

Figure 23: Basic Configurations Page

9. Click Next Networking to proceed to networking configuration.

10. In the Networking page, go to the Network Type section and select the required VPC and VSwitch from the drop-down lists. For details about creating VPC and VSwitch, see “Creating Virtual Private Cloud”.

11. In the Public IP Address section, select the Assign Public IP Address check box to select an IP address for the Internal Port.
12. In the Security Group section, click **Select Security Group**. In the Select Secure Group window displayed, select the security group assigned to the Internal Port and click **Save**. For details about creating Security Group, see “Creating Security Groups”.

**Figure 25: Select Security Group Window**

13. In the Elastic Network Interface section, click **Add ENI** and select the External Port.

**Figure 26: Elastic Network Interface Section**
14. Click **Next System Configuration** to proceed to system configuration.
15. In the System Configurations page, for Logon Credentials, select **Set Later**.
16. For **Instance Name**, enter the name of the virtual appliance.
17. In the User Data section, provide **Provisioning Parameters** in the XML format.

**Figure 27: System Configurations Page**

18. Click **Preview**. In the Summary page, accept **Terms of Service** and click **Create Instance**.

**Figure 28: Terms of Service**
Deploying Alibaba Cloud PCS using Terraform Template

This section describes how to install terraform template, deploy PCS with 2 NICs and 3 NICs.

Installing Terraform Template

1. Go to the Terraform website and install Terraform on a Linux VM of your choice.
2. Download the following directories and files in it:

<table>
<thead>
<tr>
<th>Directory</th>
<th>Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>base_setup</td>
<td>• base_setup.tf</td>
</tr>
<tr>
<td></td>
<td>• variables.tf</td>
</tr>
<tr>
<td>pcs_2_nics</td>
<td>• pcs_2_nics.tf</td>
</tr>
<tr>
<td></td>
<td>• user_data.txt</td>
</tr>
<tr>
<td></td>
<td>• variables.tf</td>
</tr>
<tr>
<td>pcs_3_nics</td>
<td>• pcs_3_nics.tf</td>
</tr>
<tr>
<td></td>
<td>• user_data.txt</td>
</tr>
<tr>
<td></td>
<td>• variables.tf</td>
</tr>
</tbody>
</table>

Configuring Base Setup

1. Customize and set the variables in variables.tf file.
2. Copy user_data.txt and variables.tf file to each of the directories (base_setup, pcs_2_nics, pcs_3_nics).
   Alternatively, create a softlink to these files.
3. Change directory to base_setup directory, and run the following commands:
   
   ```
   linux# terraform init
   linux# terraform apply
   ```
Deploying PCS with 2 NICs

1. Change directory to pcs_2_nics.
2. Customize the user_data.txt file. This file contains the PCS initial configuration data.
3. Run the following commands:
   ```
   linux# terraform init
   linux# terraform apply
   ```

Deploying PCS with 3 NICs

1. Change directory to pcs_3_nics.
2. Customize the user_data.txt file. This file contains the PCS initial configuration data.
3. Run the following commands:
   ```
   linux# terraform init
   linux# terraform apply
   ```

Pulse Connect Secure Provisioning Parameters

Provisioning parameters are those parameters which are required during the deployment of a virtual appliance. Pulse Connect Secure accepts the following parameters as provisioning parameters in the XML format.

```xml
<pulse-config>
  <wins-server><value></wins-server>
  <dns-domain><value></dns-domain>
  <admin-username><value></admin-username>
  <admin-password><value></admin-password>
  <cert-common-name><value></cert-common-name>
  <cert-random-text><value></cert-random-text>
  <cert-organisation><value></cert-organisation>
  <config-download-url><value></config-download-url>
  <config-data><value></config-data>
  <auth-code-license><value></auth-code-license>
  <enable-license-server><value></enable-license-server>
  <accept-license-agreement><value></accept-license-agreement>
  <enable-rest><value></enable-rest>
</pulse-config>
```
The below table depicts the details of the xml file.

<table>
<thead>
<tr>
<th>#</th>
<th>Parameter Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>wins-server</td>
<td>IP address</td>
<td>Wins server for Pulse Connect Secure</td>
</tr>
<tr>
<td>2</td>
<td>dns-domain</td>
<td>string</td>
<td>DNS domain of Pulse Connect Secure</td>
</tr>
<tr>
<td>3</td>
<td>admin-username</td>
<td>string</td>
<td>admin UI username</td>
</tr>
<tr>
<td>4</td>
<td>admin-password</td>
<td>string</td>
<td>admin UI password.</td>
</tr>
<tr>
<td>5</td>
<td>cert-common-name</td>
<td>string</td>
<td>Common name for the self-signed certificate generation. This certificate is used as the device certificate of Pulse Connect Secure.</td>
</tr>
<tr>
<td>6</td>
<td>cert-random-text</td>
<td>string</td>
<td>Random text for the self-certificate generation.</td>
</tr>
<tr>
<td>7</td>
<td>cert-organization</td>
<td>string</td>
<td>Organization name for the self-signed certificate generation.</td>
</tr>
<tr>
<td>8</td>
<td>config-download-url</td>
<td>String URL</td>
<td>Http based URL where XML based Pulse Connect Secure configuration can be found. During provisioning, Pulse Connect Secure fetches this file and comes up with preloaded configuration. XML based configuration can be present in another VM in Alibaba Cloud or at corporate network which is accessible for Pulse Connect Secure through site to site VPN between Alibaba Cloud and corporate data center.</td>
</tr>
<tr>
<td>9</td>
<td>config-data</td>
<td>string</td>
<td>base64 encoded XML based Pulse Connect Secure configuration</td>
</tr>
<tr>
<td>10</td>
<td>auth-code-license</td>
<td>string</td>
<td>Authentication code that needs to be obtained from Pulse Secure.</td>
</tr>
<tr>
<td>11</td>
<td>enable-license-server</td>
<td>string</td>
<td>If set to 'y', PCS will be deployed as a License server. If set to 'n', PCS will be deployed as a normal server.</td>
</tr>
<tr>
<td>12</td>
<td>accept-license-agreement</td>
<td>string</td>
<td>This value is passed to the instance for configuration at the boot time. By default, this value is set to &quot;n&quot;. <strong>This value must be set to 'y'.</strong></td>
</tr>
<tr>
<td>13</td>
<td>enable-rest</td>
<td>string</td>
<td>If set to 'y', REST API access for the administrator user is enabled.</td>
</tr>
</tbody>
</table>

**Note:**
- In the above list of parameters, **dns domain, admin username, admin password, cert-random name, cert-random text, cert-organization and accept-license-agreement** are mandatory parameters. The other parameters are optional parameters.
- From 9.1R3 release, Pulse Connect Secure supports zero touch provisioning. This feature can detect and assign DHCP networking settings automatically at the Pulse Connect Secure boot up. The Pulse Connect Secure parameters should be set to null in order to fetch the networking configuration automatically from the DHCP server.
Limitations

The following list of Pulse Connect Secure features are not supported in this release:

- Default VLAN on Internal, External and Management Ports
- VLAN functionality
- AP Cluster
- Layer 3 Tunnel IP pool assignment via DHCP
- Virtual Ports
Appendix A: Security Group (SG)

Alibaba Cloud has a limitation where virtual machine with multiple network interfaces cannot connect to different Virtual Private Cloud (VPCs). For example, a VM with two NICs, NIC1 and NIC2, will not be able to connect to VPC1 and VPC2 respectively.

**Figure 29: Virtual Machine with two NICs Connecting to VPC1 and VPC2**

Alibaba Cloud supports a virtual machine with multiple NICs to connect to different Subnets under a same Virtual Private Cloud. For example, a VM with two NICs, NIC1 and NIC2, can connect to ‘Subnet1’ and ‘Subnet2’ where these subnets exist under a same Virtual Private Cloud respectively.

**Figure 30: Virtual Machine with two NICs Connecting to Subnet1 and Subnet2**

Alibaba Cloud provides isolation between different VPCs. But it does not provide the same kind of isolation when it comes to subnets in the same VPC. For example, consider a VPC has two subnets, Subnet1 and Subnet2. And consider two VMs, VM-1 and VM-2, which are connected to Subnet1 and Subnet2 respectively. In this scenario VM-1 can access the resources from VM-2 and vice versa.

**Figure 31: Virtual Machine VM-1 can Access Resources in VM-2 and Vice Versa**

Application isolation is an important concern in enterprise environments, as enterprise customers seek to protect various environments from unauthorized or unwanted access. To achieve the traffic isolation between subnets, go for an option of filtering traffic using “Security Group” provided by Alibaba Cloud.
Pulse Connect Secure, when provisioned through the Terraform template provided by Pulse Secure, creates three subnets under a virtual private cloud named “PCSVirtualNetwork”. The three Subnets are:

1. vsw-zone-a-pcs-int-port-subnet
2. vsw-zone-a-pcs-ext-port-subnet
3. vsw-zone-a-pcs-mgmt-port-subnet

Along with above mentioned subnets, create the following three Security Groups (SG) policies:

1. sg_pcs_int_port
2. sg_pcs_ext_port
3. sg_pcs_mgmt_port

**Figure 32: Traffic Filtering by Alibaba Cloud Support Group**

**Figure 33: SG External, Internal and Management Subnets**
In Security Group (SG) we need to create policies for Inbound traffic.

1. Select **Elastic Compute Service** > **Network & Security** > **Security Groups**.
2. The list of SG Inbound rules created “sg_pcs_ext_port” are:

   **Figure 34: sg_pcs_ext_port - Inbound Rules**

   ![sg_pcs_ext_port - Inbound Rules](image)

   - **Inbound**
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 80/80
     - **Authorization Type**: IPv4 CDR
     - **Description**: HTTP port 80
   - **Outbound**
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 443/443
     - **Authorization Type**: IPv4 CDR
     - **Description**: HTTPS port 443
   - **Inbound**
     - **Action**: Allow
     - **Protocol Type**: Customized UDP
     - **Port Range**: 4500/4500
     - **Authorization Type**: IPv4 CDR
     - **Description**: UDP Ports for Pulse L3
   - **Outbound**
     - **Action**: Allow
     - **Protocol Type**: All ICMP (IPv4)
     - **Port Range**: -1/-1
     - **Authorization Type**: IPv4 CDR
     - **Description**: Allow All ICMP

3. The list of SG Inbound rules created “sg_pcs_int_port” are:

   **Figure 35: sg_pcs_int_port - Inbound Rules**

   ![sg_pcs_int_port - Inbound Rules](image)

   - **Inbound**
     - **Action**: Allow
     - **Protocol Type**: All ICMP (IPv4)
     - **Port Range**: -1/-1
     - **Authorization Type**: IPv4 CDR
     - **Description**: Allow All ICMP
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 4808/4809
     - **Authorization Type**: IPv4 CDR
     - **Description**: TCP Ports 4808 and 4809
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 4900/4900
     - **Authorization Type**: IPv4 CDR
     - **Description**: TCP Port 4900 - 4910
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 5000/5000
     - **Authorization Type**: IPv4 CDR
     - **Description**: TCP Port for Cluster
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 11000/11000
     - **Authorization Type**: IPv4 CDR
     - **Description**: TCP Port for Cluster
     - **Action**: Allow
     - **Protocol Type**: Customized UDP
     - **Port Range**: 4500/4500
     - **Authorization Type**: IPv4 CDR
     - **Description**: UDP Ports for Pulse L3
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 8180/8180
     - **Authorization Type**: IPv4 CDR
     - **Description**: DMI NacF Port
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 4803/4803
     - **Authorization Type**: IPv4 CDR
     - **Description**: UDP Ports for Cluster
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 443/443
     - **Authorization Type**: IPv4 CDR
     - **Description**: HTTPS port 443

4. The list of SG Inbound rules created “sg_pcs_mgmt_port” are:

   **Figure 36: sg_pcs_mgmt_port - Inbound Rules**

   ![sg_pcs_mgmt_port - Inbound Rules](image)

   - **Inbound**
     - **Action**: Allow
     - **Protocol Type**: All ICMP (IPv4)
     - **Port Range**: -1/-1
     - **Authorization Type**: IPv4 CDR
     - **Description**: Allow All ICMP
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 443/443
     - **Authorization Type**: IPv4 CDR
     - **Description**: HTTPS port 443
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 8180/8180
     - **Authorization Type**: IPv4 CDR
     - **Description**: DMI Netcom Port 8180
     - **Action**: Allow
     - **Protocol Type**: Custom TCP
     - **Port Range**: 80/80
     - **Authorization Type**: IPv4 CDR
     - **Description**: HTTP port 80
Appendix B: Pulse Connect Secure Terraform Template

Terraform is an open source tool to easily define, preview, and deploy cloud infrastructure on Alibaba Cloud. Pulse Secure provides sample Terraform template files for 2 NICs and 3 NICs to deploy the Pulse Connect Secure Virtual Appliance on Alibaba Cloud. Users can modify this to make it suitable for their need. Visit [https://www.pulsesecure.net](https://www.pulsesecure.net) and download the template file.

- Base Setup
- PCS with 2 NICs
- PCS with 3 NICs
- Variables
- User Data

Base Setup

```terraform
# Terraform version
terraform {
    required_version = ">= 0.12.18"
}

# Access Keys
provider "alicloud" {
    access_key = var.access_key
    secret_key = var.secret_key
    # region is important
    region = var.region
}

# Bucket Creation
terraform {
    backend "oss" {
        bucket = "bucket-beijing-darumuga"
        prefix = "terraform_state/base_setup_beijing"
        key = "terraform.tfstate"
        region = "cn-beijing"
        tablestore_endpoint = "https://oss-cn-beijing.aliyuncs.com"
    }
}

# VPCs, VSwitches, Security Groups and Alibaba Cloud PCS VA Images

# VPCs
data "alicloud_vpcs" "vpcs_ds" {
    # No args required
}

# VSwitches
data "alicloud_vswitches" "vswitches_ds" {
    vpc_id = "${local.vpc_id}"
}

# Available security groups
data "alicloud_security_groups" "sec_groups_ds" {
    vpc_id = "${local.vpc_id}"
}

# Available images loaded by the user
data "alicloud_images" "self_images_ds" {
    owners = "self" # system | marketplace | others | self
}
```
# Local variables
locals {
    vpcs_list           = "${data.alicloud_vpcs.vpcs_ds.vpcs}"  
    vpc_id              = join ("", [ for vpc in local.vpcs_list : vpc.id if vpc.vpc_name == "${var.vpc}" ] )  
    vsws_list_in_vpc    = "${data.alicloud_vswitches.vswitches_ds.vswitches}"  
    sec_groups_list     = "${data.alicloud_security_groups.sec_groups_ds.groups}"  
    images_list         = {
        "self"   = "${data.alicloud_images.self_images_ds.images}",
    }
    instance_type     = var.instance_type  
    region            = var.region  
    zone              = var.zone  
    vswitch_names_map   = {
        "${var.zone_a}" = {
            "pcs_int_port" = "vsw-zone-a-pcs-int-port-subnet",
            "pcs_ext_port" = "vsw-zone-a-pcs-ext-port-subnet",
            "pcs_mgmt_port" = "vsw-zone-a-pcs-mgmt-port-subnet",
        },
        "${var.zone_b}" = {
            "pcs_int_port" = "vsw-zone-b-pcs-int-port-subnet",
            "pcs_ext_port" = "vsw-zone-b-pcs-ext-port-subnet",
            "pcs_mgmt_port" = "vsw-zone-b-pcs-mgmt-port-subnet",
        },
    }
    security_group_names_map = {
        "pcs_int_port"      = "sg_pcs_int_port",
        "pcs_ext_port"      = "sg_pcs_ext_port",
        "pcs_mgmt_port"     = "sg_pcs_mgmt_port",
    }
}

# Create a VPC
resource "alicloud_vpc" "vpc" {
    name       = "vpc_beijing_darumuga"
    cidr_block = "172.16.5.0/24"
}

# Create vSwitch for zone-a

# for pcs internal port
resource "alicloud_vswitch" "vsw-zone-a-pcs-int-port-subnet" {
    vpc_id  = alicloud_vpc.vpc.id  
    name    = local.vswitch_names_map[var.zone_a]["pcs_int_port"]  
    cidr_block = "172.16.5.16/28"  
    availability_zone = var.zone_a
}

# for pcs external port
resource "alicloud_vswitch" "vsw-zone-a-pcs-ext-port-subnet" {
    vpc_id  = alicloud_vpc.vpc.id  
    name    = local.vswitch_names_map[var.zone_a]["pcs_ext_port"]  
    cidr_block = "172.16.5.32/28"  
    availability_zone = var.zone_a
}

# for pcs management port
resource "alicloud_vswitch" "vsw-zone-a-pcs-mgmt-port-subnet" {
    vpc_id  = alicloud_vpc.vpc.id  
    name    = local.vswitch_names_map[var.zone_a]["pcs_mgmt_port"]
}
cidr_block = "172.16.5.48/28"
availability_zone = var.zone_a
}

#Create vSwitch for zone-b
#for pcs internal port
resource "alicloud_vswitch" "vsw-zone-b-pcs-int-port-subnet" {
  vpc_id = alicloud_vpc.vpc.id
  name   = local.vswitch_names_map[var.zone_b]["pcs_int_port"]
  cidr_block = "172.16.5.64/28"
  availability_zone = var.zone_b
}
#for pcs external port
resource "alicloud_vswitch" "vsw-zone-b-pcs-ext-port-subnet" {
  vpc_id = alicloud_vpc.vpc.id
  name   = local.vswitch_names_map[var.zone_b]["pcs_ext_port"]
  cidr_block = "172.16.5.80/28"
  availability_zone = var.zone_b
}
#for pcs management port
resource "alicloud_vswitch" "vsw-zone-b-pcs-mgmt-port-subnet" {
  vpc_id = alicloud_vpc.vpc.id
  name   = local.vswitch_names_map[var.zone_b]["pcs_mgmt_port"]
  cidr_block = "172.16.5.96/28"
  availability_zone = var.zone_b
}

#Create Security Groups
#create security group for pcs internal port
resource "alicloud_security_group" "sg_pcs_int_port" {
  name   = local.security_group_names_map["pcs_int_port"]
  description = "Security Group for PCS internal port"
  vpc_id = alicloud_vpc.vpc.id
}
#create security group for pcs external port
resource "alicloud_security_group" "sg_pcs_ext_port" {
  name   = local.security_group_names_map["pcs_ext_port"]
  description = "Security Group for PCS external port"
  vpc_id = alicloud_vpc.vpc.id
}
#create security group for pcs management port
resource "alicloud_security_group" "sg_pcs_mgmt_port" {
  name   = local.security_group_names_map["pcs_mgmt_port"]
  description = "Security Group for PCS management port"
  vpc_id = alicloud_vpc.vpc.id
}
#create security group for backend servers
resource "alicloud_security_group" "sg_backend_svr" {
  name   = "sg_backend_svr"
  description = "Security Group for backend servers in protected network"
  vpc_id = alicloud_vpc.vpc.id
}

#Create Security Group Rules for PCS Internal Port and Assign to Security Group
#HTTP port 80
resource "alicloud_security_group_rule" "int_port_allow_tcp_80" {
  description = "HTTP port 80"
  type = "ingress"
Pulse Connect Secure Virtual Appliance on Alibaba Cloud

#HTTPS port 443
resource "alicloud_security_group_rule" "int_port_allow_tcp_443" {
  description       = "HTTPS port 443"
  type              = "ingress"
  ip_protocol       = "tcp"
  nic_type          = "intranet"
  policy            = "accept"
  port_range        = "443/443"
  priority          = 1
  security_group_id = alicloud_security_group.sg_pcs_int_port.id
  cidr_ip           = "0.0.0.0/0"
}

#DMI Netconf port
resource "alicloud_security_group_rule" "int_port_allow_tcp_dmi_830" {
  description       = "DMI Netconf port"
  type              = "ingress"
  ip_protocol       = "tcp"
  nic_type          = "intranet"
  policy            = "accept"
  port_range        = "830/830"
  priority          = 1
  security_group_id = alicloud_security_group.sg_pcs_int_port.id
  cidr_ip           = "0.0.0.0/0"
}

#Allow All ICMP
resource "alicloud_security_group_rule" "int_port_allow_all_icmp" {
  description       = "Allow All ICMP"
  type              = "ingress"
  ip_protocol       = "icmp"
  nic_type          = "intranet"
  policy            = "accept"
  port_range        = "-1/-1"
  priority          = 1
  security_group_id = alicloud_security_group.sg_pcs_int_port.id
  cidr_ip           = "0.0.0.0/0"
}

#TCP Ports for Cluster Communication
resource "alicloud_security_group_rule" "int_port_allow_tcp_cluster_comms" {
  description       = "TCP Ports for Cluster Communication"
  type              = "ingress"
  ip_protocol       = "tcp"
  nic_type          = "intranet"
  policy            = "accept"
  port_range        = "11000/11099"
  priority          = 1
  security_group_id = alicloud_security_group.sg_pcs_int_port.id
  cidr_ip           = "0.0.0.0/0"
}

#TCP Ports 4808 and 4809 for Cluster Communication
resource "alicloud_security_group_rule" "int_port_allow_tcp_cluster_comms_4808_4809" {
    description = "TCP Ports 4808 and 4809 for Cluster Communication"
    type        = "ingress"
    ip_protocol = "tcp"
    nic_type    = "intranet"
    policy      = "accept"
    port_range  = "4808/4809"
    priority    = 1
    security_group_id = alicloud_security_group.sg_pcs_int_port.id
    cidr_ip     = "0.0.0.0/0"
}

#TCP Ports 4900 - 4910 for Cluster Key Exchange and State Sync
resource "alicloud_security_group_rule" "int_port_allow_tcp_cluster_key_exchange_and_sync" {
    description = "TCP Ports 4900 - 4910 for Cluster Key Exchange and State Sync"
    type        = "ingress"
    ip_protocol = "tcp"
    nic_type    = "intranet"
    policy      = "accept"
    port_range  = "4900/4910"
    priority    = 1
    security_group_id = alicloud_security_group.sg_pcs_int_port.id
    cidr_ip     = "0.0.0.0/0"
}

#UDP Ports for Cluster Communication
resource "alicloud_security_group_rule" "int_port_allow_udp_cluster_comms" {
    description = "UDP Ports for Cluster Communication"
    type        = "ingress"
    ip_protocol = "tcp"
    nic_type    = "intranet"
    policy      = "accept"
    port_range  = "4803/4803"
    priority    = 1
    security_group_id = alicloud_security_group.sg_pcs_int_port.id
    cidr_ip     = "0.0.0.0/0"
}

#UDP Ports for Cluster HeartBeat
resource "alicloud_security_group_rule" "int_port_allow_udp_cluster_hearbeat" {
    description = "UDP Ports for Cluster HeartBeat"
    type        = "ingress"
    ip_protocol = "tcp"
    nic_type    = "intranet"
    policy      = "accept"
    port_range  = "4804/4804"
    priority    = 1
    security_group_id = alicloud_security_group.sg_pcs_int_port.id
    cidr_ip     = "0.0.0.0/0"
}

#UDP Ports for Pulse L3 Connection
resource "alicloud_security_group_rule" "int_port_allow_udp_pulse_client" {
    description = "UDP Ports for Pulse L3 Connection"
    type        = "ingress"
    ip_protocol = "udp"
    nic_type    = "intranet"
    policy      = "accept"
    port_range  = "4500/4500"
    priority    = 1
    security_group_id = alicloud_security_group.sg_pcs_int_port.id
}
cidr_ip          = "0.0.0.0/0"
}

Create Security Group Rules for PCS External Port and Assign to Security Group

#HTTP port 80
resource "alicloud_security_group_rule" "ext_port_allow_tcp_80" {
  description = "HTTP port 80"
  type          = "ingress"
  ip_protocol   = "tcp"
  nic_type      = "intranet"
  policy        = "accept"
  port_range    = "80/80"
  priority      = 1
  security_group_id = alicloud_security_group.sg_pcs_ext_port.id
  cidr_ip       = "0.0.0.0/0"
}

#HTTPS port 443
resource "alicloud_security_group_rule" "ext_port_allow_tcp_443" {
  description = "HTTPS port 443"
  type          = "ingress"
  ip_protocol   = "tcp"
  nic_type      = "intranet"
  policy        = "accept"
  port_range    = "443/443"
  priority      = 1
  security_group_id = alicloud_security_group.sg_pcs_ext_port.id
  cidr_ip       = "0.0.0.0/0"
}

#Allow All ICMP
resource "alicloud_security_group_rule" "ext_port_allow_all_icmp" {
  description = "Allow All ICMP"
  type          = "ingress"
  ip_protocol   = "icmp"
  nic_type      = "intranet"
  policy        = "accept"
  port_range    = ":-1/-1"
  priority      = 1
  security_group_id = alicloud_security_group.sg_pcs_ext_port.id
  cidr_ip       = "0.0.0.0/0"
}

#UDP Ports for Pulse L3 Connection
resource "alicloud_security_group_rule" "ext_port_allow_udp_pulse_client" {
  description = "UDP Ports for Pulse L3 Connection"
  type          = "ingress"
  ip_protocol   = "udp"
  nic_type      = "intranet"
  policy        = "accept"
  port_range    = "4500/4500"
  priority      = 1
  security_group_id = alicloud_security_group.sg_pcs_ext_port.id
  cidr_ip       = "0.0.0.0/0"
}

Create Security Group Rules for PCS Management Port and Assign to Security Group

#HTTP port 80
resource "alicloud_security_group_rule" "mgmt_port_allow_tcp_80" {
  description = "HTTP port 80"
  type          = "ingress"
  ip_protocol   = "tcp"
  nic_type      = "intranet"
  policy        = "accept"
  port_range    = "80/80"
  priority      = 1
  security_group_id = alicloud_security_group.sg_pcs_ext_port.id
  cidr_ip       = "0.0.0.0/0"
}
#HTTP port 80
resource "alicloud_security_group_rule" "backend_svr_allow_tcp_80" {
  description = "HTTP port 80"
  type = "ingress"
  ip_protocol = "tcp"
  nic_type = "intranet"
  policy = "accept"
  port_range = "80/80"
  priority = 1
  security_group_id = alicloud_security_group.sg_backend_svr.id
  cidr_ip = "0.0.0.0/0"
}

#HTTPS port 443
resource "alicloud_security_group_rule" "mgmt_port_allow_tcp_443" {
  description = "HTTPS port 443"
  type = "ingress"
  ip_protocol = "tcp"
  nic_type = "intranet"
  policy = "accept"
  port_range = "443/443"
  priority = 1
  security_group_id = alicloud_security_group.sg_pcs_mgmt_port.id
  cidr_ip = "0.0.0.0/0"
}

#DMI Netconf port 830
resource "alicloud_security_group_rule" "mgmt_port_allow_tcp_dmi_830" {
  description = "DMI Netconf port 830"
  type = "ingress"
  ip_protocol = "tcp"
  nic_type = "intranet"
  policy = "accept"
  port_range = "830/830"
  priority = 1
  security_group_id = alicloud_security_group.sg_pcs_mgmt_port.id
  cidr_ip = "0.0.0.0/0"
}

#Allow All ICMP
resource "alicloud_security_group_rule" "mgmt_port_allow_all_icmp" {
  description = "Allow All ICMP"
  type = "ingress"
  ip_protocol = "icmp"
  nic_type = "intranet"
  policy = "accept"
  port_range = "./.
  priority = 1
  security_group_id = alicloud_security_group.sg_pcs_mgmt_port.id
  cidr_ip = "0.0.0.0/0"
}

#Create Security Group Rules for Backend Servers and Assign to Security Group
#HTTP port 80
resource "alicloud_security_group_rule" "backend_svr_allow_tcp_80" {
  description = "HTTP port 80"
  type = "ingress"
  ip_protocol = "tcp"
  nic_type = "intranet"
  policy = "accept"
  port_range = "80/80"
  priority = 1
  security_group_id = alicloud_security_group.sg_backend_svr.id
  cidr_ip = "0.0.0.0/0"
}
#HTTPS port 443
resource "alicloud_security_group_rule" "backend_svr_allow_tcp_443" {  
description = "HTTPS port 443"  
type = "ingress"  
ip_protocol = "tcp"  
nic_type = "intranet"  
policy = "accept"  
port_range = "443/443"  
priority = 1  
security_group_id = alicloud_security_group.sg_backend_svr.id  
cidr_ip = "0.0.0.0/0"
}

#SSH port 22
resource "alicloud_security_group_rule" "backend_svr_allow_tcp_ssh_22" {  
description = "SSH port 22"  
type = "ingress"  
ip_protocol = "tcp"  
nic_type = "intranet"  
policy = "accept"  
port_range = "22/22"  
priority = 1  
security_group_id = alicloud_security_group.sg_backend_svr.id  
cidr_ip = "0.0.0.0/0"
}

#Allow All ICMP
resource "alicloud_security_group_rule" "backend_svr_allow_all_icmp" {  
description = "Allow All ICMP"  
type = "ingress"  
ip_protocol = "icmp"  
nic_type = "intranet"  
policy = "accept"  
port_range = ".-1/-1"  
priority = 1  
security_group_id = alicloud_security_group.sg_backend_svr.id  
cidr_ip = "0.0.0.0/0"
}

**PCS with 2 NICs**

#Terraform version
terraform {  
  required_version = ">= 0.12.18"
}

#Access Keys
provider "alicloud" {  
  access_key = var.access_key  
  secret_key = var.secret_key  
  #region is important  
  region = var.region
}

#VPCs, VSwitches, Security Groups and Alibaba Cloud PCS VA Images
#VPCs
data "alicloud_vpcs" "vpcs_ds" {
  #No args required
}

data "alicloud_vswitches" "vswitches_ds" {
  vpc_id = "${local.vpc_id}"
}

data "alicloud_security_groups" "sec_groups_ds" {
  vpc_id = "${local.vpc_id}"
}

data "alicloud_images" "self_images_ds" {
  owners = "self" #system|marketplace|others|self
}

locals {
  vpcs_list = "${data.alicloud_vpcs.vpcs_ds.vpcs}"%
  vpc_id = join ("", [ for vpc in local.vpcs_list : vpc.id if vpc.vpc_name == "${var.vpc}" ])
  vsws_list_in_vpc = "${data.alicloud_vswitches.vswitches_ds.vswitches}"
  sec_groups_list = "${data.alicloud_security_groups.sec_groups_ds.groups}"
  images_list = {
    "self" = "${data.alicloud_images.self_images_ds.images}"
  }
  instance_type = var.instance_type
  region = var.region
  zone = var.zone
  security_group_id_map = {
    "pcs_int_port" = join ("", [ for sec_group in local.sec_groups_list : sec_group.id if sec_group.name == "sg_pcs_int_port" ])
    "pcs_ext_port" = join ("", [ for sec_group in local.sec_groups_list : sec_group.id if sec_group.name == "sg_pcs_ext_port" ])
    "pcs_mgmt_port" = join ("", [ for sec_group in local.sec_groups_list : sec_group.id if sec_group.name == "sg_pcs_mgmt_port" ])
  }
  vswitch_id_map = {
    "$\{\text{var.zone} \_a}\" = {
      "pcs_int_port" = join ("", [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-a-pcs-int-port-subnet" ])
      "pcs_ext_port" = join ("", [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-a-pcs-ext-port-subnet" ])
      "pcs_mgmt_port" = join ("", [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-a-pcs-mgmt-port-subnet" ])
    },
    "$\{\text{var.zone} \_b}\" = {
      "pcs_int_port" = join ("", [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-b-pcs-int-port-subnet" ])
      "pcs_ext_port" = join ("", [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-b-pcs-ext-port-subnet" ])
      "pcs_mgmt_port" = join ("", [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-b-pcs-mgmt-port-subnet" ])
    }
  }
}

#PCS instance on Alibaba Cloud
resource "alicloud_instance" "pcs_instance" {
instance_name = var.instance_name
image_id = join ("", [ for pcs_image in local.images_list[var.image_from] : pcs_image.image_id if pcs_image.name == var.image_name ])
availability_zone = var.zone
#instance_type = var.instance_type #ecs.hfc5.large
instance_type = "ecs.n1.large" #customise according to your needs
instance_charge_type = var.instance_charge_type #PayByBandwidth
system_disk_category = "cloud_efficiency"
vswitch_id = local.vswitch_id_map[local.zone]['pcs_int_port']
security_groups = [ local.security_group_id_map['pcs_int_port'] ]

#internet_max_bandwidth_out = 1 #in mbps, when this value is set, a public IP(not elastic IP) is assigned
#internet_charge_type = "PayByBandwidth"

#user-data
user_data = file("user_data.txt")
}

#Create External Port
resource "alicloud_network_interface" "pcs_ext_port" {
  name = "pcs_ext_port"
  vswitch_id = local.vswitch_id_map[local.zone]['pcs_ext_port']
  security_groups = [ local.security_group_id_map['pcs_ext_port'] ]
}
resource "alicloud_network_interface_attachment" "pcs_ext_port_attachment" {
  instance_id = alicloud_instance.pcs_instance.id
  network_interface_id = alicloud_network_interface.pcs_ext_port.id
}

#Create a new EIP
# Create a new EIP for Internal Port.
resource "alicloud_eip" "pcs_int_port_eip" {
  bandwidth = "5"
  internet_charge_type = "PayByTraffic"
}
resource "alicloud_eip_association" "pcs_int_port_eip_asso" {
  allocation_id = alicloud_eip.pcs_int_port_eip.id
  instance_id = alicloud_instance.pcs_instance.id
}
# Create a new EIP for External Port.
resource "alicloud_eip" "pcs_ext_port_eip" {
  bandwidth = "5"
  internet_charge_type = "PayByTraffic"
}
resource "alicloud_eip_association" "pcs_ext_port_eip_asso" {
  allocation_id = alicloud_eip.pcs_ext_port_eip.id
  instance_id = alicloud_network_interface.pcs_ext_port.id #if instance_type is NetworkInterface, instance_id should point to the ENI ID(not VM instance ID)
  instance_type = "NetworkInterface" #for assigning EIP to ext port(which is secondary ENI)
  private_ip_address = alicloud_network_interface.pcs_ext_port.private_ip
}

#Output
output "pcs_int_port" {
  value = "${{alicloud_instance.pcs_instance.private_ip}}"
}
output "pcs_ext_port" {
  value = "${{alicloud_network_interface.pcs_ext_port.private_ip}}"
PCS with 3 NICs

```hcl
# Terraform version
terraform {
  required_version = ">= 0.12.18"
}

# Access Keys
provider "alicloud" {
  access_key = var.access_key
  secret_key = var.secret_key
  # region is important
  region = var.region
}

# VPCs, VSwitches, Security Groups and Alibaba Cloud PCS VA Images

# VPCs
data "alicloud_vpcs" "vpcs_ds" {
  # No args required
}

# VSwitches
data "alicloud_vswitches" "vswitches_ds" {
  vpc_id = "${local.vpc_id}""n
}

# Available security groups
data "alicloud_security_groups" "sec_groups_ds" {
  vpc_id = "${local.vpc_id}""n
}

# Available images loaded by the user
data "alicloud_images" "self_images_ds" {
  owners = "self" # system | marketplace | others | self
}

# Local variables
locals {
  vpcs_list = "$\{data.alicloud_vpcs.vpcs_ds.vpcs\}$"
  vpc_id = join ("", [ for vpc in local.vpcs_list : vpc.id if vpc.vpc_name == "$\{var.vpc\}$ ])
  vsws_list_in_vpc = "$\{data.alicloud_vswitches.vswitches_ds.vswitches\}$"
  sec_groups_list = "$\{data.alicloud_security_groups.sec_groups_ds.groups\}$"
  images_list = {
    "self" = "$\{data.alicloud_images.self_images_ds.images\}$"
  }
  instance_type = var.instance_type
  region = var.region
```
zone = var.zone
security_group_id_map = {
    "pcs_int_port"  = join ('', [ for sec_group in local.sec_groups_list : sec_group.id if sec_group.name == "sg_pcs_int_port" ]),
    "pcs_ext_port"  = join ('', [ for sec_group in local.sec_groups_list : sec_group.id if sec_group.name == "sg_pcs_ext_port" ]),
    "pcs_mgmt_port" = join ('', [ for sec_group in local.sec_groups_list : sec_group.id if sec_group.name == "sg_pcs_mgmt_port" ]),
}

vswitch_id_map = {
    "${var.zone_a}"  = {
        "pcs_int_port"  = join ('', [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-a-pcs-int-port-subnet" ]),
        "pcs_ext_port"  = join ('', [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-a-pcs-ext-port-subnet" ]),
        "pcs_mgmt_port" = join ('', [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-a-pcs-mgmt-port-subnet" ]),
    },
    "${var.zone_b}"  = {
        "pcs_int_port"  = join ('', [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-b-pcs-int-port-subnet" ]),
        "pcs_ext_port"  = join ('', [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-b-pcs-ext-port-subnet" ]),
        "pcs_mgmt_port" = join ('', [ for vswitch in local.vsws_list_in_vpc : vswitch.id if vswitch.name == "vsw-zone-b-pcs-mgmt-port-subnet" ]),
    }
}

#PCS Instance on Alibaba Cloud
resource "alicloud_instance" "pcs_instance" {
    instance_name     = var.instance_name
    image_id                = join ('', [ for pcs_image in local.images_list[var.image_from] : pcs_image.image_id if pcs_image.name == var.image_name ])
    availability_zone      = var.zone
    #instance_type        = var.instance_type #ecs.hfc5.large
    instance_type           = "ecs.n1.xlarge" #customise according to your needs
    instance_charge_type    = var.instance_charge_type #PayByBandwidth
    system_disk_category    = "cloud_efficiency"
    vswitch_id             = local.vswitch_id_map[var.zone]["pcs_int_port"]
    security_groups        = [ local.security_group_id_map["pcs_int_port"] ]
    
    #internet_max_bandwidth_out  = 1 #in mbps, when this value is set, a public IP(not elastic IP) is assigned
    #internet_charge_type     = "PayByBandwidth"
    
    #user-data
    user_data = file("user_data.txt")
}

#Create External Port and Management Port
#create external port and attach to PCS instance
resource "alicloud_network_interface" "pcs_ext_port" {
    name      = "pcs_ext_port"
    vswitch_id           = local.vswitch_id_map[var.zone]["pcs_ext_port"]
    security_groups     = [ local.security_group_id_map["pcs_ext_port"] ]
}
resource "alicloud_network_interface_attachment" "pcs_ext_port_attachment" {
    instance_id       = alicloud_instance.pcs_instance.id
}
network_interface_id = alicloud_network_interface.pcs_ext_port.id
}
# create management port and attach to PCS instance
resource "alicloud_network_interface" "pcs_mgmt_port" {
  name                  = "pcs_ext_port"
  vswitch_id           = local.vswitch_id_map[local.zone]["pcs_mgmt_port"]
  security_groups  = [ local.security_group_id_map["pcs_mgmt_port"] ]
}
resource "alicloud_network_interface_attachment" "pcs_mgmt_port_attachment" {
  instance_id           = alicloud_instance.pcs_instance.id
  network_interface_id  = alicloud_network_interface.pcs_mgmt_port.id
}

# Create a new EIP
# Create a new EIP for External Port.
resource "alicloud_eip" "pcs_ext_port_eip" {
  bandwidth            = "5"
  internet_charge_type = "PayByTraffic"
}
resource "alicloud_eip_association" "pcs_ext_port_eip_asso" {
  allocation_id = alicloud_eip.pcs_ext_port_eip.id
  instance_id   = alicloud_eip_attachment.pcs_mgmt_port_attachment.id
}
# Create a new EIP for Management Port.
resource "alicloud_eip" "pcs_mgmt_port_eip" {
  bandwidth            = "5"
  internet_charge_type = "PayByTraffic"
}
resource "alicloud_eip_association" "pcs_mgmt_port_eip_asso" {
  allocation_id = alicloud_eip.pcs_mgmt_port_eip.id
  instance_id   = alicloud_network_interface.pcs_mgmt_port.id
  instance_type = "NetworkInterface"
  private_ip_address = alicloud_network_interface.pcs_mgmt_port.private_ip
}

# Output
output "pcs_int_port" {
  value = "${alicloud_instance.pcs_instance.private_ip}"}
output "pcs_ext_port" {
  value = "${alicloud_network_interface.pcs_ext_port.private_ip}"}
output "pcs_mgmt_port" {
  value = "${alicloud_network_interface.pcs_mgmt_port.private_ip}"}
Variables

```unity
variable access_key {
    default = "alicloud-access-key" #replace with the actual alicloud-access key
}
variable secret_key {
    default = "alicloud-secret-key" #replace with the actual alicloud-secret-key
}
variable region {
    default = "cn-beijing"
}
variable zone {
    default = "cn-beijing-a"
}
variable zone_a {
    default = "cn-beijing-a"
}
variable zone_b {
    default = "cn-beijing-b"
}
variable vpc {
    default = "vpc_beijing"
}
variable image_name {
    default = "pcs_91r4_alicloud_image"
}
variable image_from {
    default = "self"
    #default = "marketplace"
}
variable instance_type {
    default = "ecs.n1.large"
}
variable instance_name {
    default = "pcs_91r4_on_alicloud"
}
variable cpu_core_count {
    default = "4"
}
variable memory_size {
    default = "8"
}
variable eni_amount {
    default = "2"
}
variable oss_bucket {
    default = "bucket-beijing-darumuga"
}
variable instance_charge_type {
    default = "PostPaid"
}
```
User Data

```xml
```
References

Alibaba Cloud documentation: https://www.alibabacloud.com/help/

Requesting Technical Support

Technical product support is available through the Pulse Secure Global Support Center (PSGSC). If you have a support contract, then file a ticket with PSGSC.

- Product warranties—for product warranty information, visit https://www.pulsesecure.net.