# 实验一 VPC & EC2

（所有实验请在**宁夏**区完成）

## 一、创建VPC

### 1、通过向导创建VPC

1. 登录AWS控制台，点击屏幕上方**服务**，在联网中选择**VPC**

在VPC控制面板中，可以总览VPC中的所有资源。但注意，这些资源只是在**宁夏**区域中

1. 点击左侧的**弹性IP** -> **分配新地址** -> **分配** -> **关闭**

私有子网通过NAT网关实现对Internet的访问，该弹性IP用于绑定NAT网

关。

1. 单击右上角的**VPC控制面板** -> **启动VPC向导**
2. 选择**带有公有和私有子网的VPC**

这里可以有两个选项，是否带有私有子网。公有子网中的实例可直接访问Internet，私有子网中的实例无法直接访问Internet。但私有子网中的实例可通过NAT来访问Internet。

1. 配置VPC（没有涉及的子项保持默认配置）

**IPv4 CIDR块**：无需修改，使用默认值（**10.0.0.0/16**）

CIDR（Classless Inter-Domain Routing，无类别域间路由）它消除了传统的A类、B类和C类地址以及划分子网的概念，因而可以更加有效地分配IPv4的地址空间。它可以将好几个IP网络结合在一起，使用一种无类别的域间路由选择算法，使它们合并成一条路由从而减少路由表中的路由条目减轻Internet路由器的负担。

**VPC名称**：**My First VPC**

**公有子网的IPv4 CIDR**：**10.0.1.0/24**

**可用区**：选择**cn-northwest-1a**（第一个）

**公有子网名称**：**My First Public Subnet**

**私有子网的IPv4 CIDR**：**10.0.2.0/24**

**可用区**：选择**cn-northwest-1a**（第一个）

**私有子网名称**：**My First Private Subnet**

**弹性IP分配ID**：点击输入框，这时候会弹出我们第二步创建的弹性IP地址，单击选择。

1. 点击**创建VPC -> 确定**
2. 查看VPC

点击左侧的**您的VPC**，选择刚刚创建的VPC（My First VPC）检查相关内容

1. 删除资源

选择My First VPC（确保只勾选了这一个VPC），点击上方的**操作** -> **Delete VPC**这里提示我们无法删除VPC，因为里面有部分组件还没删除。

点击第二个蓝色链接（NAT gateway）-> 点击上方**操作** -> **删除NAT网关**-> **删除NAT网关**（等待状态变为已删除）

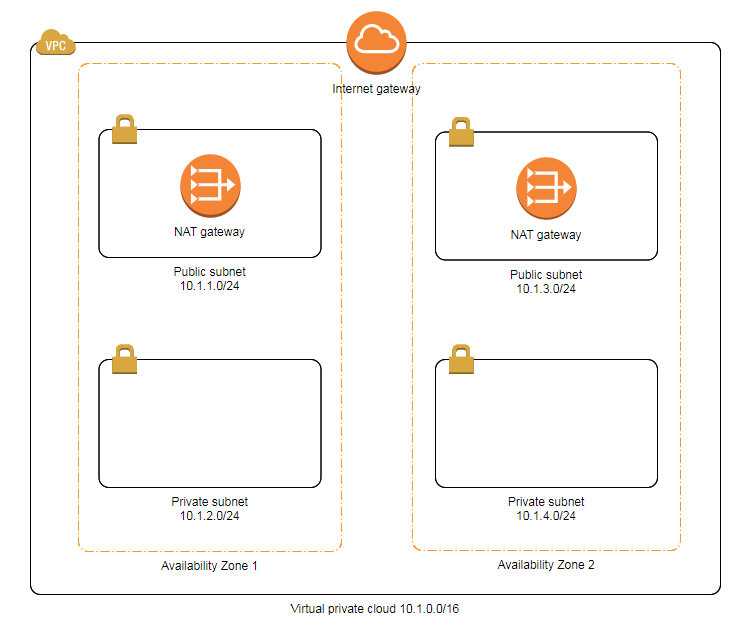
选择My First VPC（确保只勾选了这一个VPC），点击上方的**操作** -> **Delete VPC** -> **Delete VPC**

点击左侧弹性IP，选中刚刚创建的IP -> 点击上方的**操作** -> **释放地址** -> **发布**

通过向导我们创建了一个带有公有子网和私有子网的VPC，在向导创建VPC的过程中可以看到除了我们配置的内容外还创建了其它的组件。使用向导方便快捷的创建新VPC，但不够透明也不利于理解。下面我们手动的一步一步创建完整的VPC。

### 2、手动创建VPC

通过手动创建VPC我们可以实现多可用区的公有子网和私有子网，提高可靠性。同时通过VPC中的其它组件真正实现了子网的公开和私有。具体的拓扑图如下：



1. 创建VPC

在VPC服务中，选择左侧的**您的VPC** -> 点击上方的**Create VPC**

1. 配置VPC（没有涉及的子项无需修改）

**Name tag**：**VPC for Web**

**IPv4 CIDR block**：**10.1.0.0/16**

1. 点击**Create** -> **Close**，完成VPC创建
2. 创建公有子网1

在VPC服务中，选择左侧的**子网** -> 点击上方的**创建子网**

**名称标签**：**Public Subnet 1**

**VPC**：选择**VPC for Web**

**可用区**：选择**cn-northwest-1a**（第一个）

**IPv4 CIDR块**：**10.1.1.0/24**

点击**创建** -> **关闭**

1. 创建私有子网1

点击**创建子网**

**名称标签**：**Private** **Subnet 1**

**VPC**：选择**VPC for Web**

**可用区**：选择**cn-northwest-1a**（第一个）

**IPv4 CIDR块**：**10.1.2.0/24**

点击**创建** -> **关闭**

1. 创建公有子网2

点击上方的**创建子网**

**名称标签**：**Public Subnet 2**

**VPC**：选择**VPC for Web**

**可用区**：选择**cn-northwest-1b**（第二个）

**IPv4 CIDR块**：**10.1.3.0/24**

点击**创建** -> **关闭**

1. 创建私有子网2

点击**创建子网**

**名称标签**：**Private** **Subnet 2**

**VPC**：选择**VPC for Web**

**可用区**：选择**cn-northwest-1b**（第二个）

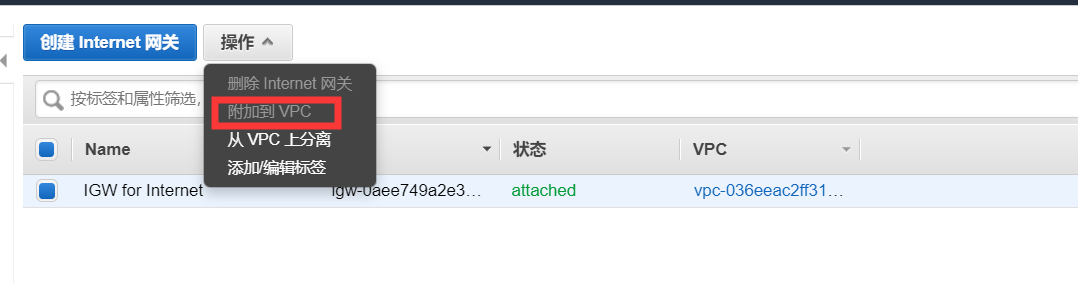
**IPv4 CIDR块**：**10.1.4.0/24**

点击**创建** -> **关闭**

1. 创建并配置Internet网关

为了让VPC中的实例与Internet进行通信，我们还需要创建Internet网关。在VPC服务中，选择左侧的**Internet网关** -> 点击上方的**创建Internet网关** -> 在**名称标签**中输入**IGW for Web** -> **创建** -> **关闭**。

我们发现刚刚创建的网关IGW for Web的状态为detached，我们需要将这个网关附加到上面创建好的VPC上。选中网关**IGW for Web** -> **操作** -> **附加到VPC** -> **VPC**选择**VPC for Web** -> **附加**。



1. 为公有子网配置路由表

上面的步骤已经创建好了2个公有子网，但这2个公有子网只是取名为公有子网而已，与其它的2个私有子网并没有什么不同。这里我们就要为2个公有子网配置路由规则，让2个公有子网名副其实。

在VPC服务中，选择左侧的**路由表** -> 点击上方的**创建路由表**。

**名称标签**：**rtw-public**

**VPC**：选择 **VPC for Web**

点击**创建** -> **关闭**

选中创建好的路由表**rtw-public** -> 选择下方的**路由**标签 -> 点击**编辑路由** -> **添加路由** -> 左侧目标输入**0.0.0.0/0** -> 右侧目标选择**Internet Gateway（IGW for Web）**-> **保存路由** -> **关闭**

选择**子网关联**标签 -> **编辑子网关联** -> 选择**Public Subnet 1**和**Public Subnet 2** -> **Save**

1. 为私有子网1创建NAT网关

NAT网关允许私有子网中的实例连接到 Internet 或其他 AWS 服务，但阻止 Internet 发起与这些实例的连接。

在VPC服务中，选择左侧的**NAT**网关 -> 点击上方的**创建NAT网关** -> 子网选择**Public Subnet 1** -> **新建EIP** -> **创建NAT网关** -> **关闭**

1. 为私有子网1配置路由表

在VPC服务中，选择左侧的**路由表** -> 点击上方的**创建路由表**

**名称标签**：**rtw-private1**

**VPC**：选择 **VPC for Web**

点击**创建** -> **关闭**

选中创建好的路由表**rtw-private1** -> 选择下方的**路由**标签 -> 点击**编辑路由** -> **添加路由** -> 左侧目标输入**0.0.0.0/0** -> 右侧目标选择**NAT Gateway**（选择刚刚创建的NAT网关ID）-> **保存路由** -> **关闭**

选择**子网关联**标签 -> **编辑子网关联** -> 选择**Private Subnet 1** -> **Save**

1. 为私有子网2创建NAT网关

在VPC服务中，选择左侧的**NAT网关** -> 点击上方的**创建NAT网关** -> 子网选择**Public Subnet 2** -> **新建EIP** -> **创建NAT网关** -> **关闭**

1. 为私有子网2配置路由表

在VPC服务中，选择左侧的**路由表** -> 点击上方的**创建路由表**

**名称标签**：**rtw-private2**

**VPC**：选择 **VPC for Web**

点击**创建** -> **关闭**

选中创建好的路由表**rtw-private2** -> 选择下方的**路由**标签 -> 点击**编辑路由** -> **添加路由** -> 左侧目标输入**0.0.0.0/0** -> 右侧目标选择**NAT Gateway**（选择刚刚创建的NAT网关ID）-> **保存路由** -> **关闭**

选择**子网关联**标签 -> **编辑子网关联** -> 选择**Private Subnet 2** -> **Save**

## 二、创建EC2

### 创建静态网站

1. 创建安全组

在AWS中安全组类似于虚拟防火墙，可以保护具体的实例。为了保护我们的静态网站实例，我们只需要开放80端口用于web访问和22端口用于SSH连接。

点击屏幕上方**服务** -> 在计算中选择**EC2** -> 左侧**安全组** -> **创建安全组**

**安全组名称**：**SG-Web**

**描述**：**security group for static web**

**VPC**：选择**VPC for Web**

在下方**安全组规则**中点击**入站** -> **添加规则** -> **类型**选择**HTTP** -> **添加规则**-> **类型**选择**SSH** -> **来源**选择**任何位置** -> **创建**

1. 创建EC2并部署静态网站

选择左侧的**实例** -> **启动实例** -> AMI选择**Amazon Linux 2 AMI**（第一个）-> 实例类型选择**t2.micro** -> **配置实例详细信息**（没有涉及的子项无需修改）

**网络**：选择**VPC for Web**

**子网**：选择**Public Subnet 1**

**自动分配公有IP**：**启用**

在**高级详细信息**的**用户数据**里输入：

#!/bin/bash

yum -y update

yum -y install httpd

chkconfig httpd on

systemctl start httpd

echo '<html><h1>Hello From Your Web Server!</h1></html>' > /var/www/html/index.html

点击**下一步：添加存储** -> **下一步：添加标签**

添加键为“Name”且值为“Webserver”的标签，注意区分大小写

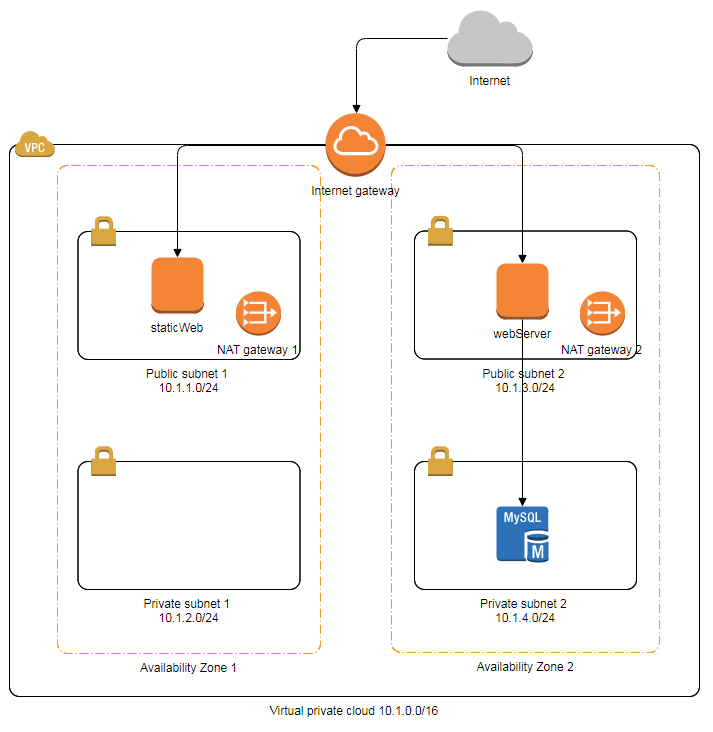
点击**下一步：配置安全组** -> **选择一个现有的安全组** -> **选择SG-Web** -> **审核和启动** -> **继续**（忽略警告）-> **启动** -> **选择在没有密钥对…** -> 勾选**我确认…** -> **启动实例** -> **查看实例**

1. 确认网站访问

等待实例的状态为**running**，复制实例的IPv4公有IP，将地址粘贴到浏览器地址栏当中访问。出现**Hello From Your Web Server!**表明实验成功。如果没出现可以在稍等数秒然后刷新页面。

### 数据与应用分离

上面完成了简单的静态网站实验，主要通过User Data 在启动实例时执行一段脚本。其实对于静态资源的托管，AWS提供了更好的选择S3。现在我们继续搭建一个简单的网站，这次我们将网站和数据分离。数据我们放到私有子网里从而保护数据的安全性，网站放在公有子网从而外部可见。具体拓扑图如下：



1. 创建安全组保护RDS的实例

点击屏幕上方**服务** -> 在计算中选择**EC2** -> 左侧**安全组** -> **创建安全组**

**安全组名称**：**SG-MySQL**

**描述**：**security group for** **MySQL**

**VPC**：选择**VPC for Web**

在下方**安全组规则**中点击**入站** -> **添加规则** -> **类型**选择**MYSQL/Aurora** -> **来源**输入**10.1.3.0/24** -> **创建**

1. 创建子网组

登录AWS控制台，点击屏幕上方**服务**，在数据库中选择**RDS**

点击左侧的**子网组** -> **创建数据库子网组**

**名称**：**subnetgroup-web**

**描述**：**for web db**

**VPC**：选择**VPC for Web**

**添加子网：可用区**先选择**cn-northwest-1a**，**子网**选择私有子网**（10.1.2.0/24）,**点击**添加子网；可用区**再选择**cn-northwest-1b**，**子网**选择私有子网**（10.1.4.0/24），**点击**添加子网。**（共添加两个私有子网）

点击**创建**完成子网组的创建

1. 创建RDS

点击左侧的**数据库** -> **Create database** -> 引擎选择**MySQL** -> 使用案例选择**开发/测试** -> 配置**数据库详细信息**（未列出子项使用默认值）

**数据库实例**：**db.t2.micro**

**取消启用存储自动扩展**

**数据库实例标识符**：**hello-db**

**主用户名**：**cstor**

**主密码**：**12345678**

**确认密码**：**12345678**

配置高级设置

**VPC**：选择**VPC for Web**

**子网组**：选择**subnetgroup-web**

**VPC安全组：选择现有安全组SG-MySQL**

**数据库名称**：**user**

**备份保留期**：选择**0天**

点击**创建数据库**等待数据库创建成功

1. 创建部署web的EC2实例

选择左侧的**实例** -> **启动实例** -> AMI选择**Amazon Linux 2 AMI**（第一个）-> 实例类型选择**t2.micro** -> **配置实例详细信息**（没有涉及的子项无需修改）

**网络**：选择**VPC for Web**

**子网**：选择**Public Subnet 2**

**自动分配公有IP**：**启用**

点击**下一步：添加存储** -> **下一步：添加标签** -> **下一步：配置安全组** -> **选择一个现有的安全组** -> **选择SG-Web**（忽略警告） -> **审核和启动** -> **启动** -> 选择**创建新密钥对 -> 密钥对名称**输入**my-server** -> **下载密钥对** -> **启动实例** -> **查看实例**

1. SSH连接EC2并修改部署网站

利用刚刚下载的my-server.pem密钥文件登录EC2，下载可执行文件webServer和配置文件server.conf。在终端中执行：

curl -o server.conf https://cstorfs.s3.cn-northwest-1.amazonaws.com.cn/ls/suze/server.conf

curl -o webServer https://cstorfs.s3.cn-northwest-1.amazonaws.com.cn/ls/suze/webServer

修改server.conf配置文件，将mysql\_host设置为刚刚创建数据库的Endpoint。user\_name修改为任意的用户名，user\_address修改为任意的地址。

切换到root用户赋予webServer可执行权限并执行。在终端中输入：

sudo –i

cd /home/ec2-user

chmod a+x webServer

nohup ./webServer &

1. 确认网站访问

在浏览器输入EC2实例的IPv4公有IP，将地址粘贴到浏览器地址栏当中访问。出现hello cstor（你自己设置的用户名）, from nanjing（你自己设置的地址）表明实验成功。如果没出现可以在稍等数秒然后刷新页面。