**Create an Amazon VPC for Use with an Amazon RDS DB Instance**

A common scenario includes an Amazon RDS DB instance in an Amazon VPC, that shares data with a web server that is running in the same VPC. In this tutorial you create the VPC for this scenario.

The following diagram shows this scenario. For information about other scenarios, see[Scenarios for Accessing a DB Instance in a VPC](https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_VPC.Scenarios.html).


            VPC and EC2 security group Scenario
        

Because your Amazon RDS DB instance only needs to be available to your web server, and not to the public Internet, you create a VPC with both public and private subnets. The web server is hosted in the public subnet, so that it can reach the public Internet. The Amazon RDS DB instance is hosted in a private subnet. The web server is able to connect to the Amazon RDS DB instance because it is hosted within the same VPC, but the Amazon RDS DB instance is not available to the public Internet, providing greater security.

**Create a VPC with Private and Public Subnets**

Use the following procedure to create a VPC with both public and private subnets.

**To create a VPC and subnets**

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. In the top-right corner of the AWS Management Console, choose the region to create your VPC in. This example uses the US West (Oregon) region.
3. In the upper-left corner, choose **VPC Dashboard**. To begin creating a VPC, choose **Start VPC Wizard**.
4. On the **Step 1: Select a VPC Configuration** page, choose **VPC with Public and Private Subnets**, and then choose **Select**.
5. On the **Step 2: VPC with Public and Private Subnets** page, set these values:
   * **IPv4 CIDR block:** 10.0.0.0/16
   * **IPv6 CIDR block:** No IPv6 CIDR Block
   * **VPC name:** tutorial-vpc
   * **Public subnet's IPv4 CIDR:** 10.0.0.0/24
   * **Availability Zone:** us-west-2a
   * **Public subnet name:** Tutorial public
   * **Private subnet's IPv4 CIDR:** 10.0.1.0/24
   * **Availability Zone:** us-west-2a
   * **Private subnet name:** Tutorial Private 1
   * **Instance type:** t2.small

**Important**

If you do not see the **Instance type** box in the console, click **Use a NAT instance instead**. This link is on the right.

**Note**

If the t2.small instance type is not listed, you can select a different instance type.

* + **Key pair name:** No key pair
  + **Service endpoints:** Skip this field.
  + **Enable DNS hostnames:** Yes
  + **Hardware tenancy:** Default

1. When you're finished, choose **Create VPC**.

**Create Additional Subnets**

You must have either two private subnets or two public subnets available to create an Amazon RDS DB subnet group for an RDS DB instance to use in a VPC. Because the RDS DB instance for this tutorial is private, add a second private subnet to the VPC.

**To create an additional subnet**

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. To add the second private subnet to your VPC, choose **VPC Dashboard**, choose **Subnets**, and then choose **Create Subnet**.
3. On the **Create Subnet** page, set these values:
   * **Name tag:** Tutorial private 2
   * **VPC:** Choose the VPC that you created in the previous step, for example: vpc-*identifier* (10.0.0.0/16) | tutorial-vpc
   * **Availability Zone:** us-west-2b

**Note**

Choose an Availability Zone that is different from the one that you chose for the first private subnet.

* + **IPv4 CIDR block:** 10.0.2.0/24

1. When you're finished, choose **Yes, Create**.
2. To ensure that the second private subnet that you created uses the same route table as the first private subnet, choose **VPC Dashboard**, choose **Subnets**, and then choose the first private subnet that you created for the VPC, Tutorial private 1.
3. Below the list of subnets, choose the **Route Table** tab, and note the value for **Route Table**—for example: rtb-98b613fd.
4. In the list of subnets, deselect the first private subnet.
5. In the list of subnets, choose the second private subnet Tutorial private 2, and choose the **Route Table** tab.
6. If the current route table is not the same as the route table for the first private subnet, choose **Edit**. For **Change to**, choose the route table that you noted earlier—for example: rtb-98b613fd.
7. To save your selection, choose **Save**.

**Create a VPC Security Group for a Public Web Server**

Next you create a security group for public access. To connect to public instances in your VPC, you add inbound rules to your VPC security group that allow traffic to connect from the internet.

**To create a VPC security group**

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. Choose **VPC Dashboard**, choose **Security Groups**, and then choose **Create Security Group**.
3. On the **Create Security Group** page, set these values:
   * **Name tag:** tutorial-securitygroup
   * **Group name:** tutorial-securitygroup
   * **Description:** Tutorial Security Group
   * **VPC:** Choose the VPC that you created earlier, for example: vpc-*identifier*(10.0.0.0/16) | tutorial-vpc
4. To create the security group, choose **Yes, Create**.

**To add inbound rules to the security group**

1. Determine the IP address that you will use to connect to instances in your VPC. To determine your public IP address, you can use the service at [https://checkip.amazonaws.com](https://checkip.amazonaws.com/). An example of an IP address is 203.0.113.25/32.

If you are connecting through an Internet service provider (ISP) or from behind your firewall without a static IP address, you need to find out the range of IP addresses used by client computers.

**Warning**

If you use 0.0.0.0/0, you enable all IP addresses to access your public instances. This approach is acceptable for a short time in a test/LAB environment, but it's unsafe for production environments. In production, you'll authorize only a specific IP address or range of addresses to access your instances.

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. Choose **VPC Dashboard**, choose **Security Groups**, and then choose the tutorial-securitygroup security group that you created in the previous procedure.
3. Choose the **Inbound Rules** tab, and then choose **Edit**.
4. Set the following values for your new inbound rule to allow Secure Shell (SSH) access to your EC2 instance. If you do this, you can connect to your EC2 instance to install the web server and other utilities, and to upload content for your web server.
   * **Type:** SSH (22)
   * **Source:** The IP address or range from Step 1, for example: 203.0.113.25/32. Or 0.0.0.0/0 for this TEST/LAB
5. Choose **Add another rule**.
6. Set the following values for your new inbound rule to allow HTTP access to your web server.
   * **Type:** HTTP (80)
   * **Source:** 0.0.0.0/0.
7. To save your settings, choose **Save**.

**Create a VPC Security Group for a Private Amazon RDS DB Instance**

To keep your Amazon RDS DB instance private, create a second security group for private access. To connect to private instances in your VPC, you add inbound rules to your VPC security group that allow traffic from your web server only.

**To create a VPC security group**

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. Choose **VPC Dashboard**, choose **Security Groups**, and then choose **Create Security Group**.
3. On the **Create Security Group** page, set these values:
   * **Name tag:** tutorial-db-securitygroup
   * **Group name:** tutorial-db-securitygroup
   * **Description:** Tutorial DB Instance Security Group
   * **VPC:** Choose the VPC that you created earlier, for example: vpc-*identifier*(10.0.0.0/16) | tutorial-vpc
4. To create the security group, choose **Yes, Create**.

**To add inbound rules to the security group**

1. Open the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
2. Choose **VPC Dashboard**, choose **Security Groups**, and then choose the tutorial-db-securitygroup security group that you created in the previous procedure.
3. Choose the **Inbound Rules** tab, and then choose **Edit**.
4. Set the following values for your new inbound rule to allow MySQL traffic on port 3306 from your EC2 instance. If you do this, you can connect from your web server to your DB instance to store and retrieve data from your web application to your database.
   * **Type:** MySQL/Aurora (3306)
   * **Source:** The identifier of the tutorial-securitygroup security group that you created previously in this tutorial, for example: sg-9edd5cfb.
5. To save your settings, choose **Save**.

**Create a DB Subnet Group**

A DB subnet group is a collection of subnets that you create in a VPC and that you then designate for your DB instances. A DB subnet group allows you to specify a particular VPC when creating DB instances.

**To create a DB subnet group**

1. Open the Amazon RDS console at <https://console.aws.amazon.com/rds/>.
2. In the navigation pane, choose **Subnet groups**.
3. Choose **Create DB Subnet Group**.
4. On the **Create DB subnet group** page, set these values in **Subnet group details**:
   * **Name:** tutorial-db-subnet-group
   * **Description:** Tutorial DB Subnet Group
   * **VPC:** tutorial-vpc (vpc-*identifier*)
5. In the **Add subnets** section, click the **Add all the subnets related to this VPC** link.
6. Choose **Create**.

Your new DB subnet group appears in the DB subnet groups list on the RDS console. You can click the DB subnet group to see details, including all of the subnets associated with the group, in the details pane at the bottom of the window.