AWS Windows Server Setup Guide

A comprehensive guide to setting up a secure Windows Server infrastructure on AWS with VPC, bastion host, and proper network configuration.

Table of Contents

Prerequisites & AWS Console Access

Step 1.1: Login to AWS Console

1. Open your web browser (Chrome, Firefox, or Edge recommended)
2. Navigate to: **https://aws.amazon.com**
3. Click **"Sign In to the Console"** (top right corner)
4. Enter your AWS account email or IAM username
5. Click **"Next"**
6. Enter your password
7. If MFA is enabled, enter the 6-digit code from your authenticator app
8. You should now see the AWS Management Console homepage

Step 1.2: Select Your Region

1. Look at the top-right corner of the console
2. Click on the region dropdown
3. Select your preferred region:
   - **US East (N. Virginia) -** `us-east-1` (recommended)
   - **US West (Oregon) -** `us-west-2`

> ⚠️ **IMPORTANT:** All resources MUST be created in the SAME region

---

Create VPC and Network Infrastructure

Step 2.1: Navigate to VPC Service

1. Click the search bar at the top of the AWS Console
2. Type: **VPC**
3. Click on **"VPC"** under Services
4. You're now in the VPC Dashboard

Step 2.2: Start VPC Creation

1. Click **"Your VPCs"** in the left sidebar
2. Click the orange **"Create VPC"** button (top-right)

Step 2.3: Choose VPC Creation Method

1. Select **"VPC and more"** radio button
2. This creates VPC + Subnets + Route Tables + Gateways automatically

Step 2.4: Configure VPC Settings

Name Tag Auto-Generation

- **Name tag auto-generation:** `windows-project`
- Preview will show: `windows-project-vpc`, `windows-project-subnet-public1`, etc.

IPv4 CIDR Block

- Select: **"IPv4 CIDR manual input"**
- Enter: `10.0.0.0/16`
- This provides 65,536 IP addresses

IPv6 CIDR Block

- Select: **"No IPv6 CIDR block"**

Tenancy

- Select: **"Default"**

Availability Zones

- **Number of Availability Zones (AZs):** `2`
- **Number of public subnets:** `2`
- **Number of private subnets:** `2`

Customize Subnet CIDR Blocks

Click "Customize subnets CIDR blocks" and enter:

Subnet Type	Availability Zone	CIDR Block
Public subnet	us-east-1a	`10.0.1.0/24`
Public subnet	us-east-1b	`10.0.2.0/24`
Private subnet	us-east-1a	`10.0.11.0/24`
Private subnet	us-east-1b	`10.0.12.0/24`

NAT Gateways

- Select: **"In 1 AZ"**
- ⚠️ Cost: $0.045/hour (~$32/month)

VPC Endpoints

- Select: **"None"**

DNS Options

- ✅ **Enable DNS hostnames**
- ✅ **Enable DNS resolution**

Step 2.5: Review and Create

1. Review the preview diagram showing your network architecture
2. Click the orange **"Create VPC"** button
3. Wait for all resources to be created (2-3 minutes)
4. Look for "Successfully created VPC" with a green checkmark
5. Click **"View VPC"**

Verify and Configure VPC Components

Step 3.1: Verify Your VPC

Confirm the following:

- **State:** Available
- **IPv4 CIDR:** `10.0.0.0/16`
- **DNS hostnames:** Enabled

- **DNS resolution:** Enabled

Step 3.2: Verify Subnets

Navigate to **Subnets** in the left sidebar. You should see 4 subnets:

Name	AZ	CIDR	Type
windows-project-subnet-public1-us-east-1a	us-east-1a	`10.0.1.0/24`	Public
windows-project-subnet-public2-us-east-1b	us-east-1b	`10.0.2.0/24`	Public
windows-project-subnet-private1-us-east-1a	us-east-1a	`10.0.11.0/24`	Private
windows-project-subnet-private2-us-east-1b	us-east-1b	`10.0.12.0/24`	Private

Step 3.3: Enable Auto-Assign Public IP (CRITICAL)

For each PUBLIC subnet:

1. Click on the subnet name
2. Click **"Actions"** dropdown (top-right)
3. Select **"Edit subnet settings"**
4. ✅ Check **"Enable auto-assign public IPv4 address"**
5. Click **"Save"**

Repeat for both public subnets. Private subnets should remain "No".

Step 3.4: Verify Internet Gateway

1. Navigate to **"Internet Gateways"** in the left sidebar
2. Verify `windows-project-igw` exists
3. Confirm:
   - **State:** Attached
   - **VPC ID:** Your VPC ID

Step 3.5: Verify NAT Gateway

1. Navigate to **"NAT Gateways"**
2. Verify `windows-project-nat-public1-us-east-1a` exists
3. Confirm:
   - **Status:** Available (green)
   - **Subnet:** public1 subnet
   - **Connectivity type:** Public

- **Elastic IP address:** Note this IP address

> 💡 **Tip:** Write down the Elastic IP address for future reference

Step 3.6: Verify Route Tables

Navigate to **"Route Tables"** and verify:

Public Route Table

Destination	Target	Status
`10.0.0.0/16`	local	Active
`0.0.0.0/0`	igw-xxxxx	Active

Private Route Table(s)

Destination	Target	Status
`10.0.0.0/16`	local	Active
`0.0.0.0/0`	nat-xxxxx	Active

Create Security Groups

Step 4.1: Create Bastion Host Security Group

1. Navigate to **"Security Groups"** in the left sidebar
2. Click **"Create security group"**

Basic Details:

- **Security group name:** `Bastion-SG`
- **Description:** Allow RDP access from my IP to bastion host
- **VPC:** `windows-project-vpc`

Inbound Rules:

Type	Protocol	Port Range	Source	Description
RDP	TCP	3389	My IP	RDP from my computer

Outbound Rules:

- Leave as default (All traffic to `0.0.0.0/0`)

3. Click **"Create security group"**

Step 4.2: Create Private Windows Server Security Group

1. Click **"Create security group"** again

Basic Details:

- **Security group name:** `Private-Windows-SG`
- **Description:** Allow RDP only from Bastion host and ICMP from VPC
- **VPC:** `windows-project-vpc`

Inbound Rules:

Type	Protocol	Port Range	Source	Description
RDP	TCP	3389	Bastion-SG	RDP from bastion host only
All ICMP - IPv4	ICMP	All	`10.0.0.0/16`	Allow ping from within VPC

Outbound Rules:

- Leave as default (All traffic to `0.0.0.0/0`)

2. Click **"Create security group"**

Launch Bastion Host

Step 5.1: Navigate to EC2

1. Click the search bar at the top
2. Type: **EC2**
3. Click on **"EC2"** under Services

Step 5.2: Start Instance Launch

1. Click **"Instances"** in the left sidebar
2. Click the orange **"Launch instances"** button

Step 5.3: Configure Bastion Instance

Name and Tags

- **Name:** `Windows-Bastion-Host`

Optional tags:

- `Environment: Demo`
- `Purpose: Bastion`

Application and OS Images (AMI)

1. Click the **"Windows"** tile under Quick Start
2. Select **"Microsoft Windows Server 2022 Base"**
3. Look for the "Free tier eligible" badge

Instance Type

- Select: **t3.small** (2 vCPU, 2 GiB Memory)
- Cost: $0.0208/hour (~$15/month if running 24/7)

Key Pair

If creating new key pair:

1. Click **"Create new key pair"**
2. **Key pair name:** `windows-server-key`
3. **Key pair type:** RSA
4. **Private key file format:** .pem
5. Click **"Create key pair"**
6. ⚠️ **CRITICAL:** Save the downloaded `.pem` file securely!

Network Settings

Click **"Edit"** and configure:

Setting	Value
VPC	`windows-project-vpc`
Subnet	`windows-project-subnet-public1-us-east-1a`
Auto-assign public IP	**Enable**
Firewall (security groups)	Select existing: `Bastion-SG`

Configure Storage

- **Size:** 30 GiB
- **Volume type:** gp3
- **Delete on termination:** ✅ Checked

Advanced Details (Optional but Recommended)

- **Shutdown behavior:** Stop

- **Termination protection:** ✅ Enable
- **Detailed CloudWatch monitoring:** ✅ Enable

Step 5.4: Review and Launch

1. Review all settings in the Summary panel
2. Click **"Launch instance"**
3. Click on the Instance ID to view details

Step 5.5: Wait for Instance to Start

1. Monitor **Instance state** - wait for "Running" (green)
2. Monitor **Status checks** - wait for "2/2 checks passed"
3. This takes 2-3 minutes for Windows instances

Step 5.6: Note Instance Details

Record the following information:

- **Instance ID:** `i-0abc...`
- **Public IPv4 address:** (e.g., `3.25.67.89`)
- **Private IPv4 address:** (e.g., `10.0.1.45`)
- **Security groups:** `Bastion-SG`

Step 5.7: Get Windows Administrator Password

1. Select your bastion instance
2. Click **"Connect"** button (orange, top-right)
3. Click the **"RDP client"** tab
4. Click **"Get password"**
5. Click **"Upload private key file"** or **"Browse"**
6. Select your `windows-server-key.pem` file
7. Click **"Decrypt password"**
8. **Copy and save the password securely**

Step 5.8: Connect to Bastion Host via RDP

On Windows:

1. Press Windows key and type: **Remote Desktop Connection**
2. Enter the **Public IPv4 address** in the Computer field

3. Click **"Show Options"** → Enter **Username:** `Administrator`
4. Click **"Connect"**
5. Enter the decrypted password
6. Click **"Yes"** on the certificate warning
7. Success! You're connected to Windows Server

On Mac:

1. Install **Microsoft Remote Desktop** from the App Store
2. Click **"Add PC"**
3. **PC name:** Enter the Public IPv4 address
4. **User account:** Add user
    - **Username:** `Administrator`
    - **Password:** The decrypted password
5. Double-click the PC to connect
6. Click **"Continue"** on the certificate warning

On Linux:

1. Install Remmina: `sudo apt install remmina`
2. Open Remmina and click "+" to add a new connection
3. **Protocol:** RDP
4. **Server:** Enter the Public IPv4 address
5. **Username:** `Administrator`
6. **Password:** The decrypted password
7. Click **"Connect"**

Summary

You have successfully created:

✅ A VPC with public and private subnets across 2 Availability Zones
✅ Internet Gateway for public subnet internet access
✅ NAT Gateway for private subnet outbound internet access
✅ Security groups with proper access controls

✅ A Windows Server 2022 bastion host in the public subnet
✅ RDP connection to your bastion host

Next Steps

- Launch a private Windows Server in the private subnet
- Connect to the private server through the bastion host
- Configure Active Directory or other Windows services
- Set up backups and monitoring

Cost Considerations

Monthly costs (if running 24/7):

- NAT Gateway: ~$32/month
- t3.small instance: ~$15/month
- EBS storage (30 GiB): ~$3/month
- **Total: ~$50/month**

> 💡 **Tip:** Stop instances when not in use to save costs!

Troubleshooting

Cannot Connect via RDP

- Verify security group allows RDP (port 3389) from your IP
- Confirm instance is in "Running" state with 2/2 status checks
- Check that the public subnet has auto-assign public IP enabled
- Verify you're using the correct public IP address

NAT Gateway Not Working

- Verify NAT Gateway status is "Available"
- Check private route table has route to NAT Gateway (`0.0.0.0/0`)
- Confirm Elastic IP is allocated and attached to NAT Gateway

Instance Fails to Launch

- Check you haven't exceeded EC2 instance limits
- Verify the subnet and VPC are in the same region
- Ensure you have sufficient permissions in your AWS account

📝 **Document Version:** 1.0

📅 **Last Updated:** November 2024

🔧 **AWS Services Used:** VPC, EC2, Security Groups, Internet Gateway, NAT Gateway