

Launch AWS EC2 Instance with Intel® Distribution of OpenVINO™ toolkit

Enabling developers to optimize pre-trained models and accelerate the deployment of deep learning solutions with a write-once-deploy-anywhere approach across Intel® powered CPUs, integrated GPUs, Intel® Movidius™ VPUs, and FPGAs.

The Intel® Distribution of OpenVINO™ toolkit on Amazon Machine Image (AMI) enables developers to optimize pre-trained models and accelerate the deployment of deep learning solutions with a write-once-deploy-anywhere approach across Intel-powered CPUs, integrated GPUs, Intel® Movidius™ VPUs, and FPGAs.

This AMI comes pre-equipped with the Intel® Distribution of OpenVINO™ toolkit development and deployment components, such as the Model Optimizer and the Inference Engine.

This document illustrates all the steps required to deploy AMI in your AWS account and access Jupyter environment from your local machine. Please note that the AWS account you are using for deploying this AMI need to have public IP assigned for Jupyter notebooks to be accessed from a local system.

Table of Contents

Search for AMI and Subscribe	3
1. Click on “Continue to Subscribe”	3
2. Click on “Accept Terms”	3
3. Click on “Continue to Configuration”	4
4. Click on “Continue to Launch”	4
5. You have two options to launch the AMI under “Choose Action”,.....	4
Launch option 1: “Launch from Website:.....	5
1. Select your desired EC2 Instance Type	5
2. Select your VPC Settings.....	5
3. Select your Subnet Settings.....	6
4. Under Security Group Settings, Click on “Create New Based on Seller Settings”.....	6
5. Select an existing Key Pair or create new key pair	7
6. After selecting all the required fields above, click “Launch”	7
AMI Launch option 2: “Launch through EC2”:	7
1. Choose instance type.....	8
2. Configure Instance.....	8
3. Add storage	8
4. Add tags	9
5. Security Group.....	9
6. Launch your instance.....	10
7. Choose or create a new private key file	10
8. Click on the instance id to see your instance status	11
9. Click “Connect” to view instructions to SSH into the instance.	12
10. Connect to your instance via Terminal.....	12
11. Connect to Jupyter Notebook.....	13
12. Change Jupyter Notebook password.	13
Tips: Stop the instance when not in use to prevent additional charges.....	14
Create VPC with internet access	15
1. Navigate to VPCàYour VPCsàCreate VPC.....	15
2. Create an internet gateway with the following details.	16
3. Attach the created VPC to internet gateway	17
4. Create Public Subnet.....	17
5. Create Route Table.....	18
6. Attach the created internet gateway in the route table.....	18

7. Attach the created public subnet in the route table	19
8. Create Network ACL.....	19
9. Click Edit inbound rules and add the following rules.....	20
Create Security Group	21

Search for AMI and Subscribe

There are two ways to subscribe to the OpenVINO™ marketplace offering:

Option 1: Navigate to AMI webpage directly by using this link

<https://aws.amazon.com/marketplace/pp/B08L6ZHYKH/>

Option 2: Navigate to AWS marketplace by using the link <https://aws.amazon.com/marketplace> and then search for “Intel® Distribution of OpenVINO™ Toolkit”. Select the “Intel® Distribution of OpenVINO™ Toolkit” AMI.

After selecting the AMI in the AWS marketplace:

1. Click on “Continue to Subscribe”

The screenshot shows the AWS Marketplace interface. At the top, there's a navigation bar with links like 'Categories', 'Delivery Methods', 'Solutions', 'Migration Mapping Assistant', 'Your Saved List', 'Partners', 'Sell in AWS Marketplace', 'Amazon Web Services Home', and 'Help'. Below the navigation, the product title 'Intel® Distribution of OpenVINO™ Toolkit' is displayed, along with the seller 'By: Intel' and the latest version '2021.1'. It says 'Get started with Intel® Distribution of OpenVINO™ Toolkit' and lists 'Linux/Unix' as the platform with '0 AWS reviews'. On the right side, there's a yellow 'Continue to Subscribe' button with a red arrow pointing to it. Below the button, there's a 'Save to List' button and a price section showing 'Typical Total Price \$0.432/hr'. At the bottom of the main content area, there are tabs for 'Overview', 'Pricing', 'Usage', 'Support', and 'Reviews', with 'Overview' being the active tab. To the right of the main content, there's a 'Highlights' section with a bulleted list of features.

2. Click on “Accept Terms”

The screenshot shows the 'Continue to Configuration' step of the subscription process. It displays a message: 'You must first review and accept terms.' Below this, there are links for '< Product Detail' and 'Subscribe'. Under the heading 'Subscribe to this software', it says 'To create a subscription, review the pricing information and accept the terms for this software.' A 'Terms and Conditions' section follows, with a sub-section 'Intel Offer'. A red arrow points to the 'Accept Terms' button, which is highlighted in orange. The page also includes a note about pricing and a table for component details at the bottom.

3. Click on “Continue to Configuration”

The screenshot shows the AWS Marketplace interface for the Intel® Distribution of OpenVINO™ Toolkit. At the top, there's a navigation bar with links like 'Categories', 'Delivery Methods', 'Solutions', 'Migration Mapping Assistant', 'Your Saved List', 'Partners', 'Sell in AWS Marketplace', 'Amazon Web Services Home', and 'Help'. A search bar is also present. Below the navigation, the product name 'Intel® Distribution of OpenVINO™ Toolkit' is displayed. A prominent yellow button labeled 'Continue to Configuration' is highlighted with a red arrow pointing towards it. Below this button, there are links for '< Product Detail' and 'Subscribe'. The main content area is titled 'Subscribe to this software' and contains a message about being subscribed and terms and pricing details. It also includes sections for 'Terms and Conditions' and 'Intel Offer', along with legal notices. A table at the bottom lists columns for 'Product', 'Effective date', 'Expiration date', and 'Action'.

4. Click on “Continue to Launch”

This screenshot shows the continuation of the AWS Marketplace process for launching the toolkit. The top navigation bar and product title are identical to the previous screen. The 'Continue to Configuration' button has been replaced by a new yellow button labeled 'Continue to Launch', which is also highlighted with a red arrow. Below this button, there are links for '< Product Detail', 'Subscribe', and 'Configure'. The main content area is titled 'Configure this software' and asks to choose a fulfillment option and enter configuration information. On the left, there are dropdown menus for 'Delivery Method' (set to '64-bit (x86) Amazon Machine Image (AMI)'), 'Software Version' (set to '2021.1 (Oct 14, 2020)'), and 'Region' (set to 'US East (N. Virginia)'). On the right, there's a 'Pricing information' section with a detailed breakdown of costs for the selected configuration. It includes 'Software Pricing' for the toolkit running on 'c5n.2xlarge' and 'Infrastructure Pricing' for EC2 usage.

5. You have two options to launch the AMI under “Choose Action”,

- Launch from Website**
- Launch Through EC2**

The screenshot shows the AWS Marketplace interface for the Intel® Distribution of OpenVINO™ Toolkit. At the top, there's a navigation bar with links like 'Categories', 'Delivery Methods', 'Solutions', 'Migration Mapping Assistant', 'Your Saved List', 'Partners', 'Sell in AWS Marketplace', 'Amazon Web Services Home', and 'Help'. The user is logged in as 'Hello, assumed-role/Local_A...'. Below the navigation, the product title 'Intel® Distribution of OpenVINO™ Toolkit' is displayed, along with a 'Launch this software' button. A 'Configuration Details' section provides information about the fulfillment option (64-bit x86 Amazon Machine Image (AMI) running on c5n.2xlarge), software version (2021.1), and region (US East (N. Virginia)). A 'Usage Instructions' button is also present. The main focus is the 'Choose Action' dropdown, which is open and set to 'Select a launch action', indicated by a red arrow.

Launch option 1: “Launch from Website”

1. Select your desired EC2 Instance Type

This screenshot continues from the previous one, showing the 'Choose Action' dropdown now set to 'Launch from Website'. The next step is to select an EC2 instance type. The 'EC2 Instance Type' dropdown is open, showing 'c5n.2xlarge' as the selected option, with a red arrow pointing to it. Below the dropdown, detailed instance specifications are shown: Memory: 21 GiB, CPU: 8 virtual cores, Storage: EBS Only, and Network Performance: Up to 25 Gigabit Ethernet.

2. Select your VPC Settings

This screenshot shows the 'VPC Settings' section. The 'EC2 Instance Type' dropdown is still set to 'c5n.2xlarge'. The 'VPC Settings' dropdown is open, showing a redacted value, with a red arrow pointing to it. Below the dropdown, a note states '* indicates a default vpc' and a link says 'Create a VPC in EC2'.

3. Select your Subnet Settings

The screenshot shows the 'Intel® Distribution of OpenVINO™ Toolkit' page in the AWS Marketplace. A red arrow points to the 'Subnet Settings' section, which includes a dropdown menu and a note about creating a subnet in EC2. Below it is a 'Security Group Settings' section with a dropdown menu set to 'launch-wizard-3'. A second red arrow points to the 'Create a subnet in EC2' link.

4. Under Security Group Settings, Click on “Create New Based on Seller Settings”.

- Enter “Name” and “Description” for the security group.
- For Source (IP or Group), we recommend “MY IP”, you could choose “anywhere” also.
- The security group needs two open ports, “22” for SSH login and “8888” for Jupyter access.

The screenshot shows the 'Intel® Distribution of OpenVINO™ Toolkit' page. A red arrow points to the 'Create New Based On Seller Settings' button in the 'Security Group Settings' section. Below it is a 'Key Pair Settings' section.

The screenshot shows the 'Create new based on seller settings' dialog. A red arrow points to the 'Name your security group' input field containing 'ov-sg'. Another red arrow points to the 'Description' input field containing 'openvino-ami-sg'. A third red arrow points to the 'Port Range' column for the first rule, which has '22' and 'My Ip' selected. A fourth red arrow points to the 'Port Range' column for the second rule, which has '8888' and 'My Ip' selected. A yellow callout box at the bottom left states: 'Rules with source of 0.0.0.0/0 allows all IP addresses to access your instance. We recommend limiting access to only known IP addresses.' A red arrow points to the 'Save' button at the bottom right.

Intel and the Intel logo are the trademarks of Intel Corporation or its subsidiaries.

5. Select an existing Key Pair or create new key pair

The screenshot shows the AWS Marketplace interface for launching the Intel® Distribution of OpenVINO™ Toolkit. In the 'Key Pair Settings' section, there is a dropdown menu labeled 'Select a key pair'. A red arrow points to this dropdown. Below it, there is a link 'Create a key pair in EC2'.

6. After selecting all the required fields above, click “Launch”

The screenshot shows the same AWS Marketplace interface as the previous one, but now the 'demo-key' has been selected in the dropdown. A red arrow points to the yellow 'Launch' button at the bottom right of the form.

AMI Launch option 2: “Launch through EC2”:

The screenshot shows the AWS Marketplace interface for launching the Intel® Distribution of OpenVINO™ Toolkit. In the 'Choose Action' section, there is a dropdown menu with 'Launch through EC2' selected. A red arrow points to this dropdown. Below it, there is a note: 'Choose this action to launch your configuration through the Amazon EC2 console.' To the right of the dropdown is a yellow 'Launch' button, which is also highlighted with a red arrow.

1. Choose instance type

Choose an instance and then click “Configure Instance Details”. Here we have picked a c5.4xlarge (68 ECUs, 16 vCPUs, 32 GiB memory, EBS only).

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Currently selected: c5.4xlarge (- ECUs, 16 vCPUs, 3.4 GHz, ~, 32 GiB memory, EBS only)

Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
c5	c5.large	2	4	EBS only	Yes	Up to 10 Gigabit	Yes
c5	c5.xlarge	4	8	EBS only	Yes	Up to 10 Gigabit	Yes
c5	c5.2xlarge	8	16	EBS only	Yes	Up to 10 Gigabit	Yes
c5	c5.4xlarge	16	32	EBS only	Yes	Up to 10 Gigabit	Yes
^	^	^	^	^	^	^	^

Cancel Previous Review and Launch **Next: Configure Instance Details**

2. Configure Instance

Compute optimized Instance (C5) type is [Available Only in a VPC](#). Choose the default VPC and enable “Auto-assign Public IP. Optionally follow instructions [Create VPC with internet access section](#) below to create new VPC with internet access.

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1

Purchasing option: Request Spot instances

Network: No default VPC found. [Create a new default VPC](#)

Subnet: 249 IP Addresses available [Create new subnet](#)

Auto-assign Public IP: Enable

Placement group: Add instance to placement group

Capacity Reservation: Open

Cancel Previous **Review and Launch** **Next: Add Storage**

3. Add storage

Update the Storage size based on your Requirement.

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-01b208bd4d49318b0	30	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#) [Previous](#) **Review and Launch** [Next: Add Tags](#)

Feedback English (US) © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use



4. Add tags

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. A copy of a tag can be applied to volumes, instances or both. Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	(128 characters maximum)	Value	(256 characters maximum)	Instances	Volumes
<i>This resource currently has no tags</i>					

Choose the Add tag button or [click to add a Name tag](#). Make sure your [IAM policy](#) includes permissions to create tags.

[Add Tag](#) (Up to 50 tags maximum)

[Cancel](#) [Previous](#) **Review and Launch** [Next: Configure Security Group](#)

Feedback English (US) © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use



5. Security Group

Choose existing security group or follow [Create security group](#) to create new security group . Click “Review and Launch”

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below.

[Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group Select an existing security group

Security Group ID	Name	Description
[REDACTED]	default	default VPC security group
[REDACTED]	Demo_SG	Allow SSH access to developers

Select a security group above to view its inbound rules.

Cancel Previous Review and Launch

Feedback English (US) © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

6. Launch your instance

Click Launch on the Review page.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

⚠ Your instance configuration is not eligible for the free usage tier

To launch an instance that's eligible for the free usage tier, check your AMI selection, instance type, configuration options, or storage devices. Learn more about [free usage tier](#) eligibility and usage restrictions.

Don't show me this again

AMI Details

Intel® Distribution of OpenVINO™ Toolkit

OpenVINO Intel Distribution of OpenVINO-2021.1-AmazonLinux2

Root Device Type: ebs Virtualization type: hvm

Hourly Software Fees: \$0.00 per hour on c5.4xlarge instance. Additional taxes or fees may apply. Software charges will begin once you launch this AMI and continue until you terminate the instance.

By launching this product, you will be subscribed to this software and agree that your use of this software is subject to the pricing terms and the seller's

Cancel Previous Launch

Feedback English (US) © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

7. Choose or create a new private key file

Choose an existing private key file or create a new one by selecting “create a new key pair” and click “Download Key Pair” to save it. Then click “Launch Instance”.

Screenshot of the AWS Step 7: Review Instance Launch wizard. The 'Select an existing key pair or create a new key pair' step is shown. A red box highlights the dropdown menu where 'demo-key' is selected. Red arrows point to the 'Launch Instances' button and the 'Select a key pair' dropdown. The 'Launch' button at the bottom right is also highlighted.

Screenshot of the AWS Launch Status page. It shows a circular progress icon and the text 'Initiating Instance Launches'. Below it, 'Please do not close your browser while this is loading' and 'Initiating launches...' are displayed. A red arrow points to the 'Initiating launches...' text.

8. Click on the “instance id” to see your instance status.

Also, copy this instance-id, as this will be your password to your Jupyter Notebook.

Screenshot of the AWS Launch Status page. A red arrow points to the instance ID 'i-0719207e3bde38334' which is listed under the message 'Your instances are now launching'. Below this, a box contains the message 'Get notified of estimated charges' and a link to 'Create billing alerts'. The 'How to connect to your instances' section is partially visible at the bottom.

Intel and the Intel logo are the trademarks of Intel Corporation or its subsidiaries.

9. Click “Connect” to view instructions to SSH into the instance.

The screenshot shows the AWS EC2 Instances page. At the top, there's a navigation bar with tabs like 'Instances', 'Launch Templates', etc., and a search/filter bar. Below it is a table listing one instance. The instance details include its ID (i-0719207e3bde38334), state (Running), type (c5.4xlarge), and status (2/2 checks). The 'Connect' button is highlighted with a red arrow. In the 'Instance summary' section, the Public IPv4 address is listed with a red arrow pointing to the 'open address' link.

10. Connect to your instance via Terminal

Open a terminal then connect to your instance using SSH. (NOTE: Replace text below in red.)

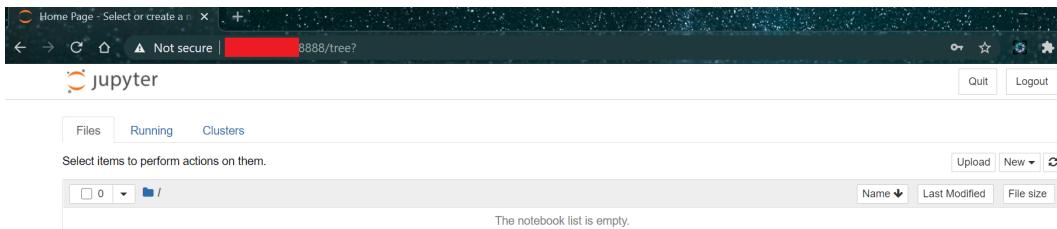
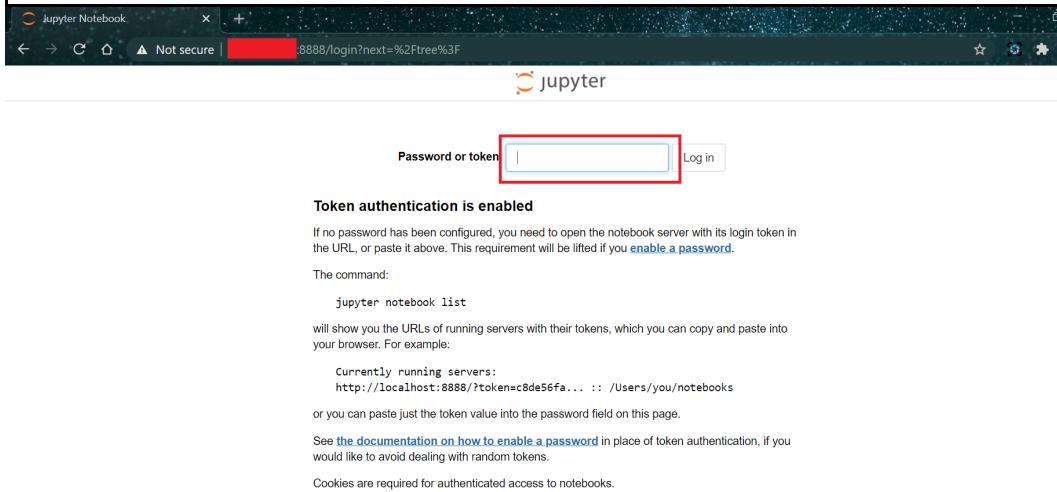
```
cd /Users/your_username/Downloads/  
chmod 0400 <your .pem file name>  
ssh -L localhost:8888:localhost:8888 -i <your .pem file name> ec2-  
user@<Your instance Public IP>  
  
#If you need to connect via proxy:  
ssh -o ProxyCommand='nc -x <your_proxy_address>:<your_proxy_port> <Your  
instance DNS> 22' -L localhost:8888:localhost:8888 -i <your .pem file  
name> ec2-user@<Your instance Public IP>
```

The screenshot shows a terminal window on Windows 10. The user is attempting to SSH into an instance. The command entered is 'ssh -L localhost:8888:localhost:8888 -i demo-key.pem ec2-user@<Your instance Public IP>'. The terminal then prompts for confirmation of the host key fingerprint. After accepting, it shows the user is connected to an 'Amazon Linux 2 AMI' instance. The session ends with a prompt '[ec2-user@ip-10-0-4-74 ~]\$'.

11. Connect to Jupyter Notebook.

Open a browser window and navigate to the URL given below. (NOTE: Replace text below in red.)

```
http://<your instance Public IP>:8888  
Password or token: <your EC2 instance-id>
```



12. Change Jupyter Notebook password.

Login to instance via SSH. (NOTE: Replace text below in red.)

```
pkill jupyter
```

```
Change password in '~/.start_jupyter.sh' and run `~/.start_jupyter.sh`
```

OR

```
jupyter notebook --no-browser --NotebookApp.allow_password_change=False  
--NotebookApp.token='<new password>' --ip 0.0.0.0 --port 8888 >  
/tmp/jupyter.out 2>&1 &
```

```
[ec2-user@ip-10-0-4-169 ~]$ pkill jupyter  
[ec2-user@ip-10-0-4-169 ~]$ cat .start_jupyter.sh  
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"  
INSTANCE_ID=`curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/instance-id'  
  
cd /home/ec2-user  
source /home/ec2-user/.bashrc  
#source /opt/intel/openvino_2021/bin/setupvars.sh  
  
export PATH=/opt/intel/openvino_2021/deployment_tools/model_optimizer:/opt/intel/openvino_2021/data_processing/gstreamer/bin:/opt/intel/openvino_2021/data_processing/gstreamer/bin/gstreamer-1.0:/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/ec2-user/.local/bin:/home/ec2-user/bin  
  
export LD_LIBRARY_PATH=/opt/intel/openvino_2021/data_processing/dl_streamer/lib:/opt/intel/openvino_2021/data_processing/gstreamer/lib:/opt/intel/openvino_2021/openvnc/lib:/opt/intel/openvino_2021/deployment_tools/ngraph/lib:/opt/intel/openvino_2021/deployment_tools/inference_engine/external/hddl/unite/lib:/opt/intel/openvino_2021/deployment_tools/inference_engine/external/hddl/lib:/opt/intel/openvino_2021/deployment_tools/inference_engine/external/gna/lib:/opt/intel/openvino_2021/deployment_tools/inference_engine/external/mkltiny_lnx/lib:/opt/intel/openvino_2021/deployment_tools/inference_engine/external/tbb/lib:/opt/intel/openvino_2021/deployment_tools/inference_engine/lib/intel64  
export PYTHONPATH=/opt/intel/openvino_2021/python/python3.7:/opt/intel/openvino_2021/python/python3:/opt/intel/openvino_2021/deployment_tools/open_model_zoo/tools/accuracy_checker:/opt/intel/openvino_2021/deployment_tools/model_optimizer:/opt/intel/openvino_2021/data_processing/dl_streamer/python:/opt/intel/openvino_2021/data_processing/gstreamer/lib/python3.6/site-packages:  
  
/home/ec2-user/.local/bin/jupyter notebook --no-browser --NotebookApp.allow_password_change=False --NotebookApp.token="$INSTANCE_ID" --ip 0.0.0.0 --port 8888 > /tmp/jupyter.out 2>&1 &  
  
[ec2-user@ip-10-0-4-169 ~]$ jupyter notebook --no-browser --NotebookApp.allow_password_change=False --NotebookApp.token='openvino@123' --ip 0.0.0.0 --port 8888 > /tmp/jupyter.out 2>&1 &  
[1] 4695
```

Tips: Stop the instance when not in use to prevent additional charges.

Create VPC with internet access

1. Navigate to VPC then click on Create VPC

The screenshot shows two overlapping AWS service pages. The top page is titled 'Your VPCs (5)' and includes a search bar and a 'Create VPC' button. The bottom page is titled 'Create VPC' and contains the following fields:

- VPC settings**
 - Name tag - *optional*: A text input field containing 'My_Demo'.
 - IPv4 CIDR block: A text input field containing '10.0.0.0/16'.
 - IPv6 CIDR block:
 - No IPv6 CIDR block
 - Amazon-provided IPv6 CIDR block
 - IPv6 CIDR owned by me
 - Tenancy: A dropdown menu set to 'Default'.
- Tags**
 - A section explaining what tags are and how they can be used.
 - A table for adding tags, showing one entry: 'Name' (Key) and 'My_Demo' (Value).
 - An 'Add new tag' button.
 - A note indicating 49 more tags can be added.

At the bottom right of the dialog are 'Cancel' and 'Create VPC' buttons.

Please fill in the following fields in the page and click “**Create VPC**”

- **Name tag – optional** → Demo_VPC (Example Name)
- **IPv4 CIDR block** → 10.0.0.0/16
- **IPv6 CIDR block** → No IPv6 CIDR block
- **Tenancy** → Default

2. Create an internet gateway with the following details.

The screenshot shows the AWS VPC Internet Gateways creation interface. At the top, there is a navigation bar with the AWS logo, services dropdown, and a search bar labeled "Filter internet gateways". On the right, there are buttons for "Actions" and "Create internet gateway". Below the navigation bar, the breadcrumb path is "VPC > Internet gateways > Create internet gateway". The main section is titled "Create internet gateway" with an "Info" link. A descriptive text explains that an internet gateway is a virtual router that connects a VPC to the internet. It asks to specify the name for the gateway below. There are two sections: "Internet gateway settings" and "Tags - optional". In the "Internet gateway settings" section, there is a "Name tag" field containing "Demo_IGW". In the "Tags - optional" section, there is a table with one row: "Key" is "Name" and "Value - optional" is "Demo_IGW". There is also a "Remove" button and an "Add new tag" button. A note says "You can add 49 more tags." At the bottom, there are "Cancel" and "Create internet gateway" buttons.

Please fill the following field in the page and click “**Create Internet gateway**”

- **Name tag → Demo_IGW (Example Name)**

3. Attach the created VPC to internet gateway

The following Internet gateway was created: Demo_IGW. You can now attach to a VPC to enable the VPC to communicate with the Internet.

VPC > Internet gateways > Demo_IGW / Demo_IGW

Actions ▾

VPC > Internet gateways > Attach to VPC (Demo_IGW) Info

VPC
Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

Available VPCs
Attach the internet gateway to this VPC.

AWS Command Line Interface command

Cancel Attach internet gateway

- Click “Attach to a VPC”
- Choose the VPC (Demo_VPC) that we created and click “Attach Internet Gateway”

4. Create Public Subnet

Subnets > Create subnet

Create subnet

Specify your subnet's IP address block in CIDR format; for example, 10.0.0.0/24. IPv4 block sizes must be between a /16 netmask and /28 netmask, and can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag: Public_Demo_Subnet

VPC*: Demo_VPC

Availability Zone: No preference

VPC CIDRs	CIDR	Status	Status Reason
10.0.0.0/16		associated	

IPv4 CIDR block*: 10.0.4.0/24

* Required

Cancel Create

Please fill the following field in the page & click “Create”

- **Name tag** → Public_Demo_Subnet (Example Name)
- **VPC*** → Choose the VPC (Demo_VPC) that we created
- **Availability Zone** → us-east-1a (Choose your respective region)
- **IPv4 CIDR block*** → 10.0.4.0/24

5. Create Route Table

The screenshot shows the 'Create route table' wizard. At the top, it says 'Route Tables > Create route table'. The main section is titled 'Create route table' with a sub-instruction: 'A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.' Below this, there are fields for 'Name tag' (set to 'Public_Demo_RT') and 'VPC*' (a dropdown menu showing a single option). A 'Tags' section follows, with a table header 'Key (128 characters maximum) | Value (256 characters maximum)'. A note below says 'This resource currently has no tags'. There is a 'Add Tag' button and a note '50 remaining (Up to 50 tags maximum)'. At the bottom right are 'Cancel' and 'Create' buttons.

Please fill the following field in the page & click “**Create**”

- **Name tag** → Public_Demo_RT (Example Name)
- **VPC*** → Choose the VPC (Demo_VPC) that we created

6. Attach the created internet gateway in the route table

The screenshot shows the 'Routes' tab of a route table's details page. The table has columns: Destination, Target, Status, and Propagated. One row is present: Destination 10.0.0.0/16, Target local, Status active, Propagated No.

Destination	Target	Status	Propagated
10.0.0.0/16	local	active	No

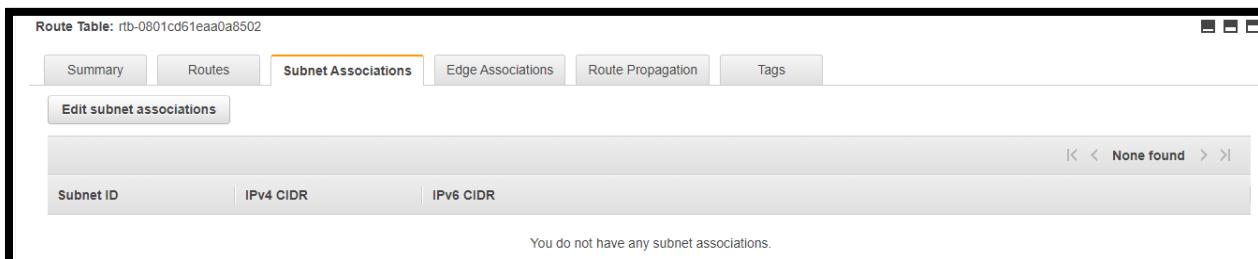
Click “Edit routes” and add a route with following details & click “Save routes”

- **Destination** → 0.0.0.0/0 (Any)
- **Target** → Choose your Internet Gateway(Demo_IGW)

The screenshot shows the 'Edit routes' wizard. It has a table with columns: Destination, Target, Status, and Propagated. Two rows are shown: one for 10.0.0.0/16 with Target 'local' and another for 0.0.0.0/0 with Target 'igw-0059ba23969ff2e9'. Both rows have 'Status' set to 'active' and 'Propagated' set to 'No'. At the bottom left is an 'Add route' button, and at the bottom right are 'Cancel' and 'Save routes' buttons.

7. Attach the created public subnet in the route table

Click “Edit subnet associations” and choose following details and click “Save”.

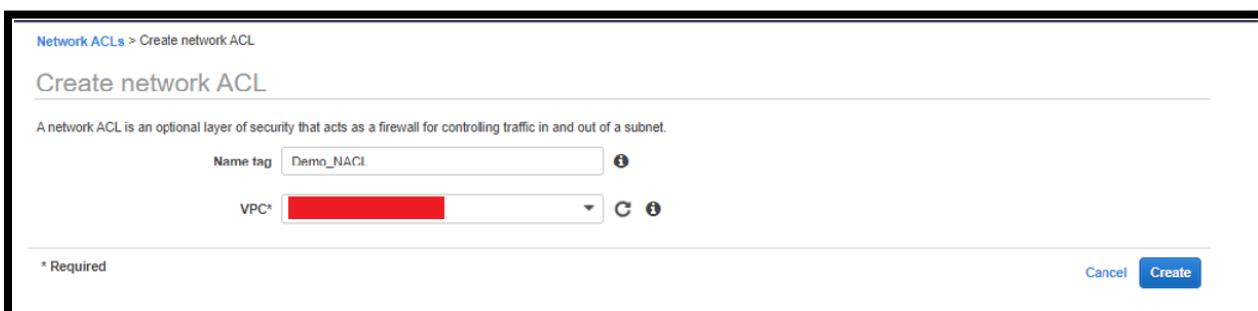


Subnet ID → Choose the created Public Subnet(Public_Demo_Subnet)



8. Create Network ACL

Please fill the following field in the page and click “Create”



- **Name tag → Demo_NACL (Example Name)**
- **VPC* → Choose the VPC (Demo_VPC) that we created**

9. Click Edit inbound rules and add the following rules.

Network ACL: acl-0484d27fee71d5c00

Inbound Rules

View All rules

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY

Network ACLs > Edit inbound rules

Edit inbound rules

Network ACL acl-0484d27fee71d5c00

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
1	SSH (22)	TCP (6)	22	10.10.10.32	ALLOW
2	Custom TCP Rule	TCP (6)	8888	10.10.10.32	ALLOW

Add Rule

* Required

Cancel Save

Inbound rules

Type	Port	Source	Allow/Deny
SSH	22	Your System IP	Allow
Custom TCP Rule	8888	Your System IP	Allow

Create Security Group

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a new security group Select an existing security group

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
Custom TCP	TCP	8888	Custom CIDR, IP or Security Group	e.g. SSH for Admin Desktop

Add Rule

Warning
Rules with source of 0.0.0.0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Cancel Previous Review and Launch

Please fill the following fields in the page & click “**Review and Launch**”

- **Security group name** → Demo_SG (Example Name)
- **Description** → Allow SSH access to developers (Example Description)
- **Inbound rules**

Type	Port	Source
SSH	22	Custom (Your System IP)
Custom TCP	8888	Custom (Your System IP)



Notices and Disclaimers

[FTC Optimization Notice](#)

Intel technologies may require enabled hardware, software or service activation.

No product or component can be absolutely secure.

Your costs and results may vary.

© 2020 Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.