aws

Developer Guide

AWS Wavelength

AWS Wavelength: Developer Guide

Table of Contents

What is AWS Wavelength?

AWS Wavelength enables developers to build applications that require edge computing infrastructure to deliver low latency to mobile devices and end users or increase the resiliency of their existing edge applications. Wavelength deploys standard AWS compute and storage services to the edge of communications service providers' (CSP) networks. You can extend a virtual private cloud (VPC) to one or more Wavelength Zones. You can then use AWS resources such as Amazon Elastic Compute Cloud (Amazon EC2) instances to run the applications that require low latency or edge resiliency within the Wavelength Zone, while seamlessly communicating back to your existing AWS services deployed in the Region.

For more information, see AWS Wavelength.

Wavelength concepts

The following are the key concepts:

- **Wavelength** — A new type of AWS infrastructure designed to run workloads that require low latency or edge resiliency.
- **Wavelength Zone** — A zone in the carrier location where the Wavelength infrastructure is deployed. Wavelength Zones are associated with an AWS Region. A Wavelength Zone is a logical extension of the Region, and is managed by the control plane in the Region.
- **VPC** — A customer virtual private cloud (VPC) that spans Availability Zones, Local Zones, and Wavelength Zones, and has deployed resources such as Amazon EC2 instances in the subnets that are associated with the zones.
- **Wavelength subnet** — A subnet that you create in a Wavelength Zone. You can create one or more subnets, and then run and manage AWS services, such as Amazon EC2 instances, in the subnet.
- **Carrier gateway** — A carrier gateway serves two purposes. It allows inbound traffic from a carrier network in a specific location, and allows outbound traffic to the carrier network and internet.
- **Network Border Group** — A unique set of Availability Zones, Local Zones, or Wavelength Zones from which AWS advertises IP addresses.
- **Wavelength application** — An application that you run on an AWS resource in a Wavelength Zone.

AWS resources on Wavelength

You can create Amazon EC2 instances, Amazon EBS volumes, and Amazon VPC subnets and carrier gateways in Wavelength Zones. You can also use the following:

- Amazon EC2 Auto Scaling
- Amazon EKS clusters
- Amazon ECS clusters
- Amazon EC2 Systems Manager
- Amazon CloudWatch
- AWS CloudTrail
- AWS CloudFormation
- Application Load Balancer in select Wavelength Zones. For a list of these Zones, see Load balancing.

The services in Wavelength are part of a VPC that is connected over a reliable connection to an AWS Region for easy access to services running in Regional subnets.

Working with Wavelength

You can create, access, and manage your EC2 resources, Wavelength Zones, and carrier gateways using any of the following interfaces:

- **AWS Management Console**— Provides a web interface that you can use to access your Wavelength resources.
- **AWS Command Line Interface (AWS CLI)** — Provides commands for a broad set of AWS services, including Amazon VPC, and is supported on Windows, macOS, and Linux. The services you use in Wavelength continue to use their own namespace, for example Amazon EC2 uses the "ec2" namespace, and Amazon EBS uses the "ebs" namespace. For more information, see AWS Command Line Interface.
- **AWS SDKs** — Provides language-specific APIs and takes care of many of the connection details, such as calculating signatures, handling request retries, and handling errors. For more information, see AWS SDKs.

When you use any of the interfaces for your Wavelength Zones, use the parent Region.

Pricing

For more information, see AWS Wavelength Pricing.

Use cases for AWS Wavelength

Using AWS Wavelength Zones can help you accomplish a variety of goals. This section lists a few to give you an idea of the possibilities.

Online betting and regulated industries

AWS Wavelength provides edge resiliency to help address data residency requirements for regulated industries, such as online sports betting. Using a combination of AWS Wavelength alongside existing AWS hybrid and edge services such as AWS Outposts or AWS Local Zones, you can create highly-available architectures within state or country borders.

Media and entertainment

Wavelength provides the low latency needed to live stream high-resolution video and high-fidelity audio, and to embed interactive experiences into live video streams. Real-time video analytics provide the ability to generate real-time statistics that enhance the live event experience.

Healthcare

Using AWS Wavelength, medical training providers can offer mobile games, medical simulations for rare disease diagnosis, advanced endoscopic maneuvers, ultrasound equipment and much more. Using AWS Wavelength to host the remote rendering engine, doctors can experience an immersive training experience without procuring the often-required expensive equipment to do so.

Augmented reality (AR) and virtual reality (VR)

By accessing compute resources on AWS Wavelength, AR/VR applications can reduce the Motion to Photon (MTP) latencies to the benchmark that is needed to offer a realistic customer experience. When you use AWS Wavelength, you can offer AR/VR in locations where it is not possible to run local system servers.

Connected vehicles

Cellular Vehicle-to-Everything (C-V2X) is an increasingly important platform for enabling functionality such as intelligent driving, real-time HD maps, and increased road safety. Low latency

access to the compute infrastructure that's needed to run data processing and analytics on AWS Wavelength enables real-time monitoring of data from sensors on the vehicle. This allows for secure connectivity, in-car telematics, and autonomous driving.

Smart factories

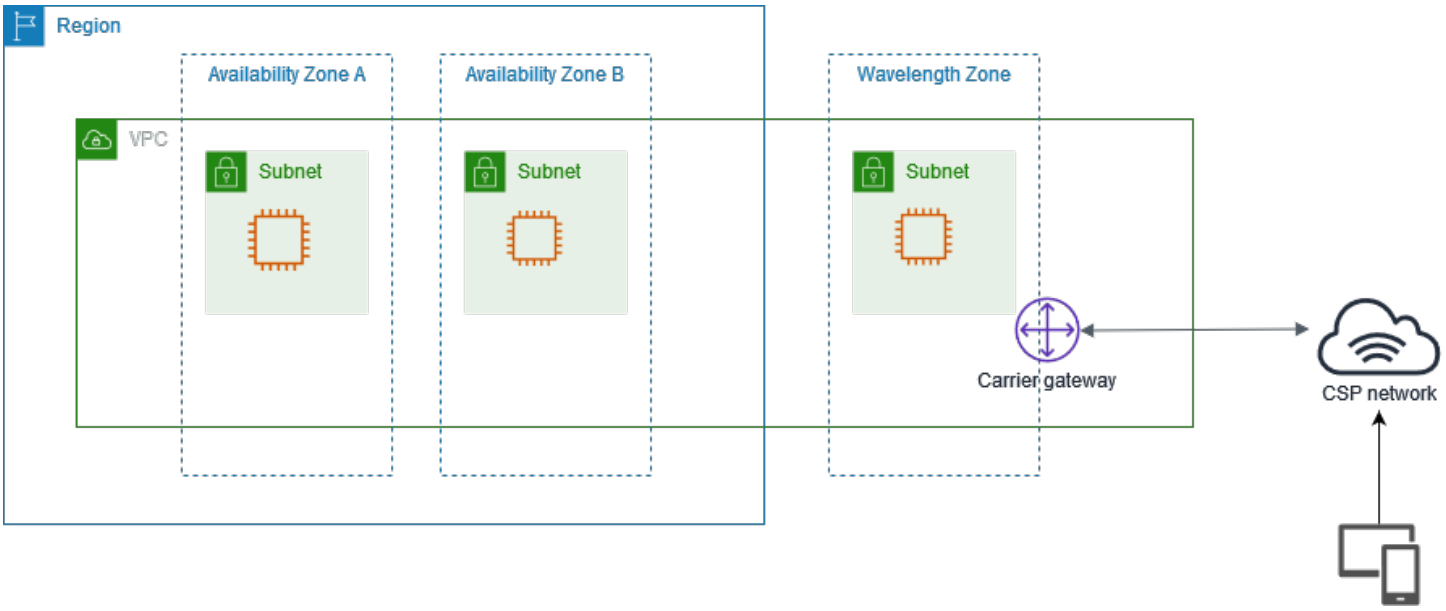
Industrial automation applications use ML inference at the edge to analyze images and videos to detect quality issues on fast moving assembly lines and to trigger actions that address the issues. With AWS Wavelength, these applications can be deployed without having to use expensive, GPU-based servers on the factory floor.

Real-time gaming

Real-time game streaming depends on low latency to preserve the user experience. With AWS Wavelength, you can stream the most demanding games from Wavelength Zones so that they are available on end devices that have limited processing power.

How AWS Wavelength works

The following diagram demonstrates how you can create a subnet that uses resources in a communications service provider (CSP) network at a specific location. For resources that must be deployed to the Wavelength Zone, first opt in to the Wavelength Zone, and then create resources in the Wavelength Zone.



Contents

VPCs

After you create a VPC in a Region, create a subnet in a Wavelength Zone that is associated with the VPC. In addition to the Wavelength Zone, you can create resources in all of the Availability Zones and Local Zones that are associated with the VPC.

You have control over the VPC networking components, such as IP address assignment, subnets, and route table creation.

VPCs that contain a subnet in a Wavelength Zone can connect to a carrier gateway. A carrier gateway allows you to connect to the following resources:

- 4G/LTE and 5G devices on the telecommunication carrier network
- Fixed wireless access for select Wavelength Zone partners. For more information, see Multi-access AWS Wavelength.
- Outbound traffic to public internet resources

Subnets

Any subnet that you create in a Wavelength Zone inherits the main VPC route table, which includes the local route. The local route enables connectivity between the subnets in the VPC, including the subnets that are in the Wavelength Zone.

AWS recommends that you configure custom route tables for your subnets in Wavelength Zones. The destinations are the same destinations as a subnet in an Availability Zone or Local Zone, with the addition of a carrier gateway. For more information, see the section called “Routing”.

Carrier gateways

A carrier gateway serves two purposes. It allows inbound traffic from a carrier network in a specific location, and it allows outbound traffic to the carrier network and internet. There is no inbound connection configuration from the internet to a Wavelength Zone through the carrier gateway.

A carrier gateway supports IPv4 traffic.

Carrier gateways are only available for VPCs that contain subnets in a Wavelength Zone. The carrier gateway provides connectivity between your Wavelength Zone and the telecommunication carrier, and devices on the telecommunication carrier network. The carrier gateway performs NAT of the Wavelength instances' IP addresses to the Carrier IP addresses from a pool that is assigned to the network border group. The carrier gateway NAT function is similar to how an internet gateway functions in a Region.

Carrier IP address

A *Carrier IP address* is the address that you assign to a network interface, which resides in a subnet in a Wavelength Zone (for example an EC2 instance). The carrier gateway uses the address for traffic from the interface to the internet or to mobile devices. The carrier gateway uses NAT to translate the address, and then sends the traffic to the destination. Traffic from the telecommunication carrier network routes through the carrier gateway.

You allocate a Carrier IP address from a network border group, which is a unique set of Availability Zones, Local Zones, or Wavelength Zones from which AWS advertises IP addresses, for example, `us-east-1-wl1-bos-wlz-1`.

Routing

You can set the carrier gateway as a destination in a route table for the following resources:

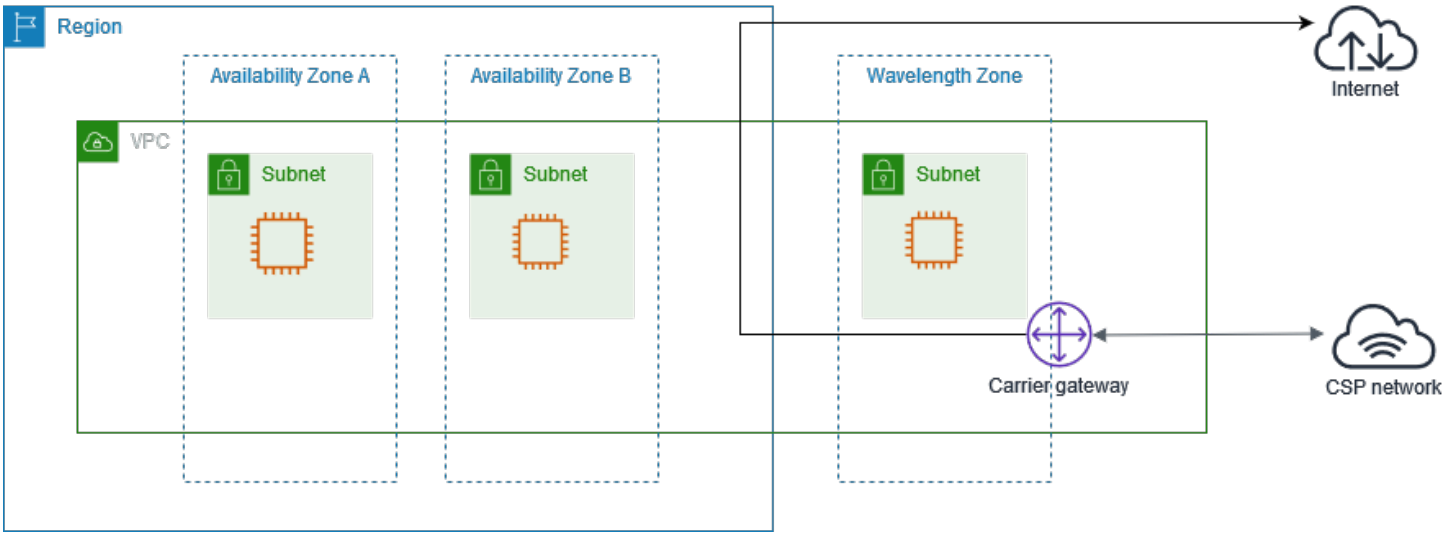
- VPCs that contain subnets in a Wavelength Zone
- Subnets in Wavelength Zones

Create a custom route table for the subnets in the Wavelength Zones so that the default route goes to the carrier gateway, which then sends traffic to the internet and telecommunication carrier network.

Example: Carrier gateway routing to the public internet

Consider a scenario with the following configuration:

- A VPC with Availability Zones and a Wavelength Zone
- A subnet in the Wavelength Zone
- An EC2 instance in the subnet in the Wavelength Zone
- A Carrier IP address for the network interface associated with the EC2 instance
- An IP address association that maps the private IP address of the EC2 instance to the Carrier IP address



You need the following entries in the Wavelength subnet route table.

Destination	Target	Notes
VPC CIDR	Local	This route allows for intra-VPC connectivity, including subnets in the Availability Zones.
0.0.0.0/0	*carrier-gateway-id*	The Carrier IP address provides internet connectivity through the carrier gateway.

Carrier gateway access to the public internet

The carrier gateway provides access to the internet from your Wavelength subnets. For information about protocol considerations, see the section called “Networking considerations”.

Traffic initiated from the EC2 instance for the internet uses the 0.0.0.0/0 route to route traffic to the carrier gateway. The carrier gateway maps the EC2 instance IP address to the Carrier IP address, and then sends the traffic to the telecommunication carrier.

DNS

EC2 instances use EC2 DNS to resolve domain names to IP addresses. Route 53 supports DNS features, such as domain registration, and DNS routing. Both public and private hosted Wavelength Zones are supported for routing traffic to specific domains. Route 53 resolvers are hosted in the Region.

You can also use your own DNS services to resolve domain names.

Maximum Transmission Unit

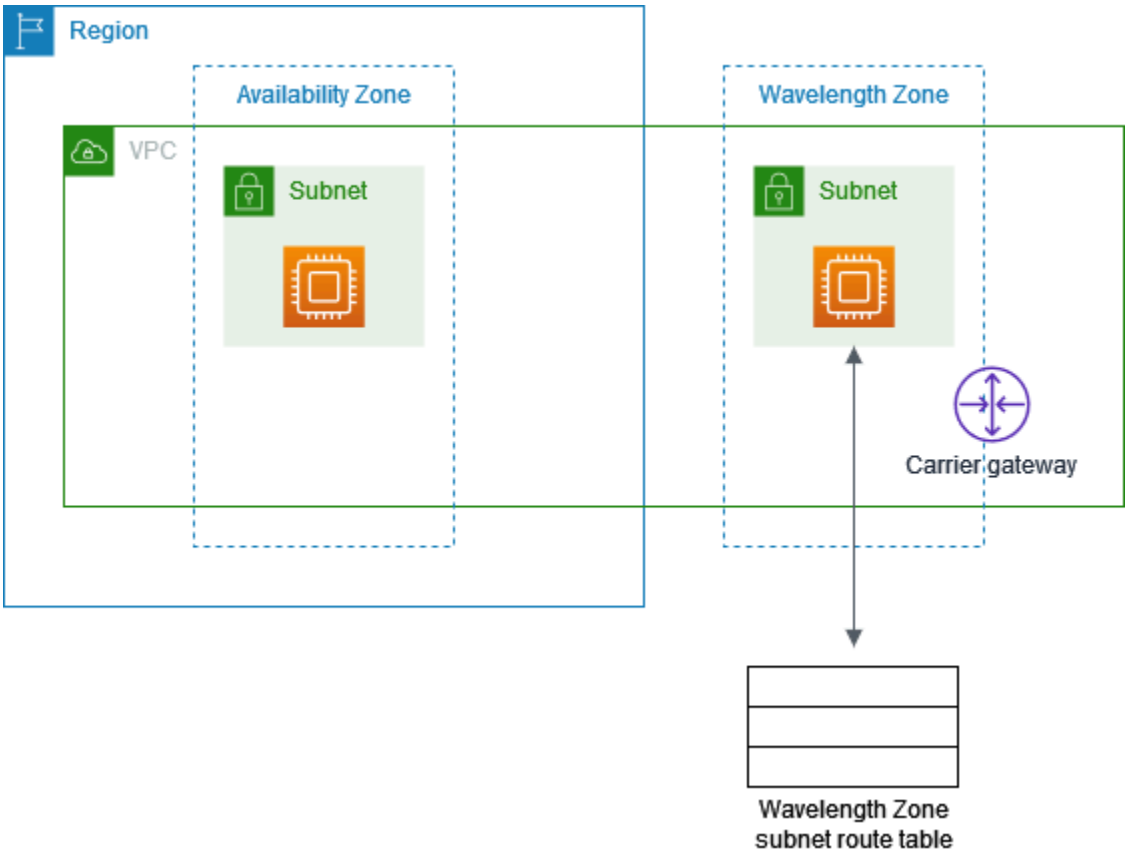
Generally, the Maximum Transmission Unit (MTU) is as follows:

- 9001 bytes between Amazon EC2 instances in the same Wavelength Zone.
- 1500 bytes between carrier gateway and a Wavelength Zone.
- 1300 bytes between an Amazon EC2 instance in a Wavelength Zone and an Amazon EC2 instance in the Region.

Get started with AWS Wavelength

The following diagram shows the resources that you need to configure to get started using AWS Wavelength.

- A VPC in your Region
- A carrier gateway
- A public subnet in an Availability Zone in your Region
- An instance in the public subnet
- An instance in the Wavelength Zone subnet with a Carrier IP address



Tasks

- Step 1: Opt in to Wavelength Zones
- Step 2: Configure your network
- Step 3: Launch an instance in your Availability Zone public subnet

- Step 4: Launch an instance in the Wavelength zone
- Step 5: Test the connectivity

Step 1: Opt in to Wavelength Zones

Before you specify a Wavelength Zone for a resource or service, you must opt in to the zone.

Prerequisites

- Some AWS resources are not available in all Regions. Make sure that you can create the resources that you need in the desired Region or Wavelength Zone before launching an instance in a specific Wavelength Zone.
- Before you begin, review *Quotas and considerations*, which includes information about available Wavelength Zones, service differences, and Service Quotas. You should also speak with your mobile operator about mobile service plans and any additional requirements.

To opt in to Wavelength Zone using the console

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. From the Region selector in the navigation bar, select the Region for the Wavelength Zone.
3. On the navigation pane, choose **EC2 Dashboard**.
4. In the upper-right corner of the page, choose **Account attributes**, **Zones**.
5. Under **Wavelength Zones**, choose **Manage**.
6. Choose **Enabled**.
7. Choose **Update zone group**.

To enable Wavelength Zones using the AWS CLI

Alternatively, use the AWS CLI to enable Wavelength Zones. To do so, use the modify-availability-zone-group command.

Step 2: Configure your network

After you opt in to the Wavelength Zone, create a VPC, a carrier gateway, and a public subnet in the Availability Zone.

Tasks

- Create a VPC
- Create a carrier gateway and a subnet associated with the Wavelength Zone
- Create a public subnet in an Availability Zone

Create a VPC

Create a VPC to extend to your Wavelength Zone.

To create a VPC using the console

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. Choose **Create VPC**.
3. For **Resources to create**, choose **VPC only**.
4. For **Name tag**, optionally provide a name for your VPC. Doing so creates the tag `Name=`*value*.
5. For **IPv4 CIDR block**, specify an IPv4 CIDR block for the VPC. We recommend that you specify a CIDR block from the private (non-publicly routable) IP address ranges as specified in RFC 1918; for example, `10.0.0.0/16`, or `192.168.0.0/16`.

   > **Note**
   >
   > You can specify a range of publicly routable IPv4 addresses. However, we currently do not support direct access to the internet from publicly routable CIDR blocks in a VPC. Windows instances cannot boot correctly if launched into a VPC with ranges from `224.0.0.0` to `255.255.255.255` (Class D and Class E IP address ranges).

6. Choose **Create VPC**.

Create a carrier gateway and a subnet associated with the Wavelength Zone

After you create a VPC, create a carrier gateway, and then select the subnets that route traffic to the carrier gateway.

When you choose to automatically route traffic from subnets to the carrier gateway, we create the following resources:

- A carrier gateway
- A subnet. You can optionally assign all carrier gateway tags except the `Name` tag to the subnet.
- A network ACL with the following resources:
  - A subnet association with the subnet in the Wavelength Zone
  - Default inbound and outbound rules for your traffic.
- A route table with the following resources:
  - A route for local traffic
  - A route that routes non-local traffic to the carrier gateway
  - An association with the subnet

To create a carrier gateway

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Carrier gateways**, and then choose **Create carrier gateway**.
3. (Optional) For **Name**, enter a name for the carrier gateway.
4. For **VPC**, choose the VPC.
5. Choose **Route subnet traffic to carrier gateway**, and under **Subnets to route** do the following:
   a. Under **Existing subnets in Wavelength Zone**, select the box for each Wavelength subnet to route to the carrier gateway.
   b. To create a subnet in the Wavelength Zone, choose **Add new subnet**, enter the required information, and then choose **Add new subnet**.
6. (Optional) To add a tag to the carrier gateway, choose **Add tag**, and then enter the tag key and tag value.
7. Choose **Create carrier gateway**.

Create a public subnet in an Availability Zone

Create a subnet in an Availability Zone in the Region.

To add a subnet

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Subnets**.

3. Choose **Create subnet**.
4. For **VPC**, choose the VPC.
5. For **Subnet name**, provide a name for the subnet. Doing so creates the tag `Name=value`.
6. For **Availability Zone**, chose an Availability Zone, or choose **No Preference** to have AWS choose one for you.
7. For **IPv4 CIDR block**, specify an IPv4 address range for your subnet, using CIDR notation.
8. Choose **Create subnet**.

Step 3: Launch an instance in your Availability Zone public subnet

Launch an EC2 instance in the subnet that you created in the Availability Zone. You will use this instance to test the connectivity from the Region to the Wavelength Zone.

You can launch EC2 instances in the public subnet that you created. For information about how to launch an instance in the Amazon EC2 console, see one of the following guides:

- For Linux instances, see Launch your instance in the *Amazon EC2 User Guide*.
- For Windows instances, see Launch your instance in the *Amazon EC2 User Guide*.

Step 4: Launch an instance in the Wavelength zone

After you complete the networking configuration, launch an instance, and then allocate a Carrier IP address for the instance.

Options

- Option 1: Auto assign a Carrier IP address
- Option 2: Allocate and associate a Carrier IP address from the network border group

Option 1: Auto assign a Carrier IP address

AWS recommends that you use the AWS CLI because you can automatically allocate and associate the Carrier IP address with the network interface.

Use the run-instances command as follows to launch an instance in the Wavelength Zone subnet.

```
aws ec2 run-instances --region us-east-1 --network-interfaces
 "DeviceIndex=0,AssociateCarrierIpAddress=true,SubnetId=subnet-036aa298f4EXAMPLE" --
image-id ami-04125ecea1EXAMPLE --instance-type t3.medium
```

- `DeviceIndex` – Specify 0 to indicate the primary network interface (eth0).
- `SubnetId` – Specify the ID of the subnet in the Wavelength Zone.
- `AssociateCarrierIpAddress` – Set this value to `true` to assign a Carrier IP address to the network interface.

Option 2: Allocate and associate a Carrier IP address from the network border group

You can launch EC2 instances in the subnet that you created when you added the carrier gateway. For more information, see the section called "Create a carrier gateway and a subnet associated with the Wavelength Zone". Security groups control inbound and outbound traffic for instances in a subnet, just as they do for instances in an Availability Zone subnet. To connect to an EC2 instance in a subnet, specify a key pair when you launch the instance, just as you do for instances in an Availability Zone subnet. For information about how to launch an instance in the Amazon EC2 console, see one of the following guides:

- For Linux instances, see Launch your instance in the *Amazon EC2 User Guide*.
- For Windows instances, see Launch your instance in the *Amazon EC2 User Guide*.

To allocate and associate a Carrier IP address

1. Use the allocate-address command as follows to allocate a Carrier IP address.

   ```
   aws ec2 allocate-address --region us-east-1 --domain vpc --network-border-group us-
   east-1-wl1-bos-wlz-1
   ```

   The following is example output.

   ```
   {
       "AllocationId": "eipalloc-05807b62acEXAMPLE",
       "PublicIpv4Pool": "amazon",
       "NetworkBorderGroup": "us-east-1-wl1-bos-wlz-1",
       "Domain": "vpc",
   ```

```
    "CarrierIp": "155.146.10.111"

}
```

2. Use the associate-address command as follows to associate the Carrier IP address with the EC2 instance.

```
aws ec2 associate-address --allocation-id eipalloc-05807b62acEXAMPLE --network-
interface-id eni-1a2b3c4d
```

The following is example output.

```
{
    "AssociationId": "eipassoc-02463d08ceEXAMPLE",
}
```

Step 5: Test the connectivity

Before you test the connectivity, do the following:

- Review the section called “Networking considerations”
- Configure the instance security group to allow ICMP traffic.

Test the connectivity from the instance in the Region to the Wavelength Zone instance. Depending on your operating system, use SSH or RDP to connect to the Carrier IP address of your Region instance. You can use a secure bastion host.

Run the ping command to the Wavelength Zone instance. In the following example, the IP address of the subnet in the Wavelength Zone is 10.0.3.112.

```
ping 10.0.3.112
Pinging 10.0.3.112
Reply from 10.0.3.112:  bytes=32 time=<1ms TTL=128
Reply from 10.0.3.112:  bytes=32 time=<1ms TTL=128
Reply from 10.0.3.112:  bytes=32 time=<1ms TTL=128

Ping statistics for 10.0.3.112
Packets:  Sent = 3,  Received = 3,  Lost = 0 (0% lost)
```

```
Approximate round trip time in milliseconds
Minimum = 0ms,  Maximum = 0ms,  Average = 0ms
```

Test the connectivity from the instance in the Wavelength Zone instance to the carrier network. Depending on your operating system, use SSH or RDP to connect to the Carrier IP address of your Wavelength Zone instance. You can use a secure bastion host.

You need a device on the carrier network in order to test the connectivity from the Wavelength Zone to the carrier network. In addition, Headspin, which is part of the AWS Partner Network, provides devices on carrier networks for functional testing. For more information, see Headspin.

Run the **ping** command to an address in the carrier network. In the following example, the carrier network IP address is 198.51.100.130.

```
ping 198.51.100.130
Pinging 198.51.100.130
Reply from 198.51.100.130:  bytes=32 time=<1ms TTL=128
Reply from 198.51.100.130:  bytes=32 time=<1ms TTL=128
Reply from 198.51.100.130:  bytes=32 time=<1ms TTL=128

Ping statistics for 198.51.100.130
Packets:  Sent = 3,  Received = 3,  Lost = 0 (0% lost)

Approximate round trip time in milliseconds
Minimum = 0ms,  Maximum = 0ms,  Average = 0ms
```

What is a carrier gateway?

A carrier gateway serves two purposes. It allows inbound traffic from a carrier network in a specific location, and it allows outbound traffic to the carrier network and the internet. There is no inbound connection configuration from the internet to a Wavelength Zone through the carrier gateway.

A carrier gateway supports IPv4 traffic.

Carrier gateways are only available for VPCs that contain subnets in a Wavelength Zone. The carrier gateway provides connectivity between your Wavelength Zone and the carrier, and devices on the carrier network. The carrier gateway performs NAT of the Wavelength instances' IP addresses to the Carrier IP addresses from a pool that is assigned to the network border group. The carrier gateway NAT function is similar to how an internet gateway functions in a Region.

Enable access to the carrier network

To enable access to or from the carrier network for instances in a Wavelength subnet, you must do the following:

- Create a VPC.
- Create a carrier gateway and attach the carrier gateway to your VPC. When you create the carrier gateway, you can optionally choose which subnets route to the carrier gateway. When you select this option, we automatically create the resources related to carrier gateways, such as route tables and network ACLs. If you do not choose this option, then you must perform the following tasks:
  - Select the subnets that route traffic to the carrier gateway.
  - Ensure that your subnet route tables have a route that directs traffic to the carrier gateway.
  - Ensure that instances in your subnet have a globally unique Carrier IP address.
  - Ensure that your network access control lists and security group rules allow the relevant traffic to flow to and from your instance.

Work with carrier gateways

The following sections describe how to manually create a carrier gateway for your VPC to support inbound traffic from the carrier network (for example, mobile phones), and to support outbound traffic to the carrier network and the internet.

Tasks

- Create a VPC
- Create a carrier gateway
- Create a security group to access the carrier network
- Allocate and associate a Carrier IP address with the instance in the Wavelength Zone subnet
- Routing to a Wavelength Zone carrier gateway
- View the carrier gateway details
- Manage carrier gateway tags
- Delete a carrier gateway

Create a VPC

You can create an empty Wavelength VPC as follows.

Limitation

You can specify a range of publicly routable IPv4 addresses. However, we do not support direct access to the internet from publicly routable CIDR blocks in a VPC. Windows instances cannot boot correctly if launched into a VPC with ranges from `224.0.0.0` to `255.255.255.255` (Class D and Class E IP address ranges).

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Your VPCs**, **Create VPC**.
3. Do the following and then choose **Create**.
   - **Name tag**: Optionally provide a name for your VPC. Doing so creates a tag with a key of `Name` and the value that you specify.
   - **IPv4 CIDR block**: Specify an IPv4 CIDR block for the VPC. We recommend that you specify a CIDR block from the private (non-publicly routable) IP address ranges as specified in RFC 1918; for example, `10.0.0.0/16`, or `192.168.0.0/16`.

To create a VPC using the AWS CLI

Use the create-vpc command.

Create a carrier gateway

After you create a VPC, create a carrier gateway and then select the subnets that route traffic to the carrier gateway.

If you have not opted in to a Wavelength Zone, the Amazon Virtual Private Cloud Console prompts you to opt in. For more information, see the section called “Manage Zones”.

When you choose to automatically route traffic from subnets to the carrier gateway, we create the following resources:

- A carrier gateway
- A subnet. You can optionally assign all carrier gateway tags that do not have a **Key** value of `Name` to the subnet.
- A network ACL with the following resources:
  - A subnet associated with the subnet in the Wavelength Zone
  - Default inbound and outbound rules for all of your traffic.
- A route table with the following resources:
  - A route for all local traffic
  - A route that routes all non-local traffic to the carrier gateway
  - An association with the subnet

To create a carrier gateway

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Carrier Gateways**, and then choose **Create carrier gateway**.
3. Optional: For **Name**, enter a name for the carrier gateway.
4. For **VPC**, choose the VPC.
5. Choose **Route subnet traffic to carrier gateway**, and under **Subnets to route** do the following.
   a. Under **Existing subnets in Wavelength Zone**, select the box for each subnet to route to the carrier gateway.
   b. To create a subnet in the Wavelength Zone, choose **Add new subnet**, specify the following information, and then choose **Add new subnet**:

- **Name tag**: Optionally provide a name for your subnet. Doing so creates a tag with a key of `Name` and the value that you specify.
- **VPC**: Choose the VPC.
- **Availability Zone**: Choose the Wavelength Zone.
- **IPv4 CIDR block**: Specify an IPv4 CIDR block for your subnet, for example, `10.0.1.0/24`.
- To apply the carrier gateway tags to the subnet, select **Apply same tags from this carrier gateway**.

6. (Optional) To add a tag to the carrier gateway, choose **Add tag**, and then do the following:
   - For **Key**, enter the key name.
   - For **Value**, enter the key value.
7. Choose **Create carrier gateway**.

To create a carrier gateway using the AWS CLI

1. Use the create-carrier-gateway command.
2. Add a VPC route table with the following resources:
   - A route for all VPC local traffic
   - A route that routes all non-local traffic to the carrier gateway
   - An association with the subnets in the Wavelength Zone

   For more information, see the section called “Routing to a Wavelength Zone carrier gateway”.

Create a security group to access the carrier network

By default, a VPC security group allows all outbound traffic. You can create a new security group and add rules that allow inbound traffic from the carrier. Then, you associate the security group with instances in the subnet.

To create a new security group and associate it with your instances

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Security Groups**, and then choose **Create Security Group**.

3. To create a security group, choose **Create security group**, specify the following information, and then choose **create**:

   - **Security group name**: Enter a name for the subnet.
   - **Description**: Enter the security group description.
   - **VPC**: Choose the VPC.
4. Select the security group. The details pane displays the details for the security group, plus tabs for working with its inbound rules and outbound rules.
5. On the **Inbound Rules** tab, choose **Edit**. Choose **Add Rule**, and complete the required information. For example, select **HTTP** or **HTTPS** from the **Type** list, and enter the **Source** as `0.0.0.0/0` for IPv4 traffic, or `::/0` for IPv6 traffic. Choose **Save**.
6. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
7. In the navigation pane, choose **Instances**.
8. Select the instance, choose **Actions**, **Networking**, and then select **Change Security Groups**.
9. Clear the check box for the currently selected security group, and then select the new one. Choose **Assign Security Groups**.

To create a security group using the AWS CLI

Use the create-security-group command.

Allocate and associate a Carrier IP address with the instance in the Wavelength Zone subnet

If you used the Amazon EC2 console to launch the instance, or you did not use the `associate-carrier-ip-address` option in the AWS CLI, then you must allocate a Carrier IP address and assign it to the instance:

To allocate and associate a Carrier IP address using the AWS CLI

1. Use the allocate-address command as follows.

   ```
   aws ec2 allocate-address --region us-east-1 --domain vpc --network-border-group us-east-1-wl1-bos-wlz-1
   ```

   The following is example output:

```
{
    "AllocationId": "eipalloc-05807b62acEXAMPLE",
    "PublicIpv4Pool": "amazon",
    "NetworkBorderGroup": "us-east-1-wl1-bos-wlz-1",
    "Domain": "vpc",
    "CarrierIp": "155.146.10.111"

}
```

2. Use the associate-address command to associate the Carrier IP address with the EC2 instance as follows.

```
aws ec2 associate-address --allocation-id eipalloc-05807b62acEXAMPLE --network-
interface-id eni-1a2b3c4d
```

The following is example output:

```
{
    "AssociationId": "eipassoc-02463d08ceEXAMPLE",
}
```

Routing to a Wavelength Zone carrier gateway

Subnets that are in Wavelength Zones can have an additional target type of a carrier gateway. Consider the case where you want to have the carrier gateway route traffic to route all non-VPC traffic to the carrier network. To do this, create and attach a carrier gateway to your VPC, and then add the following routes:

Destination	Target
0.0.0.0/0	*cagw-id*
::/0	*cagw-id*

View the carrier gateway details

You can view information about your carrier gateway, including the state and the tags.

To view the carrier gateway details

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Carrier Gateways**.
3. Select the carrier gateway and choose **Actions**, **View details**.

To view the carrier gateway details using the AWS CLI

Use the describe-carrier-gateways command.

Manage carrier gateway tags

Tags help you to identify your carrier gateways. You can add or remove tags.

To manage the carrier gateway tags

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Carrier Gateways**.
3. Select the carrier gateway and choose **Actions**, **Manage tags**.
4. To add a tag, choose **Add tag**, and then do the following:
   - For **Key**, enter the key name.
   - For **Value**, enter the key value.
5. To remove a tag, choose **Remove** to the right of the tag's Key and Value.
6. Choose **Save**.

To manage the carrier gateway tags using the AWS CLI

- To add tags, use the create-tag command.
- To delete tags, use the delete-tags command.

Delete a carrier gateway

If you no longer need a carrier gateway, you can delete it.

> **Important**
>
> If you do not delete the route that has the carrier gateway as the **Target**, the route is a blackhole route.

To delete a carrier gateway

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Carrier Gateways**.
3. Select the carrier gateway and choose **Actions**, **Delete carrier gateway**.
4. In the **Delete carrier gateway** dialog box, enter **Delete**, and then choose **Delete**.

To delete a carrier gateway using the AWS CLI

Use the delete-carrier-gateway command.

Manage Zones

Before you specify a Wavelength Zone for a resource or service, you must opt in to the zone.

Multi-access AWS Wavelength

TCP, UDP, and ICMP traffic from the device on the carrier network to an Amazon EC2 instance in the Wavelength Zone is supported. However, when a mobile subscriber connects to an external, non-cellular network offered by the communications service provider (CSP), such as a WiFi network, traffic is denied because traffic is characterized as *internet facing*.

With the proliferation of high-speed 5G networks, CSPs now offer new connectivity solutions to residential, small-business, and enterprise customers such as Fixed Wireless Access (FWA).

The following CSP partners that offer AWS Wavelength Zones have expanded the available ingress traffic flows:

Communication service provider	Ingress from 4G/5G-connected device	Ingress from Fixed Wireless Access
Verizon	Yes	Yes
Vodafone	Yes	No

Architect apps for Wavelength

Wavelength Zones are designed for the following workloads:

- Applications that require edge resiliency across existing AWS hybrid and edge infrastructure deployments
- Applications that need to connect to compute from 4G or 5G mobile devices with ultra-low latency
- Applications that need consistent data rates from mobile devices to compute in a Wavelength Zone

Review *Quotas and considerations*, which includes information about available Wavelength Zones, service differences, and Service Quotas.

Consider the following factors when using Wavelength Zones:

- AWS recommends that you architect the edge applications in a hub and spoke model with the Region to provide the most scalable, resilient, and cost-effective options for components. For more information, see the section called "Workload placement"
- Services that run in Wavelength Zones have different compliance than services in an AWS Region. For more information, see the section called "Compliance validation".

Wavelength Zones have network access that is specific to a telecommunication carrier and location. Therefore, you might need to have multiple Wavelength Zones for your latency-sensitive applications to meet your latency requirements. For more information, see the section called "Networking considerations".

Discover the closest Wavelength Zone endpoint

You can use the following procedures to have client devices discover the closest Wavelength Zone endpoint, for example an Amazon EC2 instance:

- Register the instance with a discovery service such as AWS Cloud Map. For information about how to register an instance, see Registering Instances in the *AWS Cloud Map Developer Guide*.
- Another approach is to use multiple Wavelength Zones across your deployment and utilize adjacent Zones, powered by carrier-developed edge discovery services to route mobile traffic.

For more information, see Deploying dynamic 5G Edge Discovery architectures with AWS Wavelength.

- Applications that run on client devices can run latency tests such as `ping` from the client to select the best endpoint that is registered in AWS Cloud Map, or can use the geolocation data from the mobile device.

Load balancing

Application Load Balancer (ALB) is supported in select Wavelength Zones. Load balancers distribute your incoming traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses, within the Wavelength Zone. Key considerations include:

- Network Load Balancer (NLB) is not supported in Wavelength Zones. To learn more, see Enabling load-balancing of non-HTTP(s) traffic on AWS Wavelength.
- Cross-Zone load balancing across multiple Wavelength Zones is not supported.

ALB is available in the following Wavelength Zones:

- All Wavelength Zones in the `us-east-1` Region.
- All Wavelength Zones in `us-west-2` Region.
- All Wavelength Zones in the `ap-northeast-1` Region.
- All Wavelength Zones in the `eu-central-1` Region.

High availability

Follow these strategies to deploy highly available architectures at the edge.

Deployment

Consider the following:

- **Multiple Wavelength Zones within a given VPC**: using techniques highlighted in the Discover the closest Wavelength Zone endpoint section, you can steer traffic to the optimal Wavelength Zone based on latency or application health.
- **Combine Wavelength Zones with other AWS hybrid and edge locations**: you can combine AWS Local Zones subnets with AWS Wavelength Zones subnets to create highly-available

deployments within a given geography. For example, you can create an Atlanta AWS Local Zone subnet (`us-east-1-atl-2a`) alongside an Atlanta Wavelength Zone subnet (`us-east-1-wl1-atl-wlz-1`) within the same VPC.

DNS resolution

One way to create both physical and logical redundancy across your high-availability edge deployments is to utilize the parent Region as the failover, using simple Route 53-based failover policies to steer traffic to an available endpoint. For more information, see Configuring DNS failover in the *Amazon Route 53 Developer Guide*.

Workload placement

Run the following components in the Region:

- Components that are less latency sensitive
- Components that need to be shared across Zones
- Components that need to persist state, such as databases

Run the application components that need ultra-low latency and higher bandwidth over 5G mobile networks in Wavelength Zones.

For optimal throughput, AWS recommends that you use a public service endpoint when applications in the Wavelength Zone need to connect to AWS services in the parent Region.

Available Wavelength Zones

The following tables list the Wavelength Zones by Region. For more information, see AWS Wavelength Zone locations.

US East (N. Virginia) Wavelength zones

Location	Carrier	Wavelength Zone ID	Network Border Group
Atlanta	Verizon	us-east-1-wl1-atl-wlz-1	us-east-1-wl1-atl-wlz-1
Boston	Verizon	us-east-1-wl1-bos-wlz-1	us-east-1-wl1-bos-wlz-1
Charlotte	Verizon	us-east-1-wl1-clt-wlz-1	us-east-1-wl1-clt-wlz-1
Chicago	Verizon	us-east-1-wl1-chi-wlz-1	us-east-1-wl1-chi-wlz-1
Dallas	Verizon	us-east-1-wl1-dfw-wlz-1	us-east-1-wl1-dfw-wlz-1
Detroit	Verizon	us-east-1-wl1-dtw-wlz-1	us-east-1-wl1-dtw-wlz-1
Houston	Verizon	us-east-1-wl1-iah-wlz-1	us-east-1-wl1-iah-wlz-1
Miami	Verizon	us-east-1-wl1-mia-wlz-1	us-east-1-wl1-mia-wlz-1
Minneapolis	Verizon	us-east-1-wl1-msp-wlz-1	us-east-1-wl1-msp-wlz-1
Nashville	Verizon	us-east-1-wl1-bna-wlz-1	us-east-1-wl1-bna-wlz-1

Location	Carrier	Wavelength Zone ID	Network Border Group
New York City	Verizon	us-east-1-wl1-nyc-wlz-1	us-east-1-wl1-nyc-wlz-1
Tampa	Verizon	us-east-1-wl1-tpa-wlz-1	us-east-1-wl1-tpa-wlz-1
Washington DC	Verizon	us-east-1-wl1-was-wlz-1	us-east-1-wl1-was-wlz-1

US West (Oregon) Wavelength zones

Location	Carrier	Wavelength Zone ID	Network Border Group
Denver	Verizon	us-west-2-wl1-den-wlz-1	us-west-2-wl1-den-wlz-1
Las Vegas	Verizon	us-west-2-wl1-las-wlz-1	us-west-2-wl1-las-wlz-1
Los Angeles	Verizon	us-west-2-wl1-lax-wlz-1	us-west-2-wl1-lax-wlz-1
Phoenix	Verizon	us-west-2-wl1-phx-wlz-1	us-west-2-wl1-phx-wlz-1
San Francisco Bay area	Verizon	us-west-2-wl1-sfo-wlz-1	us-west-2-wl1-sfo-wlz-1
Seattle	Verizon	us-west-2-wl1-sea-wlz-1	us-west-2-wl1-sea-wlz-1

Asia Pacific (Seoul) Wavelength zones

Location	Carrier	Wavelength Zone ID	Network Border Group
Daejeon	SKT	ap-northeast-2-wl1-cjj-wlz-1	ap-northeast-2-wl1-cjj-wlz-1
Seoul	SKT	ap-northeast-2-wl1-sel-wlz-1	ap-northeast-2-wl1-sel-wlz-1

Asia Pacific (Tokyo) Wavelength zones

Location	Carrier	Wavelength Zone ID	Network Border Group
Osaka	KDDI	ap-northeast-1-wl1-kix-wlz-1	ap-northeast-1-wl1-kix-wlz-1
Tokyo	KDDI	ap-northeast-1-wl1-nrt-wlz-1	ap-northeast-1-wl1-nrt-wlz-1

Canada (Central) Wavelength zones

Location	Carrier	Wavelength Zone ID	Network Border Group
Toronto	Bell	ca-central-1-wl1-yto-wlz-1	ca-central-1-wl1-yto-wlz-1

Europe (Frankfurt) Wavelength zones

Location	Carrier	Wavelength Zone ID	Network Border Group
Berlin	Vodafone	eu-central-1-wl1-ber-wlz-1	eu-central-1-wl1-ber-wlz-1

Location	Carrier	Wavelength Zone ID	Network Border Group
Dortmund	Vodafone	eu-central-1-wl1-dtm-wlz-1	eu-central-1-wl1-dtm-wlz-1
Munich	Vodafone	eu-central-1-wl1-muc-wlz-1	eu-central-1-wl1-muc-wlz-1

Europe (London) Wavelength zones

Location	Carrier	Wavelength Zone ID	Network Border Group
London	Vodafone	eu-west-2-wl1-lon-wlz-1	eu-west-2-wl1-lon-wlz-1
Manchester	Vodafone	eu-west-2-wl1-man-wlz-1	eu-west-2-wl1-man-wlz-1
Manchester	British Telecom (BT)	eu-west-2-wl2-man-wlz-1	eu-west-2-wl2-man-wlz-1

Describe your Wavelength Zones

The number and mapping of Wavelength Zones per Region might vary between AWS accounts. The following procedures demonstrate how to list the Wavelength Zones that are available to your account.

To find your Wavelength Zones using the console

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. From the navigation bar, choose the **Regions** selector and then choose the Region.
3. On the navigation pane, choose **EC2 Dashboard**.
4. In the upper-right corner of the page, choose **Account attributes**, **Zones**.

To find your Wavelength Zones using the AWS CLI

- Use the describe-availability-zones command as follows to describe the Wavelength Zones within the specified Region that are enabled for your account.

```
aws ec2 describe-availability-zones --region region-name
```

- Use the describe-availability-zones command as follows to describe the Wavelength Zones regardless of the opt-in status.

```
aws ec2 describe-availability-zones --all-availability-zones
```

Quotas and considerations for Wavelength Zones

Consider the following as you get started with AWS Wavelength.

Contents

Networking considerations

The following controls are enabled by the carrier gateway for internet flows by default and cannot be removed:

Protocol	Between EC2 instance and the internet	Between EC2 instance and a device on the carrier network
TCP	outbound and the response	allowed
UDP	denied	allowed
ICMP	allowed	allowed

- TCP is allowed for outbound and response
- UDP from the internet is denied

  UDP traffic from a device on the carrier network is allowed to route to an EC2 instance in a Wavelength Zone.
- ICMP is allowed

In addition, inbound routing from the carrier network is optimized for devices in the location of the Wavelength Zone. For example, a Wavelength Zone in the San Francisco Bay area allows low latency access only from devices that are in that metro area and carrier network.

Multiple Wavelength Zone considerations

EC2 instances that are in two different Wavelength Zones in the same VPC are not allowed to communicate with each other. If you need communication from one Wavelength Zone to another Wavelength Zone, we recommends that you use multiple VPCs, one for each Wavelength Zone. You can use a transit gateway to connect the VPCs. This configuration enables communication between instances in the Wavelength Zones. For information about how to configure multiple Wavelength Zones, see Extend your VPC resources to Local Zones in the*Amazon VPC User Guide*.

Amazon EC2 considerations

Take the following information into consideration when you launch EC2 instances in Wavelength Zones:

- The following instance types are supported:
  - t3.medium
  - t3.xlarge
  - r5.2xlarge
  - g4dn.2xlarge
- You cannot use Dedicated Instances or Dedicated Hosts.
- EC2 quotas are controlled by the quotas for the home Region.

Amazon EBS considerations

Take the following information into consideration when you use Amazon Elastic Block Store for EC2 instances that are in Wavelength Zones:

- Snapshots of EBS volumes and AMIs are stored in the AWS Region.
- You can only use gp2 volumes.
- The default limit for gp2 storage is 30 TB.

  You can request an increase for this value.

Amazon Elastic Kubernetes Service considerations

Take the following information into consideration when you run an Amazon EKS cluster:

- You must run Kubernetes 1.17 or later.
- When you create your Amazon EKS cluster, you must select an Availability Zone in the VPC, and not a Wavelength Zone.
- When you create your Amazon EKS cluster for private subnets only, you need to add VPC endpoints for Amazon ECR and Amazon Simple Storage Service. For more information, see the section called "Amazon VPC considerations".
- To create node groups in Wavelength Zones for your Amazon EKS cluster, see Launching self-managed Amazon Linux 2 nodes in the *Amazon EKS User Guide*.
- To apply the `aws-auth` ConfigMap to your Amazon EKS cluster, see Managing users or IAM roles for your cluster in the *Amazon EKS User Guide*.

Amazon VPC considerations

Take the following information into consideration when you run Amazon VPC:

- To use VPC endpoints, you must create the endpoint in an Availability Zone in the VPC. You cannot create the endpoint in a Wavelength Zone.
- You cannot assign IPv6 addresses to subnets that are in Wavelength Zones.

Service quotas for Amazon VPC

Wavelength VPCs and Wavelength subnets count toward your Amazon VPC service quotas. For more information about Amazon VPC quotas, see Amazon VPC quotas in the *Amazon VPC User Guide*.

For more information about how to view your service quotas, see Viewing service quotas in the *Service Quotas User Guide*.

Security in AWS Wavelength

Security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security of the cloud and security in the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs. To learn about the compliance programs that apply to AWS Wavelength, see the section called "Compliance validation".
- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

This documentation describes the differences when services run in a Wavelength Zone. For detailed information about service security, see the following:

- Security in Amazon EC2
- Security in Amazon EC2 Auto Scaling
- Security in Amazon ECS clusters
- Security in Amazon EKS

Contents

Resilience in AWS Wavelength

AWS recommends that you architect edge applications in a hub and spoke model with the Region providing the most scalable, resilient, and cost effective options for components that are less

latency sensitive, that need to be shared across Zones, or that have states that need to persist. Then, use Wavelength Zones for the application components that need ultra-low latency, higher bandwidth, or increased quality of service over 5G mobile networks.

If you need to replicate your data or applications in a Wavelength Zone, AWS recommends that you use an Availability Zone in the Region that is not the parent zone as the failover zone. In the following example, the parent Availability Zone is Availability Zone A, so the resources are replicated to Availability Zone B.

To learn more about resiliency in Amazon EC2 and Amazon EC2 Auto Scaling, see the following:

- Resilience in Amazon EC2 in the *Amazon EC2 User Guide*
- Resilience in Amazon EC2 Auto Scaling in the *Amazon EC2 Auto Scaling User Guide*.

For more information about AWS Regions, Availability Zones, Local Zones, and Wavelength Zones, see AWS Global Infrastructure.

Compliance validation for AWS Wavelength

The existing compliance certifications for AWS services apply to services running entirely in an AWS Region. The services running in a Wavelength Zone require a separate evaluation for certifications.

Under the shared responsibility model, AWS is responsible for the hardware and software that run AWS services. This applies to AWS Wavelength, just as it does to an AWS Region. This includes

patching the infrastructure software and configuring infrastructure devices. As a customer, you are responsible for implementing best practices for data encryption, patching the operating system and applications, identity and access management, and operating system, network, and firewall configurations.

AWS has responsibility for configuring and maintaining a network connection between the Wavelength Zone and the AWS Region. Communication sent over this connection between the Wavelength Zone and the Region is encrypted by AWS.

Third-party auditors assess the security and compliance of services in AWS Wavelength as part of multiple AWS compliance programs.

AWS Wavelength currently supports these certifications and standards:

- HIPAA
- ISO (9001, 27001, 27017 and 27018)
- SOC (1, 2, 3)
- Payment Card Industry Data Security Standard (PCI DSS)

For information about your compliance responsibility when using Amazon EC2, see Compliance validation for Amazon EC2 in the *Amazon EC2 User Guide*. For more information about compliance, see AWS Compliance.

Document history for AWS Wavelength Developer Guide

The following table describes significant updates to *AWS Wavelength Developer Guide*. In addition to major changes listed here, we also update the documentation frequently to improve the descriptions and examples, and to address the feedback that you send to us.

- **API version: latest** 2016-11-15
- **Latest documentation update:** November 8, 2022

Change	Description	Release Date
What is AWS Wavelength?	Multiple updates and new sections on multi-access, load balancing, and high availability	April 22, 2024
Wavelength Zones added	This release introduces a new Wavelength Zone in Manchester.	July 25, 2023
Wavelength Zones added	This release introduces a new Wavelength Zone in London.	November 8, 2022
Wavelength Zones added	This release introduces new Wavelength Zones in Seoul, Nashville, and Tampa.	May 25, 2022
Wavelength Zone added	This release introduces a new Wavelength Zone in Toronto.	April 26, 2022
Wavelength Zones added	This release introduces new Wavelength Zones in Charlotte, Detroit, Los Angeles, and Minneapolis.	January 18, 2022
Wavelength Zones added	This release introduces new Wavelength Zones in Dortmund, Berlin and Munich.	December 8, 2021
Wavelength Zones added	This release introduces new Wavelength Zones in Houston, Phoenix and Chicago.	August 5, 2021

Change	Description	Release Date
Wavelength Zone added	This release introduces a new Wavelength Zone in London.	June 16, 2021
New certifications added	AWS Wavelength is now ISO (9001, 27001, 27017 and 27018) and SOC (1, 2, 3) certified and compliant.	May 26, 2021
Payment Card Industry Data Security Standard (PCI DSS)	AWS Wavelength is now certified and compliant with the Payment Card Industry Data Security Standard (PCI DSS).	January 19, 2021
Wavelength Zones added	This release introduces new Wavelength Zones.	September 22, 2020
Initial release	This release introduces AWS Wavelength.	August 6, 2020