Amazon Virtual Private Cloud

IP Address Manager

Amazon Virtual Private Cloud: IP Address Manager

Table of Contents

What is IPAM?

Amazon VPC IP Address Manager (IPAM) is a VPC feature that makes it easier for you to plan, track, and monitor IP addresses for your AWS workloads. You can use IPAM's automated workflows to more efficiently manage IP addresses.

You can use IPAM to do the following:

- Organize IP address space into routing and security domains
- Monitor IP address space that's in use and monitor resources that are using space against business rules
- View the history of IP address assignments in your organization
- Automatically allocate CIDRs to VPCs using specific business rules
- Troubleshoot network connectivity issues
- Enable cross-region and cross-account sharing of your Bring Your Own IP (BYOIP) addresses

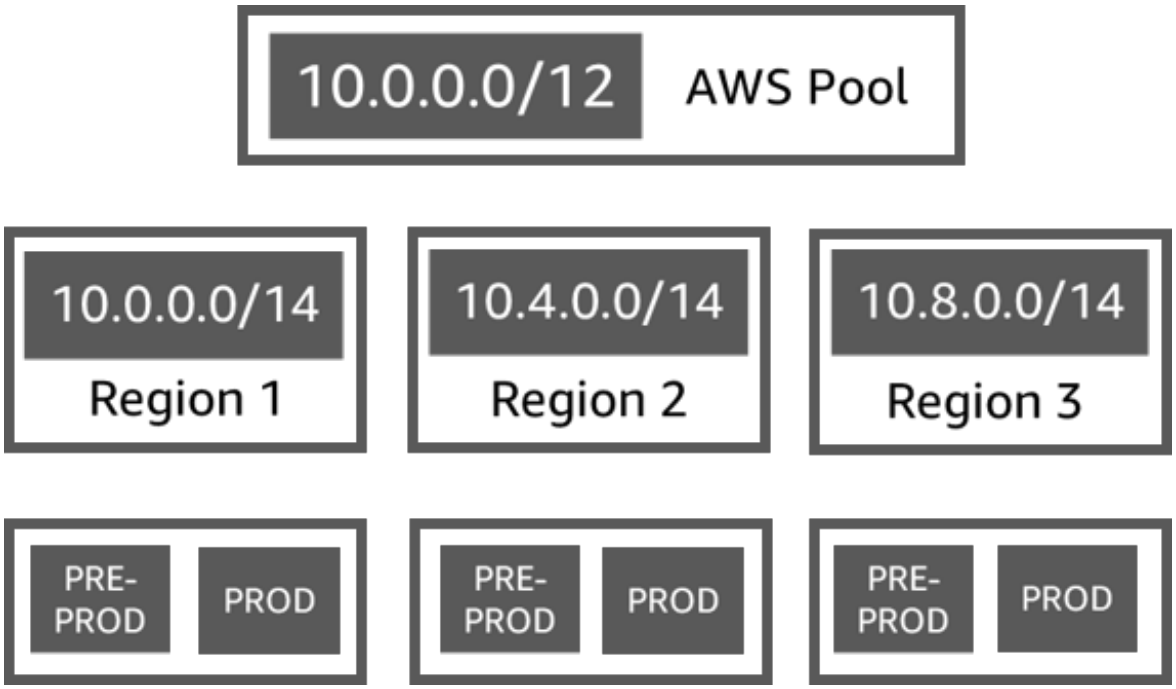
This guide consists of the following sections:

- How IPAM works (p. 2): IPAM concepts and terminology.
- Getting started with IPAM (p. 4): Steps to enable company-wide IP address management with AWS Organizations, create an IPAM, and plan IP address usage.
- Managing IP address space in IPAM (p. 17): Steps to manage your IPAM, scopes, pools, and allocations.
- Tracking IP address usage in IPAM (p. 28): Steps to monitor and track IP address usage with IPAM.
- Tutorials (p. 35): Detailed step-by-step tutorials for creating an IPAM and pools, allocating VPC CIDRs, and bringing your own public IP address CIDRs to IPAM.

How IPAM works

This topic explains some of the key concepts to help you get started with IPAM.

The following diagram shows an IPAM pool hierarchy for multiple AWS Regions within a top-level IPAM pool. Each AWS Regional pool has two IPAM development pools within it, one pool for pre-production and one pool production resources. For more information about IPAM concepts, see the descriptions below the diagram.



To use Amazon VPC IP Address Manager, you first create an IPAM.

When you create the IPAM, you choose which AWS Region to create it in. When you create an IPAM, AWS VPC IPAM automatically creates two scopes for the IPAM. The scopes, together with pools and allocations, are key components of your IPAM.

- A **scope** is the highest-level container within IPAM. An IPAM contains two default scopes. Each scope represents the IP space for a single network. The **private scope** is intended for all private space. The **public scope** is intended for all public space. Scopes enable you to reuse IP addresses across multiple unconnected networks without causing IP address overlap or conflict. Within a scope, you create IPAM pools.
- A **pool** is a collection of contiguous IP address ranges (or CIDRs). IPAM pools enable you to organize your IP addresses according to your routing and security needs. You can have multiple pools within a top-level pool. For example, if you have separate routing and security needs for development and production applications, you can create a pool for each. Within IPAM pools, you allocate CIDRs to AWS resources.
- An **allocation** is a CIDR assignment from an IPAM pool to another resource or IPAM pool. When you create a VPC and choose an IPAM pool for the VPC's CIDR, the CIDR is allocated from the CIDR provisioned to the IPAM pool. You can monitor and manage the allocation with IPAM.

IPAM can manage and monitor private IPv4 CIDRs and public IPv4/IPv6 CIDRs that you own. IPAM can only monitor (not manage) Amazon owned public IP space.

To get started and create an IPAM, see Getting started with IPAM (p. 4).

Getting started with IPAM

Follow the steps in this section to get started with IPAM. You'll begin by accessing IPAM and deciding if you want to delegate an IPAM account. By the end of this section, you will have created an IPAM, created multiple pools of IP addresses, and allocated a CIDR in a pool to a VPC.

Contents

Access IPAM

As with other AWS services, you can create, access, and manage your IPAM using the following methods:

- **AWS Management Console**: Provides a web interface that you can use to create and manage your IPAM. See https://console.aws.amazon.com/ipam/.
- **AWS Command Line Interface (AWS CLI)**: Provides commands for a broad set of AWS services, including Amazon VPC. The AWS CLI is supported on Windows, macOS, and Linux. To get the AWS CLI, see AWS Command Line Interface.
- **AWS SDKs**: Provide language-specific APIs. The AWS SDKs take care of many of the connection details, such as calculating signatures, handling request retries, and handling errors. For more information, see AWS SDKs.
- **Query API**: Provides low-level API actions that you call using HTTPS requests. Using the Query API is the most direct way to access IPAM. However, it requires your application to handle low-level details such as generating the hash to sign the request, and handling errors. For more information, see Amazon IPAM actions in the Amazon EC2 API Reference.

This guide primarily focuses on using the AWS Management Console to create, access, and manage your IPAM. In each description of how to complete a process in the console, we include links to the AWS CLI documentation that shows you how to do the same thing by using the AWS CLI.

If you are a first-time user of IPAM, review How IPAM works (p. 2) to learn about the role of IPAM in Amazon VPC and then continue with the instructions in Configure permissions for your IPAM (p. 4).

Configure permissions for your IPAM

Before you begin using IPAM, you must choose one of the options in this section to enable IPAM to monitor CIDRs associated with EC2 networking resources and store metrics:

- To enable IPAM to integrate with AWS Organizations to enable the Amazon VPC IPAM service to manage and monitor networking resources created by all AWS Organizations member accounts, see Integrate IPAM with AWS Organizations (p. 5).

- To use a single AWS account with IPAM and enable the Amazon VPC IPAM service to manage and monitor the networking resources you create with the single account, see Use IPAM with a single account (p. 6).

If you do not choose one of these options, you can still create IPAM resources, such as pools, but you won't see metrics in your dashboard and you will not be able to monitor the status of resources.

Contents

Integrate IPAM with AWS Organizations

Optionally, you can follow the steps in this section to integrate IPAM with AWS Organizations and delegate a member account as the IPAM account.

The IPAM account is responsible for creating an IPAM and using it to manage and monitor IP address usage.

Integrating IPAM with AWS Organizations and delegating an IPAM admin has the following benefits:

- **Share your IPAM pools with your organization**: When you delegate an IPAM account, IPAM enables other AWS Organizations member accounts in the organization to allocate CIDRs from IPAM pools that are shared using AWS Resource Access Manager (RAM). For more information on setting up an organization, see What is AWS Organizations? in the *AWS Organizations User Guide*.
- **Monitor IP address usage in your organization**: When you delegate an IPAM account, you give IPAM permission to monitor IP usage across all of your accounts. As a result, IPAM automatically imports CIDRs that are used by existing VPCs across other AWS Organizations member accounts into IPAM.

If you do not delegate an AWS Organizations member account as an IPAM account, IPAM will monitor resources only in the AWS account that you use to create the IPAM.

Important

- You must enable integration with AWS Organizations by using IPAM in the AWS management console or the enable-ipam-organization-admin-account AWS CLI command. This ensures that the `AWSServiceRoleForIPAM` service-linked role is created. If you enable trusted access with AWS Organizations by using the AWS Organizations console or the register-delegated-administrator AWS CLI command, the `AWSServiceRoleForIPAM` service-linked role isn't created, and you can't manage or monitor resources within your organization.

Note
When integrating with AWS Organizations:

- You cannot use IPAM to manage IP addresses across multiple AWS Organizations.
- IPAM charges you for each active IP address that it monitors in your organization's member accounts. For more information about pricing, see IPAM pricing.
- You must have an account in AWS Organizations and a management account set up with one or more member accounts. For more information about account types, see Terminology and concepts in the *AWS Organizations User Guide*. For more information on setting up an organization, see Getting started with AWS Organizations.
- The IPAM account must be an AWS Organizations member account. You cannot use the AWS Organizations management account as the IPAM account.

- The IPAM account must have an IAM policy attached to it that permits the `iam:CreateServiceLinkedRole` action. When you create the IPAM, you automatically create the AWSServiceRoleForIPAM service-linked role.
- The IAM user account associated with the AWS Organizations management account must have the following IAM policy actions attached:
  - `ec2:EnableIpamOrganizationAdminAccount`
  - `organizations:EnableAwsServiceAccess`
  - `organizations:RegisterDelegatedAdministrator`
  - `iam:CreateServiceLinkedRole`

  For more information on managing IAM policies, see Editing IAM policies in the *IAM User Guide*.

AWS Management Console

To select an IPAM account

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the AWS Management Console, choose the AWS Region in which you want to work with IPAM.
3. In the navigation pane, choose **Settings**.
4. Enter the AWS account ID for an IPAM account. The IPAM administrator must be an AWS Organizations member account.
5. Choose **Delegate**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

- To delegate an IPAM admin account using AWS CLI, use the following command: enable-ipam-organization-admin-account

When you delegate an Organizations member account as an IPAM account, IPAM automatically creates a service-linked IAM role in all member accounts in your organization. IPAM monitors the IP address usage in these accounts by assuming the service-linked IAM role in each member account, discovering the resources and their CIDRs, and integrating them with IPAM. The resources within all member accounts will be discoverable by IPAM regardless of their Organizational Unit. If there are member accounts that have created a VPC, for example, you'll see the VPC and its CIDR in the Resources section of the IPAM console.

Important
The role of the AWS Organizations management account that delegated the IPAM admin is now complete. To continue using IPAM, the IPAM admin account must log into Amazon VPC IPAM and create an IPAM.

Use IPAM with a single account

If you choose not to Integrate IPAM with AWS Organizations (p. 5), you can use IPAM with a single AWS account.

When you create an IPAM in the next section, a service-linked role is automatically created for the Amazon VPC IPAM service in AWS Identity and Access Management. IPAM uses the service-linked role to monitor and store metrics for CIDRs associated with EC2 networking resources. For more information on the service-linked role and how IPAM uses it, see Service-linked roles for IPAM (p. 98).

Important
If you use IPAM with a single AWS account, you must ensure that the AWS account you use to create the IPAM has an IAM policy attached to it that permits the `iam:CreateServiceLinkedRole` action. When you create the IPAM, you automatically create the AWSServiceRoleForIPAM service-linked role. For more information on managing IAM policies, see Editing IAM policies in the *IAM User Guide*.

Once the single AWS account has permission to create the IPAM service-linked role, go to Create an IPAM (p. 7).

Create an IPAM

Follow the steps in this section to create your IPAM. If you have delegated an IPAM administrator, these steps should be completed by the IPAM account.

Important
When you create an IPAM, you will be asked to allow IPAM to replicate data from source accounts into an IPAM delegate account. To integrate IPAM with AWS Organizations, IPAM needs your permission to replicate resource and IP usage details across accounts (from member accounts to the delegated IPAM member account) and across AWS Regions (from operating Regions to the home Region of your IPAM). For single account IPAM users, IPAM needs your permission to replicate resource and IP usage details across operating Regions to the home Region of your IPAM.

When you create the IPAM, you choose the AWS Regions where the IPAM is allowed to manage IP address CIDRs. These AWS Regions are called *operating Regions*. IPAM discovers and monitors resources only in the AWS Regions that you select as operating Regions. IPAM doesn't store any data outside of the operating Regions that you select.

The following example hierarchy shows how the AWS Regions that you assign when you create the IPAM will impact the Regions that will be available for pools that you create later.

- **IPAM operating in AWS Region 1 and AWS Region 2**
  - Private scope
    - Top-level IPAM pool
      - Regional IPAM pool in **AWS Region 2**
        - Development pool
          - Allocation for a VPC in **AWS Region 2**

You can only create one IPAM. For more information about increasing quotas related to IPAM, see Quotas for your IPAM (p. 101).

AWS Management Console

To create an IPAM

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the AWS Management Console, choose the AWS Region in which you want to create the IPAM.
3. On the service home page, choose **Create IPAM**.
4. Select **Allow Amazon VPC IP Address Manager to replicate data from source account(s) into the IPAM delegate account**. If you do not select this option, you cannot create an IPAM.
5. Under **Operating regions**, select the AWS Regions in which this IPAM can manage and discover resources. The AWS Region in which you are creating your IPAM is selected as one of the operating Regions by default. For example, if you're creating this IPAM in AWS Region `us-`

`east-1` but you want to create Regional IPAM pools later that provide CIDRs to VPCs in `us-west-2`, select `us-west-2` here. If you forget an operating Region, you can return at a later time and edit your IPAM settings.

6. Choose **Create**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to create, modify, and view details related to your IPAM:

1. Create the IPAM: create-ipam
2. View the IPAM that you've created: describe-ipams
3. View the scopes that are created automatically: describe-ipam-scopes
4. Modify an existing IPAM: modify-ipam

When you have completed these steps, IPAM has done the following:

- Created your IPAM. You can see the IPAM and the currently selected operating Regions by choosing IPAMs in the left navigation pane of the console.
- Created one private and one public scope. You can see the scopes by choosing **Scopes** in the navigation pane. For more information about scopes, see How IPAM works (p. 2).

Plan for IP address provisioning

Follow the steps in this section to plan for IP address provisioning by using IPAM pools. If you have configured an IPAM account, these steps should be completed by that account.

Important
To use IPAM pools across AWS accounts, you must integrate IPAM with AWS Organizations or some features may not work properly. For more information, see Integrate IPAM with AWS Organizations (p. 5).

In IPAM, a pool is a collection of contiguous IP address ranges (or CIDRs). Pools enable you to organize your IP addresses according to your routing and security needs. You can create pools for AWS Regions outside of your IPAM Region. For example, if you have separate routing and security needs for development and production applications, you can create a pool for each.

In the first step in this section, you'll create a top-level pool. Then, you'll create a Regional pool within the top-level pool. Within the Regional pool, you can create additional pools as needed, such as a production and development environment pools. By default, you can create pools up to a depth of 10. For information on IPAM quotas, see Quotas for your IPAM (p. 101).

Note
The terms *provision* and *allocate* are used throughout this user guide and the IPAM console. *Provision* is used when you add a CIDR to an IPAM pool. *Allocate* is used when you associate a CIDR from an IPAM pool with a resource.

The following is an example hierarchy of the pool structure that you will create by completing the steps in this section:

- IPAM operating in AWS Region 1 and AWS Region 2
  - Private scope
    - Top-level pool

- Regional pool in AWS Region 1
  - Development pool
    - Allocation for a VPC

This structure serves as an example of how you might want to use IPAM, but you can use IPAM to suit the needs of your organization. If you are creating a single IPAM pool, complete the steps in Create a top-level pool (p. 10) and then skip to Allocate CIDRs (p. 15).

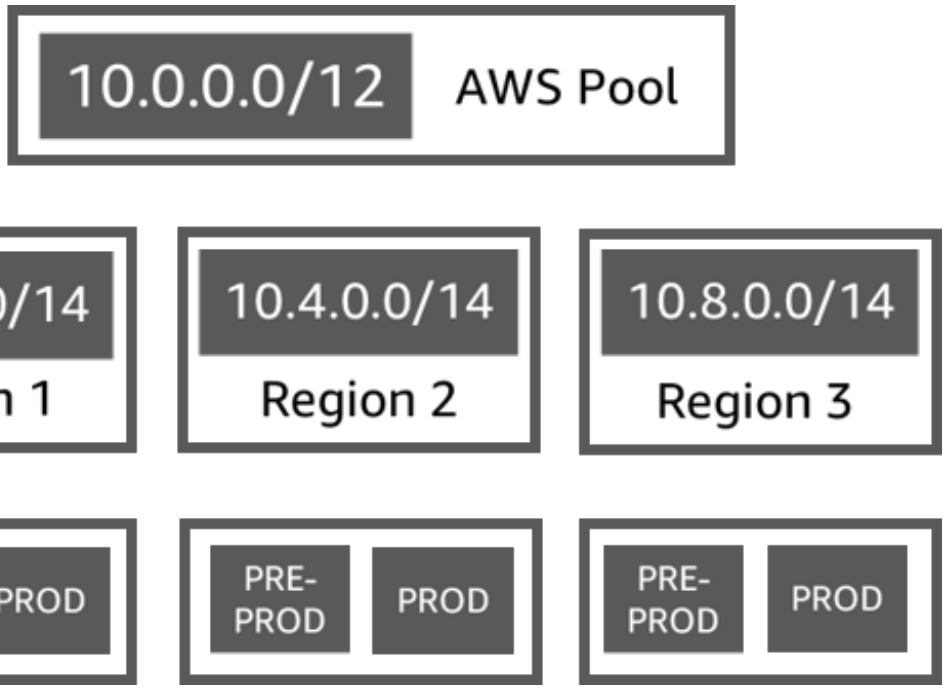
Contents

Example IPAM pool plans

You can use IPAM to suit the needs of your organization. This section provides examples of how you might organize your IP addresses.

Pools in multiple AWS Regions

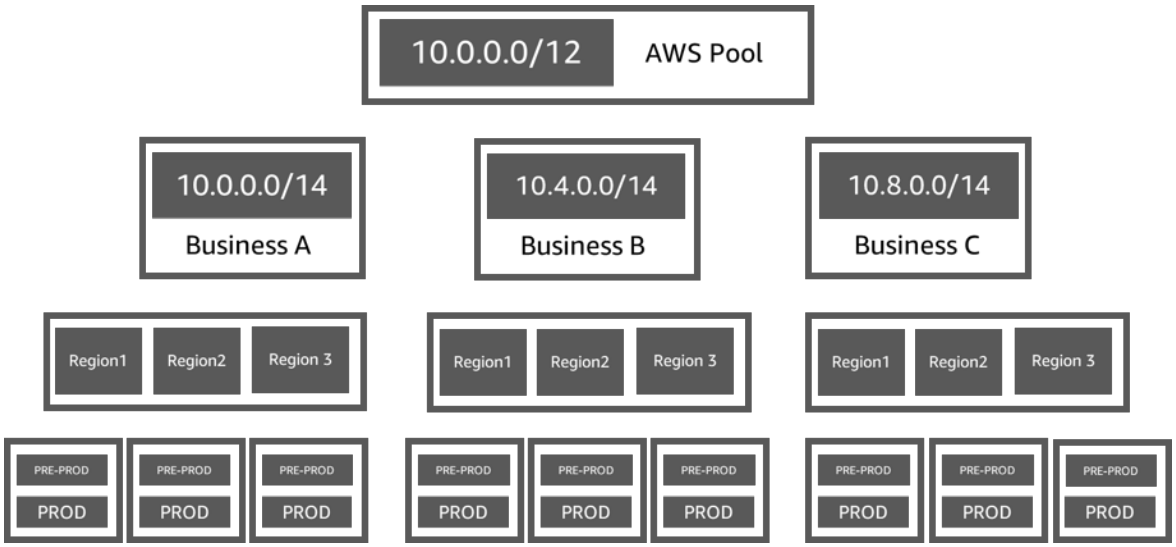
The following example shows an IPAM pool hierarchy for multiple AWS Regions within a top-level pool. Each AWS Regional pool has two IPAM development pools within it, one pool for pre-production resources and one pool for production resources.



Pools for multiple lines of business

The following example shows an IPAM pool hierarchy for multiple lines of business within a top-level pool. Each pool for each line of business contains three AWS Regional pools. Each Regional pool has two

IPAM development pools within it, one pool for pre-production resources and one pool for production resources.



Create a top-level pool

Follow the steps in this section to create a top-level IPAM pool. When you create the pool, you provision a CIDR for the pool to use. The pool assigns space within that CIDR to allocations within the pool. An allocation is a CIDR assignment from an IPAM pool to another resource or IPAM pool.

The following example shows the hierarchy of the pool structure that you can create with instructions in this guide. At this step, you are creating the top-level IPAM pool:

- IPAM operating in AWS Region 1 and AWS Region 2
  - Private scope
    - **Top-level pool (10.0.0.0/8)**
      - Regional pool in AWS Region 2 (10.0.0.0/16)
        - Development pool (10.0.0.0/24)
          - Allocation for a VPC (10.0.0.0/25)

In the preceding example, the CIDRs that are used are examples only. They illustrate that each pool within the top-level pool is provisioned with a portion of the top-level CIDR.

When you create an IPAM pool, you can configure rules for the allocations that are made within the IPAM pool.

Allocation rules enable you to configure the following:

- Whether IPAM should automatically import CIDRs into the IPAM pool if it finds them within this pool's CIDR range
- The required netmask length for allocations within the pool
- The required tags for resources within the pool
- The required locale for resources within the pool. The locale is the AWS Region where an IPAM pool is available for allocations.

Allocation rules determine whether resources are compliant or noncompliant. For additional information about compliance, see Monitor CIDR usage by resource (p. 29).

Important
There is an additional implicit rule that is not displayed in the allocation rules. If the resource is in an IPAM pool that is a shared resource in AWS Resource Access Manager (RAM), the resource owner must be configured as a principal in AWS RAM. For more information about sharing pools with RAM, see Share an IPAM pool using AWS RAM (p. 18).

The following example shows how you might use allocation rules to control access to an IPAM pool:

Example

When you create your pools based on routing and security needs, you might want to allow only certain resources to use a pool. In such cases, you can set an allocation rule stating that any resource that wants a CIDR from this pool must have a tag that matches the allocation rule tag requirements. For example, you could set an allocation rule stating that only VPCs with the tag *prod* can get CIDRs from an IPAM pool. You could also set a rule stating that CIDRs allocated from this pool can be no larger than /24. In this case, a resource could still be created using a CIDR larger than /24 from this pool if the space is available, but because doing so violates an allocation rule on the pool, IPAM flags this resource as noncompliant.

AWS Management Console

To create a pool

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. If you don't want to use the default private scope, from the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2).
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a description for the pool.
6. Choose **No source pool**.
7. For the **Locale**, choose **None**. You will set the locale on the Regional pool.

   The locale is the AWS Region where you want this IPAM pool to be available for allocations. For example, you can only allocate a CIDR for a VPC from an IPAM pool that shares a locale with the VPC's Region. Note that when you have chosen a locale for a pool, you cannot modify it.

   Note
   If you are creating a single pool only and not a top-level pool with Regional pools within it, you would want to choose a Locale for this pool so that the pool is available for allocations.
8. Select the **Address family** for this pool. Choose IPv4 if the IP addresses in this pool will be IPv4 addresses. Choose IPv6 if they will be IPv6 addresses. If the scope you've chosen for this pool is the public scope, you'll have the option of using either IPv4 or IPv6. If the scope you've chosen for this pool is private, IPv4 is the only option.
9. (Optional) Choose a CIDR to provision for the pool. You can create a pool without a CIDR, but you won't be able to use the pool for allocations until you've provisioned a CIDR for it.
10. Choose optional allocation rules for this pool:

    - **Automatically import discovered resources**: This option is not available if the **Locale** is set to **None**. If selected, IPAM will continuously look for resources within the CIDR range of this pool and automatically import them as allocations into your IPAM. Note the following:
      - The CIDRs that will be allocated for these resources must not already be allocated to other resources in order for the import to succeed.
      - IPAM will import a CIDR regardless of its compliance with the pool's allocation rules, so a resource might be imported and subsequently marked as noncompliant.

  - If IPAM discovers multiple CIDRs that overlap, IPAM will import the largest CIDR only.
  - If IPAM discovers multiple CIDRs with matching CIDRs, IPAM will randomly import one of them only.
- **Minimum netmask length**: The minimum netmask length required for CIDR allocations in this IPAM pool to be compliant and the largest size CIDR block that can be allocated from the pool. The minimum netmask length must be less than the maximum netmask length. Possible netmask lengths for IPv4 addresses are 0 - 32. Possible netmask lengths for IPv6 addresses are 0 - 128.
- **Default netmask length**: A default netmask length for allocations added to this pool. For example, if the CIDR that's provisioned to this pool is **`10.0.0.0/8`** and you enter **`16`** here, any new allocations in this pool will default to a netmask length of **`/16`**.
- **Maximum netmask length**: The maximum netmask length that will be required for CIDR allocations in this pool. This value dictates the smallest size CIDR block that can be allocated from the pool.
- **Tagging requirements**: The tags that are required for resources to allocate space from the pool. If the resources have their tags changed after they have allocated space or if the allocation tagging rules are changed on the pool, the resource may be marked as noncompliant.
- **Locale**: The locale that will be required for resources that use CIDRs from this pool. Automatically imported resources that do not have this locale will be marked noncompliant. Resources that are not automatically imported into the pool will not be allowed to allocate space from the pool unless they are in this locale.

11. (Optional) Choose **Tags** for the pool.
12. Choose **Create pool**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to create or edit a top-level pool in your IPAM:

1. Create a pool: create-ipam-pool.
2. Edit the pool after you create it to modify the allocation rules: modify-ipam-pool.

Create a Regional pool

Follow the steps in this section to create a Regional pool within your top-level pool. If you need only a top-level pool, and don't need additional Regional and development pools, skip to Allocate CIDRs (p. 15).

The following example shows the hierarchy of the pool structure that you create by following the instructions in this guide. At this step, you are creating the Regional IPAM pool:

- IPAM operating in AWS Region 1 and AWS Region 2
  - Private scope
    - Top-level pool (10.0.0.0/8)
      - **Regional pool in AWS Region 2 (10.0.0.0/16)**
        - Development pool (10.0.0.0/24)
          - Allocation for a VPC (10.0.0.0/25)

In the preceding example, the CIDRs that are used are examples only. They illustrate that each pool within the top-level pool is provisioned with a portion of the top-level CIDR.

AWS Management Console

To create a Regional pool within a top-level pool

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. If you don't want to use the default private scope, from the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2).
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a description for the pool.
6. Under **Source pool**, choose the top-level pool that you created in the previous section.
7. Choose the locale for the pool. Choosing a locale ensures there are no cross-region dependencies between your pool and the resources allocating from it. The available options come from the operating Regions that you chose when you created your IPAM.

   The locale is the AWS Region where you want this IPAM pool to be available for allocations. For example, you can only allocate a CIDR for a VPC from an IPAM pool that shares a locale with the VPC's Region. Note that when you have chosen a locale for a pool, you cannot modify it.
8. (Optional) Choose a CIDR to provision for the pool. You can create a pool without a CIDR, but you won't be able to use the pool for allocations until you've provisioned a CIDR for it. You can add CIDRs to a pool at any time by editing the pool.
9. You have the same allocation rule options here as you did when you created the top-level pool. See Create a top-level pool (p. 10) for an explanation of the options that are available when you create pools. The allocation rules for the Regional pool are not inherited from the top-level pool. If you do not apply any rules here, there will be no allocation rules set for the pool.
10. (Optional) Choose **Tags** for the pool.
11. When you've finished configuring your pool, choose **Create pool**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to create a Regional pool in your IPAM:

1. Get the ID of the scope that you want to create the pool in: describe-ipam-scopes
2. Get the ID of the pool that you want to create the pool in: describe-ipam-pools
3. Create the pool: create-ipam-pool
4. View the new pool: describe-ipam-pools

Repeat these steps to create additional pools within the top-level pool, as needed.

Create a development pool

Follow the steps in this section to create a development pool within your Regional pool. If you need only a top-level and Regional pool, and don't need development pools, skip to Allocate CIDRs (p. 15).

The following example shows the hierarchy of the pool structure that you can create with the instructions in this guide. At this step, you are creating a development IPAM pool:

- IPAM operating in AWS Region 1 and AWS Region 2
  - Private scope
    - Top-level pool (10.0.0.0/8)
      - Regional pool in AWS Region 1 (10.0.0.0/16)
        - **Development pool for non-production VPCs (10.0.0.0/24)**
          - Allocation for a VPC (10.0.1.0/25)
        - Development pool for production VPCs (10.0.1.0/24)
      - Regional pool in AWS Region 2 (10.1.0.0/16)

In the preceding example, the CIDRs that are used are examples only. They illustrate that each pool within the top-level pool is provisioned with a portion of the top-level CIDR.

AWS Management Console

To create a development pool within a Regional pool

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. If you don't want to use the default private scope, from the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2).
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a description for the pool.
6. Under **Source pool**, choose the Regional pool.
7. Choose the locale for the pool. Choosing a locale ensures there are no cross-region dependencies between your pool and the resources allocating from it. The available options here come from the operating Regions that you chose when you created your IPAM.

   The locale is the AWS Region where you want this IPAM pool to be available for allocations. For example, you can only allocate a CIDR for a VPC from an IPAM pool that shares a locale with the VPC's Region. Note that when you have chosen a locale for a pool, you cannot modify it.
8. (Optional) Choose a CIDR to provision for the pool. You can only provision a CIDR that was provisioned to the top-level pool. You can create a pool without a CIDR, but you won't be able to use the pool for allocations until you've provisioned a CIDR for it. You can add CIDRs to a pool at any time by editing the pool.
9. You have the same allocation rule options here as you did when you created the top-level and Regional pool. See Create a top-level pool (p. 10) for an explanation of the options that are available when you create pools. The allocation rules for the pool are not inherited from the pool above it in the hierarchy. If you do not apply any rules here, no allocation rules will be set for the pool.
10. (Optional) Choose **Tags** for the pool.
11. When you've finished configuring your pool, choose **Create pool**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to create a Regional pool in your IPAM:

1. Get the ID of the scope that you want to create the pool in: describe-ipam-scopes

2. Get the ID of the pool that you want to create the pool in: describe-ipam-pools
3. Create the pool: create-ipam-pool
4. View the new pool: describe-ipam-pools

Repeat these steps to create additional development pools within the Regional pool, as needed.

Allocate CIDRs

Follow the steps in this section to allocate a CIDR from an IPAM pool to a resource.

Note
The terms *provision* and *allocate* are used throughout this user guide and the IPAM console. *Provision* is used when you add a CIDR to an IPAM pool. *Allocate* is used when you associate a CIDR from an IPAM pool with a resource.

The following example shows the hierarchy of the pool structure that you can create with the instructions in this section:

- IPAM operating in AWS Region 1 and AWS Region 2
  - Private scope
    - Top-level IPAM pool (10.0.0.0/8)
      - Regional IPAM pool in AWS Region 2 (10.0.0.0/16)
        - Development pool (10.0.0.0/24)
          - **Allocation - VPC (10.0.0.0/25)**

In the preceding example, the CIDRs that are used are examples only. They illustrate that each pool within the top-level pool is provisioned with a portion of the top-level CIDR.

You can allocate CIDRs from an IPAM pool in the following ways:

- Use an AWS service that's integrated with IPAM, such as Amazon VPC, and select the option to use an IPAM pool for the CIDR. IPAM automatically creates the allocation in the pool for you.
- Manually allocate a CIDR within an IPAM pool to reserve it for later use with an AWS service that's integrated with IPAM, such as Amazon VPC.

This section walks you through both options: how to use the AWS services integrated with IPAM to provision an IPAM pool CIDR, and how to manually reserve IP address space.

Contents

Create a VPC that uses an IPAM pool CIDR

Follow the steps in Creating a VPC in the *Amazon VPC User Guide*. When you reach the step to choose a CIDR for the VPC, you will have an option to use a CIDR from an IPAM pool.

If you choose the option to use an IPAM pool when you create the VPC, AWS allocates a CIDR in the IPAM pool. You can view the allocation in IPAM by choosing a pool in the content pane of the IPAM console and viewing the Resources tab for the pool.

Note
For complete instructions using the AWS CLI, including creating a VPC, see the Tutorials (p. 35) section.

Manually allocate a CIDR to a pool to reserve IP address space

Follow the steps in this section to manually allocate a CIDR to a pool. You might do this in order to reserve a CIDR within an IPAM pool for later use. You can also reserve space in your IPAM pool to represent an on-premises network. IPAM will manage that reservation for you and indicate if any CIDRs overlap with your on-premises IP space.

Important
You cannot manually allocate CIDRs from pools in the public scope.

AWS Management Console

To manually allocate a CIDR

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, the default private scope is selected. If you don't want to use the default private scope, from the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2).
4. In the content pane, choose a pool.
5. Choose **Actions** > **Allocate CIDR**.
6. Choose whether to define the exact CIDR to allocate (for example, **`10.0.0.0/24`**), or choose the netmask length only (or example, **`/24`**).
7. Choose **Allocate**.
8. You can view the allocation in IPAM by choosing **Pools** in the navigation pane, choosing a pool, and viewing the **Allocations** tab for the pool.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to manually allocate a CIDR to a pool:

1. Get the ID of the IPAM pool that you want to create the allocation in: describe-ipam-pools.
2. Create the allocation: allocate-ipam-pool-cidr.
3. View the allocation: get-ipam-pool-allocations.

To release a manually allocated CIDR, see Release an allocation (p. 25).

Managing IP address space in IPAM

The tasks in this section are optional. If you want to complete the tasks in this section, and you have delegated an IPAM account, the tasks should be completed by the IPAM administrator.

Follow the steps in this section to manage your IP address space in IPAM.

Contents

Enforce IPAM use for VPC creation

> **Note**
> This section describes how to create a service-control policy in AWS Organizations that enforces IPAM use for VPC creation. This section is only applicable to you if you've enabled IPAM to integrate with AWS Organizations. For more information, see Integrate IPAM with AWS Organizations (p. 5).

Service control policies (SCP) in AWS Organizations are a type of organization policy that enable you to manage permissions in your organization. Follow the steps in this section to create an SCP and restrict users in your AWS Organizations account to creating VPCs with CIDRs from an IPAM pool and associating CIDRs to the VPCs from the pool. Users in the account will not be able to create VPCs with CIDRs or associate CIDRs to VPCs from any other pools from the one you choose.

To create an SCP and restrict user access to a pool

1. Follow the steps in Creating an SCP in the *AWS Organizations User Guide* and enter the following text in the JSON editor:

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Deny",
   "Action": ["ec2:CreateVpc", "ec2:AssociateVpcCidrBlock"],
   "Resource": "arn:aws:ec2:*:*:vpc/*",
   "Condition": {
    "StringNotEquals": {
     "ec2:Ipv4IpamPoolId": "ipam-pool-0123456789abcdefg"
```

```
    }
   }
  }
 ]
}
```

2. Change the `ipam-pool-0123456789abcdefg` example value to the IPv4 pool ID you would like to restrict users to.
3. Optionally, you can also add a condition for `ec2:Ipv6IpamPoolId` to restrict access to a specific IPv6 pool.

Share an IPAM pool using AWS RAM

Follow the steps in this section to share an IPAM pool using AWS Resource Access Manager (RAM). When you share an IPAM pool with RAM, "principals" can allocate CIDRs from the pool to AWS resources, such as VPCs, from their respective accounts. A principal is a concept in RAM that means any AWS account, IAM role, IAM user, or organizational unit in AWS Organizations. For more information, see Sharing your AWS resources in the *AWS RAM User Guide*.

Note

- You can only share an IPAM pool with AWS RAM if you've integrated IPAM with AWS Organizations. For more information, see Integrate IPAM with AWS Organizations (p. 5). You cannot share an IPAM pool with AWS RAM f you are a single account IPAM user.
- You must enable resource sharing with AWS Organizations in AWS RAM. For more information, see Enable resource sharing within AWS Organizations in the *AWS RAM User Guide*.
- RAM sharing is only available in your IPAM's home AWS Region. You must create the share in the AWS Region that the IPAM is in, not in the Region of the IPAM pool.
- The account that creates and deletes IPAM pool resource shares must have the following permissions in their IAM policy:
  - `ec2:PutResourcePolicy`
  - `ec2:DeleteResourcePolicy`
- You can add multiple IPAM pools to a RAM share.

AWS Management Console

To share an IPAM pool using RAM

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, the default private scope is selected. If you don't want to use the default private scope, from the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2).
4. In the content pane, choose the pool you want to share and choose **Actions** > **View details**.
5. Under **Resource sharing**, choose **Create resource share**. As a result, the AWS RAM console opens. You'll create the shared pool in AWS RAM.
6. Choose **Create a resource share**.
7. Add a **Name** for the shared resource.
8. Under **Select resource type**, select IPAM pools and choose one or more IPAM pools.
9. Choose **Next**.
10. Choose one of the permissions for the resource share:

- **AWSRAMDefaultPermissionsIpamPool**: Choose this permission to allow principals to view the CIDRs and allocations in the shared IPAM pool and allocate/release CIDRs in the pool.
- **AWSRAMPermissionIpamPoolByoipCidrImport**: Choose this permission to allow principals to import BYOIP CIDRs into the shared IPAM pool. You will need this permission only if you have existing BYOIP CIDRs and you want to import them to IPAM and share them with principals. For additional information on BYOIP CIDRs to IPAM, see Tutorial: Transfer existing BYOIP IPv4 CIDRs to IPAM (p. 92).

11. Choose the principals that are allowed to access this resource. If principals will be importing existing BYOIP CIDRs to this shared IPAM pool, add the BYOIP CIDR owner account as principal.
12. Review the resource share options and the principals you'll be sharing with and choose **Create**.

Command line

The command(s) in this section link to the AWS CLI Reference documentation. There you'll find detailed descriptions of the options you can use when you run the command(s).

Use the following AWS CLI commands to share an IPAM pool using RAM:

1. Get the ARN of the IPAM: describe-ipam-pools
2. Create the resource share: create-resource-share
3. View the resource share: get-resource-shares

As a result of creating the resource share in RAM, other principals can now allocate CIDRs to resources using the IPAM pool. For information on monitoring resources created by principals, see Monitor CIDR usage by resource (p. 29). For more information on how to create a VPC and allocate a CIDR from a shared IPAM pool, see Creating a VPC in the *Amazon VPC User Guide*.

Provision CIDRs to a pool

Follow the steps in this section to provision CIDRs to a pool. If you already provisioned a CIDR when you created the pool, you might need to provision additional CIDRs if a pool is nearing full allocation. To monitor pool usage, see Monitor CIDR usage with the IPAM dashboard (p. 28).

Note
The terms *provision* and *allocate* are used throughout this user guide and the IPAM console. *Provision* is used when you add a CIDR to an IPAM pool. *Allocate* is used when you associate a CIDR from an IPAM pool with a resource.

AWS Management Console

To provision CIDRs to a pool

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, the default private scope is selected. If you don't want to use the default private scope, from the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2)
4. In the content pane, choose the pool that you want to add a CIDR to.
5. Choose **Actions** > **Provision CIDRs**.
6. Enter the CIDR that you want to add, and then choose **Add new CIDR** for additional CIDRs.

   Note
   When you provision CIDRs to a pool:

- The CIDR you want to provision must be available in the scope.
- If you are provisioning CIDRs to a pool within a pool, then the CIDR space you want to provision must be available in the pool.

7. Choose **Request provisioning**.
8. You can view the CIDR in IPAM by choosing **Pools** in the navigation pane, choosing a pool, and viewing the CIDRs tab for the pool.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to provision CIDRs to a pool:

1. Get the ID of an IPAM pool: describe-ipam-pools
2. Get the CIDRs that are provisioned to the pool: get-ipam-pool-cidrs
3. Provision a new CIDR to the pool: provision-ipam-pool-cidr
4. Get the CIDRs that are provisioned to the pool and view the new CIDR: get-ipam-pool-cidrs

Deprovision CIDRs from a pool

Follow the steps in this section to deprovision CIDRs from an IPAM pool. When you deprovision all pool CIDRs, the pool can no longer be used for allocations. You must first provision a new CIDR to the pool before you can use the pool for allocations.

Important
You cannot deprovision the CIDR if there are allocations in the pool. To remove allocations, see Release an allocation (p. 25).

AWS Management Console

To deprovision a pool CIDR

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. From the dropdown menu at the top of the content pane, choose the scope that you want to use. For more information about scopes, see How IPAM works (p. 2).
4. In the content pane, choose the pool whose CIDRs you want to deprovision.
5. Choose the **CIDRs** tab.
6. Select one or more CIDRs and choose **Deprovision CIDRs**.
7. Choose **Deprovision CIDR**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to deprovision a pool CIDR:

1. Get an IPAM pool ID: describe-ipam-pools
2. View your current CIDRs for the pool: get-ipam-pool-cidrs
3. Deprovision CIDRs: deprovision-ipam-pool-cidr

4. View your updated CIDRs: get-ipam-pool-cidrs

To provision a new CIDR to the pool, see Deprovision CIDRs from a pool (p. 20). If you want to delete the pool, see Delete a pool (p. 21).

Edit a pool

Follow the steps in this section to edit an IPAM pool. You may want to edit a pool to change the allocation rules in the pool. For more information about allocation rules, see Create a top-level pool (p. 10).

AWS Management Console

To edit a pool

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, the default private scope is selected. If you don't want to use the default private scope, from the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2)
4. In the content pane, choose the pool whose CIDR you want to edit.
5. Choose **Actions** > **Edit**.
6. Make any changes you need to the pools. For information about pool configuration options, see Create a top-level pool (p. 10).
7. Choose **Update**.

Command line

Use the following AWS CLI commands to edit a pool:

1. Get an IPAM pool ID: describe-ipam-pools
2. Modify the pool: modify-ipam-pool

Delete a pool

Follow the steps in this section to delete an IPAM pool.

Important
You cannot delete an IP address pool if there are allocations in it. You must first release the allocations and Deprovision CIDRs from a pool (p. 20) before you can delete the pool.

AWS Management Console

To delete a pool

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. From the dropdown menu at the top of the content pane, choose the scope that you want to use. For more information about scopes, see How IPAM works (p. 2).
4. In the content pane, choose the pool whose CIDR you want to delete.
5. Choose **Actions** > **Delete pool**.

6. Enter **`delete`** and then choose **Delete**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to delete a pool:

1. View pools and get an IPAM pool ID: describe-ipam-pools
2. Delete a pool: delete-ipam-pool
3. View your pools: describe-ipam-pools

To create a new pool, see Create a top-level pool (p. 10).

Create additional scopes

Follow the steps in this section to create an additional scope.

A scope is the highest-level container within IPAM. When you create an IPAM, IPAM creates two default scopes for you. Each scope represents the IP space for a single network. The private scope is intended for all private space. The public scope is intended for all public space. Scopes enable you to reuse IP addresses across multiple unconnected networks without causing IP address overlap or conflict.

When you create an IPAM, default scopes (one private and one public) are created for you. You can create additional private scopes. You cannot create additional public scopes.

You can create additional private scopes if you require support for multiple disconnected private networks. Additional private scopes allow you to create pools and manage resources that use the same IP space.

Important
If IPAM discovers resources with private IPv4 CIDRs, the resource CIDRs are imported into the default private scope and do not appear in any additional private scopes you create. You can move CIDRs from the default private scope to another private scope. For information, see Move resource CIDRs between scopes (p. 23).

AWS Management Console

To create an additional private scope

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Scopes**.
3. Choose **Create scope**.
4. Choose the IPAM that you want to add the scope to.
5. Add a description for the scope.
6. Choose **Create scope**.
7. You can view the scope in IPAM by choosing **Scopes** in the navigation pane.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to create an additional private scope:

1. View your current scopes: describe-ipam-scopes
2. Create a new private scope: create-ipam-scope
3. View your current scopes to view the new scope: describe-ipam-scopes

Move resource CIDRs between scopes

Follow the steps in this section to move a resource CIDR from one scope to another.

Important

- You can only move resource CIDRs from one private scope to another. You cannot move resource CIDRs out of a public scope to a private scope or from a private scope to a public scope.
- You can only move CIDRs for resources that IPAM can manage.
- The same AWS account must own both scopes.

AWS Management Console

To move a single CIDR allocated to a resource

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Resources**.
3. From the dropdown menu at the top of the content pane, choose the scope you want to use.
4. In the content pane, choose a resource and view the details of the resource.
5. Under **Associated CIDRs**, select one of the CIDRs allocated to the resource and choose **Actions** > **Move CIDR to different scope**.
6. Select the scope you want to move the resource CIDR to.
7. Choose **Change scope**.

To move all CIDRs allocated to a resource

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Resources**.
3. From the dropdown menu at the top of the content pane, choose the scope you want to use.
4. In the content pane, choose the resource whose CIDRs you want to move.
5. Choose **Actions** > **Move all associated CIDRs to different scope**.
6. Select the scope you want to move the resource CIDR to.
7. Choose **Move scope**.

Command line

Use the following AWS CLI commands to modify a pool:

1. Get an IPAM pool ID: describe-ipam-pools
2. Get a resource CIDR in current scope: get-ipam-pool-cidrs
3. Move a resource CIDR: modify-ipam-resource-cidr

4. Get a resource CIDR in the other scope: get-ipam-pool-cidrs

Change the monitoring state of resource CIDRs

Follow the steps in this section to change the monitoring state of a resource CIDR. You may want to change a resource CIDR from monitored to ignored if you do not want IPAM to manage or monitor the resource and allow the CIDR allocated to the resource to be available for use. You may want to change a resource CIDR from ignored to monitored if you want IPAM to manage and monitor the resource CIDR.

Note
You cannot ignore resources in the public scope.

You can change the monitoring state of a resource CIDR to monitored or ignored:

- **Monitored**: The resource CIDR has been detected by IPAM and is being monitored for overlap with other CIDRs and Allocation rule compliance.
- **Ignored**: The resource has been chosen to be exempt from monitoring. Ignored resources are not evaluated for overlap with other CIDRs or Allocation rule compliance. Once a resource is chosen to be ignored, any space allocated to it from an IPAM pool is returned to the pool and the resource will not be imported again via auto-import (if the auto-import Allocation rule is set on the pool).

AWS Management Console

To change the monitoring status of a single CIDR allocated to a resource

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Resources**.
3. From the dropdown menu at the top of the content pane, choose the private scope you want to use.
4. In the content pane, choose the resource and view the details of the resource.
5. Under **Associated CIDRs**, select one of the CIDRs allocated to the resource and choose **Actions** > **Mark as ignored** or **Unmark as ignored**.
6. Choose **Mark as ignored** or **Unmark as ignored**.

To change the monitoring status of all CIDRs allocated to a resource

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Resources**.
3. From the dropdown menu at the top of the content pane, choose the scope you want to use.
4. In the content pane, choose the resource whose monitoring state you want to change.
5. Choose **Actions** > **Mark all associated CIDRs as ignored** or **Unmark all associated CIDRs as ignored**.
6. Choose **Mark as ignored** or **Unmark as ignored**.

Command line

Use the following AWS CLI commands to change the monitoring state of a resource CIDR:

1. Get a scope ID: describe-ipam-scopes
2. View the current monitoring state for the resource: get-ipam-resource-cidrs
3. Change the state of the resource CIDR: modify-ipam-resource-cidr
4. View the new monitoring state for the resource: get-ipam-resource-cidrs

Delete a scope

Follow the steps in this section to delete an IPAM scope.

Important
You can't delete a scope if either of the following is true:

- The scope is a default scope. When you create an IPAM, two default scopes (one public, one private) are created automatically, and cannot be deleted. To see if a scope is a default scope, view the **Scope type** in the details of the scope.
- There are one or more pools in the scope. You must first Delete a pool (p. 21) before you can delete the scope.

AWS Management Console

To delete a scope

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Scopes**.
3. In the content pane, choose the scope that you want to delete.
4. Choose **Actions** > **Delete scope**.
5. Enter **`delete`** and then choose **Delete**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to delete a scope:

1. View scopes: describe-ipam-scopes
2. Delete a scope: delete-ipam-scope
3. View updated scopes: describe-ipam-scopes

To create a new scope, see Create additional scopes (p. 22). To delete the IPAM, see Delete an IPAM (p. 26).

Release an allocation

Follow the steps in this section to release a CIDR allocation from an IPAM pool. An allocation is a CIDR assignment from an IPAM pool to another resource or IPAM pool.

If you are planning to delete a pool, you might need to release a pool allocation. You cannot delete pools if the pools have CIDRs provisioned, and you cannot deprovision CIDRs if the CIDRs are allocated to resources.

Note

- To release a manual allocation, use the steps in this section or call the ReleaseIpamPoolAllocation API.
- To release an allocation in a private scope, you must ignore or delete the resource CIDR. For more information, see Change the monitoring state of resource CIDRs (p. 24). After some time, Amazon VPC IPAM will automatically release the allocation on your behalf.

Example

Example

If you have a VPC CIDR in a private scope, to release the allocation you must either ignore or delete the VPC CIDR. After some time, Amazon VPC IPAM will automatically release the VPC CIDR allocation from the IPAM pool.

- To release an allocation in a public scope, you must delete the resource CIDR. You cannot ignore public resource CIDRs. For more information, see *Cleanup* in Bring your own public IPv4 CIDR to IPAM using only the AWS CLI (p. 65) or *Cleanup* in Bring your own IPv6 CIDR to IPAM using only the AWS CLI (p. 79). After some time, Amazon VPC IPAM will automatically release the allocation on your behalf.

For Amazon VPC IPAM to release allocations on your behalf, all account permissions must be properly configured for either single-account use (p. 6) or multi-account use (p. 5).

When you release a CIDR that's managed by your IPAM, Amazon VPC IPAM recycles the CIDR back into an IPAM pool. It takes a few minutes for the CIDR to become available for future allocations. For more information about pools and allocations, see How IPAM works (p. 2).

AWS Management Console

To release a pool allocation

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. From the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2).
4. In the content pane, choose the pool that the allocation is in.
5. Choose the **Allocations** tab.
6. Select one or more allocations and choose **Deallocate CIDRs**.
7. Choose **Deallocate CIDR**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to release a pool allocation:

1. Get an IPAM pool ID: describe-ipam-pools
2. View your current allocations in the pool: get-ipam-pool-allocations
3. Release an allocation: release-ipam-pool-allocation
4. View your updated allocations: get-ipam-pool-allocations

To add a new allocation, see Allocate CIDRs (p. 15). To delete the pool after releasing allocations, you must first Deprovision CIDRs from a pool (p. 20).

Delete an IPAM

Follow the steps in this section to delete an IPAM. For information on increasing the default number of IPAMs you can have rather than deleting an existing IPAM, see Quotas for your IPAM (p. 101).

Important
Deleting an IPAM removes all monitored data associated with the IPAM including the historical data for CIDRs.

AWS Management Console

To delete an IPAM

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **IPAMs**.
3. In the content pane, select your IPAM.
4. Choose **Actions** > **Delete IPAM**.
5. Do one of the following:
   - Choose **Cascade delete** to delete the IPAM, private scopes, pools in private scopes, and any allocations in the pools in private scopes. You cannot delete the IPAM with this option if there is a pool in your public scope. If you use this option, IPAM does the following:
     - Deallocates any CIDRs allocated to VPC resources (such as VPCs) in pools in private scopes.

       Note
       No VPC resources are deleted as a result of enabling this option. The CIDR associated with the resource will no longer be allocated from an IPAM pool, but the CIDR itself will remain unchanged.
     - Deprovisions all IPv4 CIDRs provisioned to IPAM pools in private scopes.
     - Deletes all IPAM pools in private scopes.
     - Deletes all non-default private scopes in the IPAM.
     - Deletes the default public and private scopes and the IPAM.
   - If you don't choose the **Cascade delete** checkbox, before you can delete an IPAM, you must do the following:
     - Release allocations within the IPAM pools. For more information, see Release an allocation (p. 25).
     - Deprovision CIDRs provisioned to pools within the IPAM. For more information, see Deprovision CIDRs from a pool (p. 20).
     - Delete any additional non-default scopes. For more information, see Delete a scope (p. 25).
     - Delete your IPAM pools. For more information, see Delete a pool (p. 21).
6. Enter **`delete`** and then choose **Delete**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to delete an IPAM:

1. View current IPAMs: describe-ipams
2. Delete an IPAM: delete-ipam
3. View your updated IPAMs: describe-ipams

To create a new IPAM, see Create an IPAM (p. 7).

Tracking IP address usage in IPAM

The tasks described in this section are optional. If you want to complete the tasks in this section, and you have delegated an IPAM account, the tasks should be completed by the IPAM account.

Follow the steps in this section to track IP address usage with IPAM.

Contents

Monitor CIDR usage with the IPAM dashboard

Follow the steps in this section to access the IPAM dashboard and view the status of all CIDRs within a particular IPAM scope.

AWS Management Console

To monitor CIDR usage using the IPAM dashboard

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Dashboard**.
3. By default, when you view the dashboard, the default private scope is selected. If you don't want to use the default private scope, from the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2).
4. View the monitoring data in the following sections:

   - **Scope**: The details for this scope.
     - **Scope ID**: The ID for this scope.
     - **Description**: An optional description for the scope.
     - **IPAM ID**: The ID of the IPAM that the scope is in.
     - **Scope type**: The type of scope.
   - **Summary**: The number of CIDRs per category.
     - **Managed CIDRs**: The number of resource CIDRs for manageable resources (VPCs or public IPv4 pools) that are allocated from an IPAM pool in the scope.
     - **Unmanaged CIDRs**: The number of resource CIDRs for unmanaged resources in this scope.
     - **Ignored CIDRs**: The number of resource CIDRs that you have chosen to be exempt from monitoring with IPAM in the scope. IPAM does not evaluate ignored resources for overlap or compliance within a scope. When a resource is chosen to be ignored, any space that's allocated to it from an IPAM pool is returned to the pool, and the resource will not be imported again through automatic import (if the automatic import allocation rule is set on the pool).
     - **Pools**: The number of pools in the scope.

    - **Compliant CIDRs**: The number of resource CIDRs that comply with the allocation rules for IPAM pools in the scope.
    - **Overlapping CIDRs**: The number of resource CIDRs that overlap within a pool in the scope.
    - **Noncompliant CIDRs**: The number of resource CIDRs that do not comply with the allocation rules for the IPAM pools in the scope.
  - **Compliant vs. noncompliant CIDRs**: The number of compliant versus noncompliant CIDRs in the scope
  - **Overlapping CIDRs**: The number of CIDRs that currently overlap within the IPAM pools in this scope. Overlapping CIDRs can lead to incorrect routing in your VPCs.
  - **Pool assignment**: The percentage of IP space that has been assigned to resources and manual allocations in the scope.
  - **Pool allocation**: The percentage of a pool's IP space that has been allocated to other pools in the scope.

Command line

The information displayed in the dashboard comes from metrics stored in Amazon CloudWatch. Use the Amazon CloudWatch options in the AWS CLI Reference to view metrics for allocations in your IPAM pools and scopes.

If you find that the CIDR that's provisioned for a pool is almost fully allocated, you might need to provision additional CIDRs. For more information, see Provision CIDRs to a pool (p. 19).

Monitor CIDR usage by resource

In IPAM, a resource is an AWS service entity that is assigned an IP address or CIDR block. IPAM manages some resources, but only monitors other resources.

- **Managed resource**: A managed resource has a CIDR allocated from an IPAM pool. IPAM monitors the CIDR for potential IP address overlap with other CIDRs in the pool, and monitors the CIDR's compliance with a pool's allocation rules. IPAM supports managing the following type of resources:
  - VPCs
  - Public IPv4 pools

    Important
    Public IPv4 pools and IPAM pools are managed by distinct resources in AWS. Public IPv4 pools are single account resources that enable you to convert your publicly-owned CIDRs to Elastic IP addresses. IPAM pools can be used to allocate your public space to public IPv4 pools.
- **Monitored resource**: If a resource is monitored by IPAM, the resource has been detected by IPAM and you can view details about the resource's CIDR when you use `get-ipam-resource-cidrs` with the AWS CLI, or when you view **Resources** in the navigation pane. IPAM supports monitoring the following resources:
  - VPCs
  - Public IPv4 pools
  - VPC subnets
  - Elastic IP addresses
  - Subnet reserves

The following steps show you how to monitor CIDR usage and allocation rule compliance by resource.

AWS Management Console

To monitor CIDR usage by resource

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Resources**.
3. From the dropdown menu at the top of the content pane, choose the scope that you want to use. For more information about scopes, see How IPAM works (p. 2).
4. View the monitoring data in the following sections:

   - **Resource ID**: The ID for the scope.
   - **Management state**: The state of the resource.
     - **Managed**: The resource has a CIDR allocated from an IPAM pool and is being monitored by IPAM for potential CIDR overlap and compliance with pool allocation rules.
     - **Unmanaged**: The resource does not have a CIDR allocated from an IPAM pool and is not being monitored by IPAM for potential CIDR compliance with pool allocation rules. The CIDR is monitored for overlap.
     - **Ignored**: The managed resource has been chosen to be exempt from monitoring. Ignored resources are not evaluated for overlap or allocation rule compliance. When a resource is chosen to be ignored, any space allocated to it from an IPAM pool is returned to the pool and the resource will not be imported again through automatic import (if the automatic import allocation rule is set on the pool).
     - **-**: This resource is not one of the types of resources that IPAM can monitor or manage.
   - **Compliance status**: The compliance status of the CIDR.
     - **Compliant**: A managed resource complies with the allocation rules of the IPAM pool.
     - **Noncompliant**: The resource CIDR does not comply with one or more of the allocation rules of the IPAM pool.

       Example

       If a VPC has a CIDR that does not meet the netmask length parameters of the IPAM pool, or if the resource is not in the same AWS Region as the IPAM pool, it will be flagged as noncompliant.
     - **Unmanaged**: The resource does not have a CIDR allocated from an IPAM pool and is not being monitored by IPAM for potential CIDR compliance with pool allocation rules. The CIDR is monitored for overlap.
     - **Ignored**: The managed resource has been chosen to be exempt from monitoring. Ignored resources are not evaluated for overlap or allocation rule compliance. When a resource is chosen to be ignored, any space allocated to it from an IPAM pool is returned to the pool and the resource will not be imported again through automatic import (if the automatic import allocation rule is set on the pool).
     - **-**: This resource is not one of the types of resources that IPAM can monitor or manage.
   - **Overlap status**: The overlap status of CIDR.
     - **Nonoverlapping**: The resource CIDR does not overlap with another CIDR in the same scope.
     - **Overlapping**: The resource CIDR overlaps with another CIDR in the same scope. Note that if a resource CIDR is overlapping, it could be overlapping with a manual allocation.
     - **Ignored**: The managed resource has been chosen to be exempt from monitoring. IPAM does not evaluate ignored resources for overlap or allocation rule compliance. When a resource is chosen to be ignored, any space allocated to it from an IPAM pool is returned to the pool and the resource will not be imported again through automatic import (if the automatic import allocation rule is set on the pool).
     - **-**: This resource is not one of the types of resources that IPAM can monitor or manage.
   - **Resource name**: The name of the resource.

- **IP usage**: The percentage of IP address space in the resource that is in use.
- **CIDR**: The CIDR associated with the resource.
- **Region**: The AWS Region of the resource.
- **Owner ID**: The AWS account ID of the person that created this resource.
- **Pool ID**: The ID of the IPAM pool that the resource is in.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

Use the following AWS CLI commands to monitor CIDR usage by resource:

1. Get the scope ID: describe-ipam-scopes
2. Request resource information: get-ipam-resource-cidrs

Monitor IPAM with Amazon CloudWatch

IPAM automatically stores metrics related to IPAM IP address usage (such as the IP address space available in your IPAM pools and the number of resource CIDRs that comply with allocation rules) in the `AWS/IPAM` Amazon CloudWatch namespace in your IPAM's home Region. You can use these metrics to create alarms for IPAM pools to notify you if the address pools are nearing exhaustion or if resources fail to comply with allocation rules set on a pool. Creating alarms and setting up notifications is outside the scope of this User Guide. For more information, see Using Amazon CloudWatch alarms in the *Amazon CloudWatch User Guide*.

The metrics and dimensions that IPAM sends to Amazon CloudWatch are listed below.

IPAM Pool Metrics

Metric name	Description
CompliantResourceCidrs	The number of managed resource CIDRs that comply with the allocation rules of the IPAM pool. For more information about allocation rules, see Create a top-level pool (p. 10).
NoncompliantResourceCidrs	The number of managed resource CIDRs that do not comply with the allocation rules of the IPAM pool. For more information about allocation rules, see Create a top-level pool (p. 10).
PercentAllocated	The percentage of a pool's IP space that has been allocated to other pools.
PercentAssigned	The percentage of a pools IP space that has been allocated to resources, including manual allocations.
PercentAvailable	The percentage of a pool's IP space that has not been allocated to other pools or resources.

IPAM Scope Metrics

Metric name	Description
CompliantResourceCidrs	The number of resource CIDRs that comply with the allocation rules for IPAM pools in the scope.

Metric name	Description
ManagedResourceCidrs	The number of resource CIDRs for manageable resources (VPCs or public IPv4 pools) that are allocated from an IPAM pool in the scope.
NoncompliantResourceCidrs	The number of resource CIDRs that do not comply with the allocation rules for the IPAM pools in the scope.
OverlappingResourceCidrs	The number of resource CIDRs that overlap within a pool in the scope.
UnmanagedResourceCidrs	The number of resource CIDRs in the scope that are currently associated with manageable resources but are not managed by IPAM.

The dimensions you can use to filter IPAM metrics are listed below.

Dimension	Description
AddressFamily	The IP address family for resource CIDRs (IPv4 or IPv6).
Locale	The AWS Region where an IPAM pool is available for allocations.
PoolID	The ID of a pool.
ScopeID	The ID of a scope.

View IP address history

Follow the steps in this section to view the history of an IP address or CIDR in an IPAM scope. You can use the historical data to analyze and audit your network security and routing policies. IPAM automatically retains IP address monitoring data for up to three years.

You can use the IP historical data to search for the status change of IP addresses or CIDRs for the following types of resources:

- VPCs
- VPC subnets
- Elastic IP addresses
- EC2 instances
- EC2 network interfaces attached to instances

Important
Although IPAM doesn't monitor Amazon EC2 instances or EC2 network interfaces attached to instances, you can use the IP historical insights feature to search for historical data on EC2 instance and network interface CIDRs.

AWS Management Console

To view the history of a CIDR

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.

2. In the navigation pane, choose **IP historical insights**.
3. Enter an IPv4 or IPv6 IP address or CIDR. This must be a specific CIDR for the resource.
4. Choose an IPAM scope ID.

   Note
   If you move a resource from one IPAM scope to another, the previous history record ends and a new history record is created under the new scope.
5. Choose a date/time range.
6. If you want to filter the results by VPC, enter a VPC ID. Use this option if the CIDR appears in multiple VPCs.
7. Choose **Search**.

Command line

The commands in this section link to the AWS CLI Reference documentation. The documentation provides detailed descriptions of the options that you can use when you run the commands.

- View the history of a CIDR: get-ipam-address-history

To see examples of how you can use the AWS CLI to analyze and audit IP address usage, see Tutorial: View IP address history using the AWS CLI.

The results of the search are organized into the following columns:

- **Sampled end time**: Sampled end time of the resource-to-CIDR association within the IPAM scope. Changes are picked up in periodic snapshots, so the end time might have occurred before this specific time.
- **Sampled start time**: Sampled start time of the resource-to-CIDR association within the IPAM scope. Changes are picked up in periodic snapshots, so the start time might have occurred before this specific time.

  Example

  To help explain the times that you see under Sampled start time and Sampled end time, let's look at an example use case:

  At 2:00 PM, a VPC was created with CIDR 10.0.0.0/16. At 3:00 PM, you create an IPAM and IPAM pool with CIDR 10.0.0.0/8, and select the auto-import option to allow IPAM to discover and import any CIDRs that fall within the 10.0.0.0/8 IP address range. Because IPAM picks up changes to CIDRs in periodic snapshots, it doesn't discover the existing VPC CIDR until 3:05 PM. When you search for the ID of this VPC using the IP historical insights feature, the Sampled start time for your VPC is 3:05 PM, which is when IPAM discovered it, not 2:00 PM, which is when you created the VPC. Now, let's say that you decide to delete the VPC at 5:00 PM. When the VPC is deleted, the CIDR 10.0.0.0/16 that was allocated to the VPC is recycled back into the IPAM pool. IPAM takes its periodic snapshot at 5:05 PM and picks up the change. When you search for the ID of this VPC in IP historical insights, 5:05 PM is the Sampled end time for the VPC's CIDR, not 5:00 PM, which is when the VPC was deleted.
- **Resource ID**: The ID generated when the resource was associated with the CIDR.
- **Name:** The name of the resource (if applicable).
- **Compliance status**: The compliance status of the CIDR.
  - **Compliant**: A managed resource complies with the allocation rules of the IPAM pool.
  - **Noncompliant**: The resource CIDR does not comply with one or more of the allocation rules of the IPAM pool.

Example

If a VPC has a CIDR that does not meet the netmask length parameters of the IPAM pool, or if the resource is not in the same AWS Region as the IPAM pool, it will be flagged as noncompliant.

  - **Unmanaged**: The resource does not have a CIDR allocated from an IPAM pool and is not being monitored by IPAM for potential CIDR compliance with pool allocation rules. The CIDR is monitored for overlap.
  - **Ignored**: The managed resource has been chosen to be exempt from monitoring. Ignored resources are not evaluated for overlap or allocation rule compliance. When a resource is chosen to be ignored, any space allocated to it from an IPAM pool is returned to the pool and the resource will not be imported again through automatic import (if the automatic import allocation rule is set on the pool).
  - **-**: This resource is not one of the types of resources that IPAM can monitor or manage.
- **Overlap status**: The overlap status of CIDR.
  - **Nonoverlapping**: The resource CIDR does not overlap with another CIDR in the same scope.
  - **Overlapping**: The resource CIDR overlaps with another CIDR in the same scope. Note that if a resource CIDR is overlapping, it could be overlapping with a manual allocation.
  - **Ignored**: The managed resource has been chosen to be exempt from monitoring. IPAM does not evaluate ignored resources for overlap or allocation rule compliance. When a resource is chosen to be ignored, any space allocated to it from an IPAM pool is returned to the pool and the resource will not be imported again through automatic import (if the automatic import allocation rule is set on the pool).
  - **-**: This resource is not one of the types of resources that IPAM can monitor or manage.
- **Resource type**
  - **vpc**: The CIDR is associated with a VPC.
  - **subnet**: The CIDR is associated with a VPC subnet.
  - **eip**: The CIDR is associated with an Elastic IP address.
  - **instance**: The CIDR is associated with an EC2 instance.
  - **network-interface**: The CIDR is associated with a network interface.
- **VPC ID**: The ID of the VPC this resource belongs to (if applicable).
- **CIDR**: The CIDR that's associated with this resource.
- **Region**: The AWS Region of this resource.
- **Owner ID**: The AWS account ID of the user that created this resource (if applicable).

Tutorials

The following tutorials show you how to perform common IPAM tasks using the AWS CLI. To get the AWS CLI, see Access IPAM (p. 4). For more information on the IPAM concepts that are mentioned in these tutorials, see How IPAM works (p. 2).

Contents

Tutorial: Create an IPAM, create pools, and allocate a VPC using the AWS CLI

Follow the steps in this tutorial to use the AWS CLI to create an IPAM, create pools, and allocate a VPC.

The following is an example hierarchy of the pool structure that you will create by following the steps in this section:

- IPAM operating in AWS Region 1, AWS Region 2
  - Private scope
    - Top-level pool
      - Regional pool in AWS Region 2
        - Development pool
          - Allocation for a VPC

Note
In this section, you'll create an IPAM. By default, you can only create one IPAM. For more information, see Quotas for your IPAM (p. 101). If you have already delegated an IPAM account and created an IPAM, you can skip steps 1 and 2.

Contents

Step 1: Enable IPAM in your organization

This step is optional. Complete this step to enable IPAM in your organization and configure your delegated IPAM using the AWS CLI. For more information about the role of the IPAM account, see Integrate IPAM with AWS Organizations (p. 5).

This request must be made from an AWS Organizations management account. When you run the following command, ensure that you're using a role with an IAM policy that permits the following actions:

- `ec2:EnableIpamOrganizationAdminAccount`
- `organizations:EnableAwsServiceAccess`
- `organizations:RegisterDelegatedAdministrator`
- `iam:CreateServiceLinkedRole`

```
aws ec2 enable-ipam-organization-admin-account --region us-east-1 --delegated-admin-
account-id 11111111111
```

You should see the following output, indicating that enabling was successful.

```
{
    "Success": true
}
```

Step 2: Create an IPAM

Follow the steps in this section to create an IPAM and view additional information about the scopes that are created. You will use this IPAM when you create pools and provision IP address ranges for those pools in later steps.

Note
The operating Regions option determines which AWS Regions the IPAM pools can be used for. For more information about operating Regions, see Create an IPAM (p. 7).

To create an IPAM using the AWS CLI

1. Run the following command to create the IPAM instance.

   ```
   aws ec2 create-ipam --description my-ipam --region us-east-1 --operating-
   regions RegionName=us-west-2
   ```

   When you create an IPAM, AWS automatically does the following:

   - Returns a globally unique resource ID (`IpamId`) for the IPAM.
   - Creates a default public scope (`PublicDefaultScopeId`) and a default private scope (`PrivateDefaultScopeId`).

   ```
   {
       "Ipam": {
           "OwnerId": "123456789012",
           "IpamId": "ipam-0de83dba6694560a9",
   ```

```
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-0de83dba6694560a9",
        "PublicDefaultScopeId": "ipam-scope-02a24107598e982c5",
        "PrivateDefaultScopeId": "ipam-scope-065e7dfe880df679c",
        "ScopeCount": 2,
        "Description": "my-ipam",
        "OperatingRegions": [
            {
                "RegionName": "us-west-2"
            },
            {
                "RegionName": "us-east-1"
            }
        ],
        "Tags": []
    }
}
```

2. Run the following command to view additional information related to the scopes. The public scope is intended for IP addresses that are going to be accessed via public internet. The private scope is intended for IP addresses that are not going to be accessed via public internet.

```
aws ec2 describe-ipam-scopes --region us-east-1
```

In the output, you see the available scopes. You'll use the private scope ID in the next step.

```
{
    "IpamScopes": [
        {
            "OwnerId": "123456789012",
            "IpamScopeId": "ipam-scope-02a24107598e982c5",
            "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02a24107598e982c5",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-0de83dba6694560a9",
            "IpamScopeType": "public",
            "IsDefault": true,
            "PoolCount": 0
        },
        {
            "OwnerId": "123456789012",
            "IpamScopeId": "ipam-scope-065e7dfe880df679c",
            "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-065e7dfe880df679c",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-0de83dba6694560a9",
            "IpamScopeType": "private",
            "IsDefault": true,
            "PoolCount": 0
        }
    ]
}
```

Step 3: Create an IPv4 address pool

Follow the steps in this section to create an IPv4 address pool.

Important
You won't use the `--locale` option on this top-level pool. You will set the locale option later on the Regional pool. The locale is the AWS Region where you want a pool to be available for CIDR allocations. As a result of not setting the locale on the top-level pool, the locale will default to `None`. If a pool has a locale of `None`, the pool won't be available to VPC resources in any AWS Region. You can only manually allocate IP address space in the pool to reserve space.

To create an IPv4 address pool for all of your AWS resources using the AWS CLI

1. Run the following command to create an IPv4 address pool. Use the ID of the private scope of the IPAM that you created in the previous step.

```
aws ec2 create-ipam-pool --ipam-scope-id ipam-scope-065e7dfe880df679c --
description "top-level-pool" --address-family ipv4
```

   In the output, you'll see a state of `create-in-progress` for the pool.

```
{
    "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0008f25d7187a08d9",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0008f25d7187a08d9",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-065e7dfe880df679c",
        "IpamScopeType": "private",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-0de83dba6694560a9",
        "Locale": "None",
        "PoolDepth": 1,
        "State": "create-in-progress",
        "Description": "top-level-pool",
        "AutoImport": false,
        "AddressFamily": "ipv4",
        "Tags": []
    }
}
```

2. Run the following command until you see a state of `create-complete` in the output.

```
aws ec2 describe-ipam-pools
```

   The following example output shows the correct state.

```
{
    "IpamPools": [
        {
            "OwnerId": "123456789012",
            "IpamPoolId": "ipam-pool-0008f25d7187a08d9",
            "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0008f25d7187a08d9",
            "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-065e7dfe880df679c",
            "IpamScopeType": "private",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-0de83dba6694560a9",
            "Locale": "None",
            "PoolDepth": 1,
            "State": "create-complete",
            "Description": "top-level-pool",
            "AutoImport": false,
            "AddressFamily": "ipv4"
        }
    ]
}
```

Step 4: Provision a CIDR to the top-level pool

Follow the steps in this section to provision a CIDR to the top-level pool, and then verify that the CIDR is provisioned. For more information, see Provision CIDRs to a pool (p. 19).

To provision a CIDR block to the pool using the AWS CLI

1. Run the following command to provision the CIDR.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-0008f25d7187a08d9 --cidr 10.0.0.0/8
```

   In the output, you can verify the state of the provisioning.

```
{
    "IpamPoolCidr": {
        "Cidr": "10.0.0.0/8",
        "State": "pending-provision"
    }
}
```

2. Run the following command until you see a state of `provisioned` in the output.

```
aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-
pool-0008f25d7187a08d9
```

   The following example output shows the correct state.

```
{
    "IpamPoolCidrs": [
        {
            "Cidr": "10.0.0.0/8",
            "State": "provisioned"
        }
    ]
}
```

Step 5. Create a Regional pool with CIDR sourced from the top-level pool

When you create an IPAM pool, the pool belongs to the AWS Region of the IPAM by default. When you create a VPC, the pool that the VPC draws from must be in the same Region as the VPC. You can use the `--locale` option when you create a pool to make the pool available to services in a Region other than the Region of the IPAM. Follow the steps in this section to create a Regional pool in another locale.

To create a pool with a CIDR sourced from the previous pool using the AWS CLI

1. Run the following command to create the pool and insert space with a known available CIDR from the previous pool.

```
aws ec2 create-ipam-pool --description "regional--pool" --region us-east-1 --ipam-
scope-id ipam-scope-065e7dfe880df679c --source-ipam-pool-id
ipam-pool-0008f25d7187a08d9 --locale us-west-2 --address-family ipv4
```

In the output, you'll see the ID of the pool that you created. You'll need this ID in the next step.


```
{
    "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0da89c821626f1e4b",
        "SourceIpamPoolId": "ipam-pool-0008f25d7187a08d9",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0da89c821626f1e4b",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-065e7dfe880df679c",
        "IpamScopeType": "private",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-0de83dba6694560a9",
        "Locale": "us-west-2",
        "PoolDepth": 2,
        "State": "create-in-progress",
        "Description": "regional--pool",
        "AutoImport": false,
        "AddressFamily": "ipv4",
        "Tags": []
    }
}
```


2. Run the following command until you see a state of `create-complete` in the output.

```
aws ec2 describe-ipam-pools
```

In the output, you see the pools that you have in your IPAM. In this tutorial, we created a top-level and a Regional pool, so you'll see them both.


```
{
    "IpamPools": [
        {
            "OwnerId": "123456789012",
            "IpamPoolId": "ipam-pool-0008f25d7187a08d9",
            "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0008f25d7187a08d9",
            "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-065e7dfe880df679c",
            "IpamScopeType": "private",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-0de83dba6694560a9",
            "Locale": "None",
            "PoolDepth": 1,
            "State": "create-complete",
            "Description": "top-level-pool",
            "AutoImport": false,
            "AddressFamily": "ipv4"
        },
        {
            "OwnerId": "123456789012",
            "IpamPoolId": "ipam-pool-0da89c821626f1e4b",
            "SourceIpamPoolId": "ipam-pool-0008f25d7187a08d9",
            "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0da89c821626f1e4b",
            "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-065e7dfe880df679c",
            "IpamScopeType": "private",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-0de83dba6694560a9",
            "Locale": "us-west-2",
            "PoolDepth": 2,
            "State": "create-complete",
            "Description": "regional--pool",
```

```
            "AutoImport": false,
            "AddressFamily": "ipv4"
        }
    ]
}
```

Step 6: Provision a CIDR to the Regional pool

Follow the steps in this section to assign a CIDR block to the pool, and validate that it's been successfully provisioned.

To assign a CIDR block to the Regional pool using the AWS CLI

1. Run the following command to provision the CIDR.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-0da89c821626f1e4b --cidr 10.0.0.0/16
```

   In the output, you see the state of the pool.

```
{
    "IpamPoolCidr": {
        "Cidr": "10.0.0.0/16",
        "State": "pending-provision"
    }
}
```

2. Run the following command until you see the state of `provisioned` in the output.

```
aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-
pool-0da89c821626f1e4b
```

   The following example output shows the correct state.

```
{
    "IpamPoolCidrs": [
        {
            "Cidr": "10.0.0.0/16",
            "State": "provisioned"
        }
    ]
}
```

3. Run the following command to query the top-level pool to view the allocations. The Regional pool is considered an allocation within the top-level pool.

```
aws ec2 get-ipam-pool-allocations --region us-east-1 --ipam-pool-id ipam-
pool-041ff84c50166914f
```

   In the output, you see the Regional pool as an allocation in the top-level pool.

```
{
    "IpamPoolAllocations": [
        {
            "Cidr": "10.0.0.0/16",
            "IpamPoolAllocationId": "ipam-pool-alloc-fbd525f6c2bf4e77a75690fc2d93479a",
```

```
            "ResourceId": "ipam-pool-0da89c821626f1e4b",
            "ResourceType": "ipam-pool",
            "ResourceOwner": "123456789012"
        }
    ]
}
```

Step 7. Create a RAM share for enabling IP assignments across accounts

This step is optional. You can complete this step only if you completed Integrate IPAM with AWS Organizations (p. 5).

When you create an IPAM pool AWS RAM share, it enables IP assignments across accounts. RAM sharing is only available in your home AWS Region. Note that you create this share in the same Region as the IPAM, not in the local Region for the pool. All administrative operations on IPAM resources are made through the IPAM's home Region. The example in this tutorial creates a single share for a single pool, but you can add multiple pools to a single share. For more information, including an explanation of the options that you must enter, see Share an IPAM pool using AWS RAM (p. 18).

Run the following command to create a resource share.

```
aws ram create-resource-share --region us-east-1 --name pool_share --resource-arns
  arn:aws:ec2::123456789012:ipam-pool/ipam-pool-0dec9695bca83e606 --principals 123456
```

The output shows that the pool was created.

```
{
    "resourceShare": {
        "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/3ab63985-99d9-1cd2-7d24-75e93EXAMPLE",
        "name": "pool_share",
        "owningAccountId": "123456789012",
        "allowExternalPrincipals": false,
        "status": "ACTIVE",
        "creationTime": 1565295733.282,
        "lastUpdatedTime": 1565295733.282
    }
}
```

Step 8. Create a VPC

Run the following command to create a VPC and assign a CIDR block to the VPC from the pool in your newly created IPAM.

```
aws ec2 create-vpc --region us-east-1 --ipv4-ipam-pool-id
ipam-pool-04111dca0d960186e --cidr-block 10.0.0.0/24
```

The output shows that the VPC was created.

```
{
    "Vpc": {
        "CidrBlock": "10.0.0.0/24",
        "DhcpOptionsId": "dopt-19edf471",
```

```
        "State": "pending",
        "VpcId": "vpc-0983f3c454f3d8be5",
        "OwnerId": "123456789012",
        "InstanceTenancy": "default",
        "Ipv6CidrBlockAssociationSet": [],
        "CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-00b24cc1c2EXAMPLE",
                "CidrBlock": "10.0.0.0/24",
                "CidrBlockState": {
                    "State": "associated"
                }
            }
        ],
        "IsDefault": false
    }
}
```

Step 9. Cleanup

Follow the steps in this section to delete the IPAM resources you've created in this tutorial.

1. Delete the VPC.

```
aws ec2 delete-vpc --vpc-id vpc-0983f3c454f3d8be5
```

2. Delete the IPAM pool RAM share.

```
aws ram delete-resource-share --resource-share-arn arn:aws:ram:us-
west-2:123456789012:resource-share/3ab63985-99d9-1cd2-7d24-75e93EXAMPLE
```

3. Deprovision pool CIDR from the Regional pool.

```
 aws ec2 deprovision-ipam-pool-cidr --ipam-pool-id ipam-pool-0da89c821626f1e4b --
region us-east-1
```

4. Deprovision pool CIDR from the top-level pool.

```
 aws ec2 deprovision-ipam-pool-cidr --ipam-pool-id ipam-pool-0008f25d7187a08d9 --region
 us-east-1
```

5. Delete the IPAM

```
aws ec2 delete-ipam --region us-east-1
```

Tutorial: View IP address history using the AWS CLI

The scenarios in this section show you how to analyze and audit IP address usage using the AWS CLI. For general information about using the AWS CLI, see Using the AWS CLI in the *AWS Command Line Interface User Guide*.

Contents

Overview

IPAM automatically retains your IP address monitoring data for up to three years. You can use the historical data to analyze and audit your network security and routing policies. You can search for historical insights for the following types of resources:

- VPCs
- VPC subnets
- Elastic IP addresses
- EC2 instances that are running
- EC2 network interfaces attached to instances

Important
Although IPAM doesn't monitor Amazon EC2 instances or EC2 network interfaces attached to instances, you can use the IP historical insights feature to search for historical data on EC2 instance and network interface CIDRs.

Note

- The commands in this tutorial must be run using the account that owns the IPAM and the AWS Region that hosts the IPAM.
- Records of changes to CIDRs are picked up in periodic snapshots, which means that it can take some time for records to appear or be updated, and the values for SampledStartTime and SampledEndTime can differ from the actual times they occurred.

Scenarios

The scenarios in this section show you how to analyze and audit IP address usage using the AWS CLI. For more information about the values mentioned in this tutorial like sampled end time and start time, see View IP address history (p. 32).

Scenario 1: Which resources were associated with `10.2.1.155/32` between 1:00 AM and 9:00 PM on December 27, 2021 (UTC)?

1. Run the following command:

```
aws ec2 get-ipam-address-history --region us-east-1 --cidr 10.2.1.155/32 --ipam-
scope-id ipam-scope-05b579a1909c5fc7a --start-time 2021-12-20T01:00:00.000Z --end-
time 2021-12-27T21:00:00.000Z
```

2. View the results of the analysis. In the example below, the CIDR was allocated to a network interface and EC2 instance over the course of the time period. Note that no **SampledEndTime** value means the record is still active. For more information about the values shown in the following output, see View IP address history (p. 32).

```
{
    "HistoryRecords": [
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-1",
            "ResourceType": "network-interface",
            "ResourceId": "eni-0b4e53eb1733aba16",
            "ResourceCidr": "10.2.1.155/32",
            "VpcId": "vpc-0f5ee7e1ba908a378",
            "SampledStartTime": "2021-12-27T20:08:46.672000+00:00"
```

```
        },
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-1",
            "ResourceType": "instance",
            "ResourceId": "i-064da1f79baed14f3",
            "ResourceCidr": "10.2.1.155/32",
            "VpcId": "vpc-0f5ee7e1ba908a378",
            "SampledStartTime": "2021-12-27T20:08:46.672000+00:00"
        }
    ]
}
```

If the owner ID of the instance to which a network interface is attached differs from the owner ID of the network interface (as is the case for NAT gateways, Lambda network interfaces in VPCs, and other AWS services), the `ResourceOwnerId` is `amazon-aws` rather than the account ID of the owner of the network interface. The following example shows the record for a CIDR associated with a NAT gateway:

```
{
    "HistoryRecords": [
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-1",
            "ResourceType": "network-interface",
            "ResourceId": "eni-0b4e53eb1733aba16",
            "ResourceCidr": "10.0.0.176/32",
            "VpcId": "vpc-0f5ee7e1ba908a378",
            "SampledStartTime": "2021-12-27T20:08:46.672000+00:00"
        },
        {
            "ResourceOwnerId": "amazon-aws",
            "ResourceRegion": "us-east-1",
            "ResourceType": "instance",
            "ResourceCidr": "10.0.0.176/32",
            "VpcId": "vpc-0f5ee7e1ba908a378",
            "SampledStartTime": "2021-12-27T20:08:46.672000+00:00"
        }
    ]
}
```

Scenario 2: Which resources were associated with `10.2.1.0/24` from December 1, 2021 to December 27, 2021 (UTC)?

1. Run the following command:

```
aws ec2 get-ipam-address-history --region us-east-1 --cidr 10.2.1.0/24 --ipam-
scope-id ipam-scope-05b579a1909c5fc7a --start-time 2021-12-01T00:00:00.000Z --end-
time 2021-12-27T23:59:59.000Z
```

2. View the results of the analysis. In the example below, the CIDR was allocated to a subnet and VPC over the course of the time period. Note that no **SampledEndTime** value means the record is still active. For more information about the values shown in the following output, see View IP address history (p. 32).

```
{
    "HistoryRecords": [
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-1",
```

```
            "ResourceType": "subnet",
            "ResourceId": "subnet-0864c82a42f5bffed",
            "ResourceCidr": "10.2.1.0/24",
            "VpcId": "vpc-0f5ee7e1ba908a378",
            "SampledStartTime": "2021-12-27T20:08:46.672000+00:00"
        },
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-1",
            "ResourceType": "vpc",
            "ResourceId": "vpc-0f5ee7e1ba908a378",
            "ResourceCidr": "10.2.1.0/24",
            "ResourceComplianceStatus": "compliant",
            "ResourceOverlapStatus": "nonoverlapping",
            "VpcId": "vpc-0f5ee7e1ba908a378",
            "SampledStartTime": "2021-12-27T20:08:46.672000+00:00"
        }
    ]
}
```

Scenario 3: Which resources were associated with `2605:9cc0:409::/56` from December 1, 2021 to December 27, 2021 (UTC)?

1. Run the following command, where --region is the IPAM home Region:

```
aws ec2 get-ipam-address-history --region us-east-1 --cidr 2605:9cc0:409::/56 --ipam-
scope-id ipam-scope-07cb485c8b4a4d7cc --start-time 2021-12-01T01:00:00.000Z --end-
time 2021-12-27T23:59:59.000Z
```

2. View the results of the analysis. In the example below, the CIDR was allocated to two different VPCs over the course of the time period in a Region outside the IPAM home Region. Note that no **SampledEndTime** value means the record is still active. For more information about the values shown in the following output, see View IP address history (p. 32).

```
{
    "HistoryRecords": [
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-2",
            "ResourceType": "vpc",
            "ResourceId": "vpc-01d967bf3b923f72c",
            "ResourceCidr": "2605:9cc0:409::/56",
            "ResourceName": "First example VPC",
            "ResourceComplianceStatus": "compliant",
            "ResourceOverlapStatus": "nonoverlapping",
            "VpcId": "vpc-01d967bf3b923f72c",
            "SampledStartTime": "2021-12-23T20:02:00.701000+00:00",
            "SampledEndTime": "2021-12-23T20:12:59.848000+00:00"
        },
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-2",
            "ResourceType": "vpc",
            "ResourceId": "vpc-03e62c7eca81cb652",
            "ResourceCidr": "2605:9cc0:409::/56",
            "ResourceName": "Second example VPC",
            "ResourceComplianceStatus": "compliant",
            "ResourceOverlapStatus": "nonoverlapping",
            "VpcId": "vpc-03e62c7eca81cb652",
            "SampledStartTime": "2021-12-27T15:11:00.046000+00:00"
        }
    ]
```

```
}
```

Scenario 4: Which resources were associated with `10.0.0.0/24` in the last 24 hours (assuming the current time is midnight on December 27, 2021 (UTC))?

1. Run the following command:

```
aws ec2 get-ipam-address-history --region us-east-1 --cidr 10.0.0.0/24 --ipam-scope-
id ipam-scope-05b579a1909c5fc7a --start-time 2021-12-27T00:00:00.000Z
```

2. View the results of the analysis. In the example below, the CIDR has been allocated to numerous subnets and VPCs over the time period. Note that no **SampledEndTime** value means the record is still active. For more information about the values shown in the following output, see View IP address history (p. 32).


```
{
    "HistoryRecords": [
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-2",
            "ResourceType": "subnet",
            "ResourceId": "subnet-0d1b8f899725aa72d",
            "ResourceCidr": "10.0.0.0/24",
            "ResourceName": "Example name",
            "VpcId": "vpc-042b8a44f64267d67",
            "SampledStartTime": "2021-12-11T16:35:59.074000+00:00",
            "SampledEndTime": "2021-12-28T15:34:00.017000+00:00"
        },
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-2",
            "ResourceType": "vpc",
            "ResourceId": "vpc-09754dfd85911abec",
            "ResourceCidr": "10.0.0.0/24",
            "ResourceName": "Example name",
            "ResourceComplianceStatus": "unmanaged",
            "ResourceOverlapStatus": "overlapping",
            "VpcId": "vpc-09754dfd85911abec",
            "SampledStartTime": "2021-12-27T20:07:59.947000+00:00",
            "SampledEndTime": "2021-12-28T15:34:00.017000+00:00"
        },
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-west-2",
            "ResourceType": "vpc",
            "ResourceId": "vpc-0a8347f594bea5901",
            "ResourceCidr": "10.0.0.0/24",
            "ResourceName": "Example name",
            "ResourceComplianceStatus": "unmanaged",
            "ResourceOverlapStatus": "overlapping",
            "VpcId": "vpc-0a8347f594bea5901",
            "SampledStartTime": "2021-12-11T16:35:59.318000+00:00"
        },
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-1",
            "ResourceType": "subnet",
            "ResourceId": "subnet-0af7eadb0798e9148",
            "ResourceCidr": "10.0.0.0/24",
            "ResourceName": "Example name",
            "VpcId": "vpc-03298ba16756a8736",
            "SampledStartTime": "2021-12-14T21:07:22.357000+00:00"
```

```
        }
    ]
}
```

Scenario 5: Which resources are currently associated with `10.2.1.155/32`?

1. Run the following command:

```
aws ec2 get-ipam-address-history --region us-east-1 --cidr 10.2.1.155/32 --ipam-scope-
id ipam-scope-05b579a1909c5fc7a
```

2. View the results of the analysis. In the example below, the CIDR was allocated to a network interface and EC2 instance over the time period. Note that no **SampledEndTime** value means the record is still active. For more information about the values shown in the following output, see View IP address history (p. 32).

```
{
    "HistoryRecords": [
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-1",
            "ResourceType": "network-interface",
            "ResourceId": "eni-0b4e53eb1733aba16",
            "ResourceCidr": "10.2.1.155/32",
            "VpcId": "vpc-0f5ee7e1ba908a378",
            "SampledStartTime": "2021-12-27T20:08:46.672000+00:00"
        },
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-1",
            "ResourceType": "instance",
            "ResourceId": "i-064da1f79baed14f3",
            "ResourceCidr": "10.2.1.155/32",
            "VpcId": "vpc-0f5ee7e1ba908a378",
            "SampledStartTime": "2021-12-27T20:08:46.672000+00:00"
        }
    ]
}
```

Scenario 6: Which resources are currently associated with `10.2.1.0/24`?

1. Run the following command:

```
aws ec2 get-ipam-address-history --region us-east-1 --cidr 10.2.1.0/24 --ipam-scope-
id ipam-scope-05b579a1909c5fc7a
```

2. View the results of the analysis. In the example below, the CIDR was allocated to a VPC and subnet over the time period. Only the results that match this exact `/24` CIDR are returned, not all `/32` within the `/24` CIDR. Note that no **SampledEndTime** value means the record is still active. For more information about the values shown in the following output, see View IP address history (p. 32).

```
{
    "HistoryRecords": [
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-1",
            "ResourceType": "subnet",
            "ResourceId": "subnet-0864c82a42f5bffed",
```

```
            "ResourceCidr": "10.2.1.0/24",
            "VpcId": "vpc-0f5ee7e1ba908a378",
            "SampledStartTime": "2021-12-27T20:08:46.672000+00:00"
        },
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-1",
            "ResourceType": "vpc",
            "ResourceId": "vpc-0f5ee7e1ba908a378",
            "ResourceCidr": "10.2.1.0/24",
            "ResourceComplianceStatus": "compliant",
            "ResourceOverlapStatus": "nonoverlapping",
            "VpcId": "vpc-0f5ee7e1ba908a378",
            "SampledStartTime": "2021-12-27T20:08:46.672000+00:00"
        }
    ]
}
```


Scenario 7: Which resources are currently associated with `54.0.0.9/32`?

In this example, `54.0.0.9/32` is assigned to an Elastic IP address that is not part of the AWS Organization integrated with your IPAM.

1. Run the following command:

```
aws ec2 get-ipam-address-history --region us-east-1 --cidr 54.0.0.9/32 --ipam-scope-
id ipam-scope-05b579a1909c5fc7a
```

2. Since `54.0.0.9/32` is assigned to an Elastic IP address that is not part of the AWS Organization integrated with the IPAM in this example, no records are returned.

```
{
    "HistoryRecords": []
}
```

Tutorial: BYOIP address CIDRs to IPAM

The tutorials in this section walk you through the process of bringing public IP address space to AWS and managing the space with IPAM.

Managing public IP address space with IPAM has the following benefits:

- **Improves public IP addresses utilization across your organization**: You can use IPAM to share IP address space across AWS accounts. Without using IPAM, you cannot share your public IP space across AWS Organizations accounts.
- **Simplifies the process of bringing public IP space to AWS**: You can use IPAM to onboard public IP address space once, and then use IPAM to distribute your public IPs across Regions. Without IPAM, you have to onboard your public IPs for each AWS Region.

Important
To complete the steps in this tutorial, you first need to complete the following steps using the *Amazon EC2 User Guide for Linux Instances* for the CIDR range you want to bring to AWS and IPAM. Once you complete these steps, continue with this tutorial:

1. Create a key pair and certificate.

2. Create an ROA object in your RIR.

   When you create the ROAs, for IPv4 CIDRs you must set the maximum length of an IP address prefix to `/24`. For IPv6 CIDRs, if you are adding them to an advertisable pool, the maximum length of an IP address prefix must be `/48`. This ensures that you have full flexibility to divide your public IP address across AWS Regions. IPAM enforces the maximum length you set. The maximum length is the smallest prefix length announcement you will allow for this route. For example, if you bring a `/20` CIDR block to AWS, by setting the maximum length to `/24`, you can divide the larger block any way you like (such as with `/21`, `/22`, or `/24`) and distribute those smaller CIDR blocks to any Region. If you were to set the maximum length to `/23`,you would not be able to divide and advertise a `/24` from the larger block. Also, note that `/24` is the smallest IPv4 block and `/48` is the smallest IPv6 block you can advertise from a Region to the internet.
3. Update the RDAP record in your RIR.

Contents

Bring your own public IPv4 CIDR to IPAM using both the AWS Management Console and the AWS CLI

Follow these steps to bring an IPv4 or IPv6 CIDR to IPAM using both the AWS Management Console and the AWS CLI.

Important
To complete the steps in this tutorial, you first need to complete the following steps using the *Amazon EC2 User Guide for Linux Instances* for the CIDR range you want to bring to AWS and IPAM. Once you complete these steps, continue with this tutorial:

1. Create a key pair and certificate.
2. Create an ROA object in your RIR.

   When you create the ROAs, for IPv4 CIDRs you must set the maximum length of an IP address prefix to `/24`. For IPv6 CIDRs, if you are adding them to an advertisable pool, the maximum length of an IP address prefix must be `/48`. This ensures that you have full flexibility to divide your public IP address across AWS Regions. IPAM enforces the maximum length you set. The maximum length is the smallest prefix length announcement you will allow for this route. For example, if you bring a `/20` CIDR block to AWS, by setting the maximum length to `/24`, you can divide the larger block any way you like (such as with `/21`, `/22`, or `/24`) and distribute those smaller CIDR blocks to any Region. If you were to set the maximum length to `/23`,you would not be able to divide and advertise a `/24` from the larger block. Also, note that `/24` is the smallest IPv4 block and `/48` is the smallest IPv6 block you can advertise from a Region to the internet.
3. Update the RDAP record in your RIR.

Contents

Bring your own IPv4 CIDR to IPAM using both the AWS Management Console and the AWS CLI

Follow these steps to bring an IPv4 CIDR to IPAM and allocate an Elastic IP address (EIP) using both the AWS Management Console and the AWS CLI.

Important

- This tutorial assumes you have already completed the steps in the following sections:
  - Integrate IPAM with AWS Organizations (p. 5).
  - Create an IPAM (p. 7).
- Each step of this tutorial must be done by one of three AWS Organizations accounts:
  - The management account.
  - The member account configured to be your IPAM administrator in Integrate IPAM with AWS Organizations (p. 5). In this tutorial, this account will be called the IPAM account.
  - The member account in your organization which will allocate CIDRs from an IPAM pool. In this tutorial, this account will be called the member account.

Contents

Step 1: Create AWS CLI named profiles

To complete this tutorial as a single AWS user, you can use AWS CLI named profiles to switch from one AWS account to another. Named profiles are collections of IAM access key IDs and secret access keys that you store locally and then refer to using the `--profile` option when you use the AWS CLI. For more information about how to create or retrieve IAM access keys for AWS accounts, see Managing access keys for IAM users in the *AWS Identity and Access Management User Guide*.

Complete the steps in Creating named profiles in the *AWS Command Line Interface User Guide* to create one named profiles for each of the three AWS accounts you will use in this tutorial:

- A profile called `management-account` for the AWS Organizations management account.
- A profile called `ipam-account` for the AWS Organizations member account that is configured to be your IPAM administrator.
- A profile called `member-account` for the AWS Organizations member account in your organization which will allocate CIDRs from an IPAM pool.

Once you have created the named profiles, return to this page and go to the next step. You will notice throughout the rest of this tutorial that the sample AWS CLI commands use the `--profile` option with one of the named profiles to indicate which account must run the command.

Step 2: Create a top-level IPAM pool

Complete the steps in this section to create a top-level IPAM pool.

This step must be done by the IPAM account.

To create a pool

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. Choose the public scope. For more information about scopes, see How IPAM works (p. 2).
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a **Description** for the pool.
6. Under **Source pool**, choose **No source pool**.
7. Under **Address family**, choose **IPv4**.
8. Under **Locale**, choose **None**.

   The locale is the AWS Region where you want this IPAM pool to be available for allocations. For example, you can only allocate a CIDR for a VPC from an IPAM pool that shares a locale with the VPC's Region. Note that when you have chosen a locale for a pool, you cannot modify it.

   The IPAM integration with BYOIP requires that the locale is set on whichever pool will be used for the BYOIP CIDR. Since we are going to create a top-level IPAM pool with a Regional pool within it, and we're going to allocate space to an Elastic IP address from the Regional pool, you will set the locale on the Regional pool and not the top-level pool. You'll add the locale to the Regional pool when you create the Regional pool in a later step.

   Note
   If you are creating a single pool only and not a top-level pool with Regional pools within it, you would want to choose a Locale for this pool so that the pool is available for allocations.
9. Under **CIDRs to provision**, choose a CIDR to provision for the pool. Note that when provisioning an IPv4 CIDR to a pool within the top-level pool, the minimum IPv4 CIDR you can provision is `/24`; more specific CIDRs (such as `/25`) are not permitted. You must include the CIDR and the BYOIP message and certificate signature in the request so we can verify that you own the public space. For a list of BYOIP prerequisites including how to get this BYOIP message and certificate signature, see Bring your own public IPv4 CIDR to IPAM using both the AWS Management Console and the AWS CLI (p. 50).
10. Leave **Use this pool to allocate CIDRs to resources such as VPCs** unchecked.
11. (Optional) Choose **Tags** for the pool.
12. Choose **Create pool**.

Ensure that this CIDR has been provisioned before you continue. You can see the state of provisioning in the **CIDRs** tab in the pool details page. Note that it can take up to one week for the BYOIP CIDR to be provisioned.

Step 3. Create a Regional pool within the top-level pool

Create a Regional pool within the top-level pool. The IPAM integration with BYOIP requires that the locale is set on whichever pool will be used for the BYOIP CIDR. You'll add the locale to the Regional pool

when you create the Regional pool in this section. The `Locale` must be one of the operating Regions you configured when you created the IPAM.

This step must be done by the IPAM account.

To create a Regional pool within a top-level pool

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. If you don't want to use the default private scope, from the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2).
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a **Description** for the pool.
6. Under **Source pool**, choose the top-level pool that you created in the previous section.
7. Under **Locale**, choose the locale for the pool. In this tutorial, we'll use `us-east-2` as the locale for the Regional pool. The available options come from the operating Regions that you chose when you created your IPAM.

   The locale is the AWS Region where you want this IPAM pool to be available for allocations. For example, you can only allocate a CIDR for a VPC from an IPAM pool that shares a locale with the VPC's Region. Note that when you have chosen a locale for a pool, you cannot modify it. Choosing a locale ensures there are no cross-region dependencies between your pool and the resources allocating from it.
8. Under **Service**, choose **EC2 (EIP/VPC)**.
9. Under **CIDRs to provision**, choose a CIDR to provision for the pool. Note that when provisioning a CIDR to a pool within the top-level pool, the minimum IPv4 CIDR you can provision is `/24`; more specific CIDRs (such as `/25`) are not permitted.
10. Choose **Use this pool to allocate CIDRs to resources such as VPCs**. You have the same allocation rule options here as you did when you created the top-level pool. See Create a top-level pool (p. 10) for an explanation of the options that are available when you create pools. The allocation rules for the Regional pool are not inherited from the top-level pool. If you do not apply any rules here, there will be no allocation rules set for the pool.
11. Choose **Use this pool to allocate CIDRs to resources such as VPCs** and choose optional allocation rules for this pool:

    - **Automatically import discovered resources**: This option is not available if the **Locale** is set to **None**. If this option is selected, IPAM will continuously look for resources within the CIDR range of this pool and automatically import them as allocations into your IPAM. Note the following:
      - The CIDRs that will be allocated for these resources must not already be allocated to other resources in order for the import to succeed.
      - IPAM will import a CIDR regardless of its compliance with the pool's allocation rules, so a resource might be imported and subsequently marked as noncompliant.
      - If IPAM discovers multiple CIDRs that overlap, IPAM will import the largest CIDR only.
      - If IPAM discovers multiple CIDRs with matching CIDRs, IPAM will randomly import one of them only.
    - **Minimum netmask length**: The minimum netmask length required for CIDR allocations in this IPAM pool to be compliant and the largest size CIDR block that can be allocated from the pool. The minimum netmask length must be less than the maximum netmask length. Possible netmask lengths for IPv4 addresses are `0 - 32`. Possible netmask lengths for IPv6 addresses are `0 - 128`.
    - **Default netmask length**: A default netmask length for allocations added to this pool. For example, if the CIDR that's provisioned to this pool is **`10.0.0.0/8`** and you enter **`16`** here, any new allocations in this pool will default to a netmask length of `/16`.

- **Maximum netmask length**: The maximum netmask length that will be required for CIDR allocations in this pool. This value dictates the smallest size CIDR block that can be allocated from the pool.
- **Tagging requirements**: The tags that are required for resources to allocate space from the pool. If the resources have their tags changed after they have allocated space or if the allocation tagging rules are changed on the pool, the resource may be marked as noncompliant.
- **Locale**: The locale that will be required for resources that use CIDRs from this pool. Automatically imported resources that do not have this locale will be marked noncompliant. Resources that are not automatically imported into the pool will not be allowed to allocate space from the pool unless they are in this locale.

12. (Optional) Choose **Tags** for the pool.
13. When you've finished configuring your pool, choose **Create pool**.

Ensure that this CIDR has been provisioned before you continue. You can see the state of provisioning in the **CIDRs** tab in the pool details page.

Step 4: Enable resource sharing with AWS Organizations using AWS RAM

You will use AWS RAM to share your Regional pool with the AWS Organizations member account who would like to allocate a CIDR from the Regional pool for an Elastic IP address (EIP). Before you can do that, you must enable RAM integration with AWS Organizations.

Complete the steps in Enable resource sharing within AWS Organizations in the *AWS RAM User Guide* using the management account. If you are using the AWS CLI to enable resource sharing, use the `--profile` **`management-account`** option. Once resource sharing is enabled in RAM, go to the next step in this tutorial.

Step 5: Share your Regional pool with an AWS Organizations member account using AWS RAM

Complete the process in Share an IPAM pool using AWS RAM (p. 18) and share the Regional pool with the AWS Organizations member account.

This step must be done by the IPAM account. If you are using the AWS CLI to share the pool, use the `--profile` **`ipam-account`** option.

Important
When you create the resource share, ensure the following:

- The principal is the account ID of the member account who will be allocating a CIDR from the pool for the Elastic IP address.
- You assign the *AWSRAMPermissionIpamPoolByoipCidrImport* permission to the pool.

Step 6: Create a public IPv4 pool

Creating a public IPv4 pool is a required step for bringing a public IPv4 address to AWS to be managed with IPAM. This step should be done by the member account that will provision an Elastic IP address.

This step must be done by the member account using the AWS CLI.

Important
Public IPv4 pools and IPAM pools are managed by distinct resources in AWS. Public IPv4 pools are single account resources that enable you to convert your publicly-owned CIDRs to Elastic IP addresses. IPAM pools can be used to allocate your public space to public IPv4 pools.

To create a public IPv4 pool using the AWS CLI

- Run the following command to provision the CIDR. When you run the command in this section, the value for `--region` must match the `Locale` option you chose when you created the pool that will be used for the BYOIP CIDR.

```
aws ec2 create-public-ipv4-pool --region us-east-2 --profile member-account
```

In the output, you'll see the public IPv4 pool ID. You will need this ID in the next step.

```
{
    "PoolId": "ipv4pool-ec2-09037ce61cf068f9a"
}
```

Step 7: Provision the public IPv4 CIDR to your public IPv4 pool

Provision the public IPv4 CIDR to your public IPv4 pool. The value for `--region` must match the `Locale` value you chose when you created the pool that will be used for the BYOIP CIDR.

This step must be done by the member account using the AWS CLI.

To create a public IPv4 pool using the AWS CLI

1. Run the following command to provision the CIDR.

```
aws ec2 provision-public-ipv4-pool-cidr --region us-east-2 --ipam-pool-id ipam-
pool-04d8e2d9670eeab21 --pool-id ipv4pool-ec2-09037ce61cf068f9a --netmask-length 24 --
profile member-account
```

In the output, you'll see the provisioned CIDR.

```
{
    "PoolId": "ipv4pool-ec2-09037ce61cf068f9a",
    "PoolAddressRange": {
        "FirstAddress": "130.137.245.0",
        "LastAddress": "130.137.245.255",
        "AddressCount": 256,
        "AvailableAddressCount": 256
    }
}
```

2. Run the following command to view the CIDR provisioned in the public IPv4 pool.

```
aws ec2 describe-byoip-cidrs --region us-east-2 --max-results 10 --profile member-
account
```

In the output, you'll see the provisioned CIDR. By default the CIDR is not advertised, which means it's not publicly accessible over the internet. You will have the chance to set this CIDR to advertised in the last step of this tutorial.

```
{
    "ByoipCidrs": [
        {
            "Cidr": "130.137.245.0/24",
            "StatusMessage": "Cidr successfully provisioned",
            "State": "provisioned"
        }
```

```
    ]
}
```

Once you create the public IPv4 pool, to view the public IPv4 pool allocated in the IPAM Regional pool, open the IPAM console and view the allocation in the Regional pool under **Allocations** or **Resources**.

Step 8: Create an Elastic IP address from the public IPv4 pool

Complete the steps in Allocate an Elastic IP address in the *Amazon EC2 User Guide for Linux Instances* to create an Elastic IP address (EIP) from the public IPv4 pool. When you open EC2 in the AWS Management console, the AWS Region you allocate the EIP in must match the `Locale` option you chose when you created the pool that will be used for the BYOIP CIDR.

This step must be done by the member account. If you are using the AWS CLI, use the `--profile` **`member-account`** option.

Step 9: Associate the Elastic IP address with an EC2 instance

Complete the steps in Associate an Elastic IP address with an instance or network interface in the *Amazon EC2 User Guide for Linux Instances* to associate the EIP with an EC2 instance. When you open EC2 in the AWS Management console, the AWS Region you associate the EIP in must match the `Locale` option you chose when you created the pool that will be used for the BYOIP CIDR. In this tutorial, that pool is the Regional pool.

This step must be done by the member account. If you are using the AWS CLI, use the `--profile` **`member-account`** option.

Step 10: Advertise the CIDR

The steps in this section must be done by the IPAM account. Once you associate the Elastic IP address (EIP) with an instance or Elastic Load Balancer, you can then start advertising the CIDR you brought to AWS that is in pool that has the **Service EC2 (EIP/VPC)** configured. In this tutorial, that's your Regional pool. By default the CIDR is not advertised, which means it's not publicly accessible over the internet.

This step must be done by the IPAM account.

To advertise the CIDR

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. Choose the public scope. For more information about scopes, see How IPAM works (p. 2).
4. Choose the Regional pool you created in this tutorial.
5. Choose the **CIDRs** tab.
6. Select the BYOIP CIDR and choose **Actions** > **Advertise**.
7. Choose **Advertise CIDR**.

As a result, the BYOIP CIDR is advertised and the value in the **Advertising** column changes from **Withdrawn** to **Advertised**.

Step 11: Cleanup

Follow the steps in this section to clean up the resources you've provisioned and created in this tutorial.

Step 1: Withdraw the CIDR from advertising

This step must be done by the IPAM account.

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. Choose the public scope.
4. Choose the Regional pool you created in this tutorial.
5. Choose the **CIDRs** tab.
6. Select the BYOIP CIDR and choose **Actions** > **Withdraw from advertising**.
7. Choose **Withdraw CIDR**.

As a result, the BYOIP CIDR is no longer advertised and the value in the **Advertising** column changes from **Advertised** to **Withdrawn**.

Step 2: Disassociate the Elastic IP address

This step must be done by the member account. If you are using the AWS CLI, use the `--profile` **`member-account`** option.

- Complete the steps in Disassociate an Elastic IP address in the *Amazon EC2 User Guide for Linux Instances* to disassociate the EIP. When you open EC2 in the AWS Management console, the AWS Region you disassociate the EIP in must match the `Locale` option you chose when you created the pool that will be used for the BYOIP CIDR. In this tutorial, that pool is the Regional pool.

Step 3: Release the Elastic IP address

This step must be done by the member account. If you are using the AWS CLI, use the `--profile` **`member-account`** option.

- Complete the steps in Release an Elastic IP address in the *Amazon EC2 User Guide for Linux Instances* to release an Elastic IP address (EIP) from the public IPv4 pool. When you open EC2 in the AWS Management console, the AWS Region you allocate the EIP in must match the `Locale` option you chose when you created the pool that will be used for the BYOIP CIDR.

Step 4: Deprovision the public IPv4 CIDR from your public IPv4 pool

This step must be done by the member account using the AWS CLI.

1. View your BYOIP CIDRs.

```
aws ec2 describe-public-ipv4-pools --region us-east-2 --profile member-account
```

In the output, you'll see the IP addresses in your BYOIP CIDR.

```
{
    "PublicIpv4Pools": [
        {
            "PoolId": "ipv4pool-ec2-09037ce61cf068f9a",
            "Description": "",
            "PoolAddressRanges": [
                {
                    "FirstAddress": "130.137.245.0",
                    "LastAddress": "130.137.245.255",
                    "AddressCount": 256,
                    "AvailableAddressCount": 256
                }
            ],
            "TotalAddressCount": 256,
            "TotalAvailableAddressCount": 256,
```

```
            "NetworkBorderGroup": "us-east-2",
            "Tags": []
        }
    ]
}
```

2. Run the following command to release the last IP address in the CIDR from the public IPv4 pool. Enter the IP address with a netmask of `/32`.

```
aws ec2 deprovision-public-ipv4-pool-cidr --region us-east-2 --pool-id ipv4pool-
ec2-09037ce61cf068f9a --cidr 130.137.245.255/32 --profile member-account
```

In the output, you'll see the deprovisioned CIDR.

```
{
    "PoolId": "ipv4pool-ec2-09037ce61cf068f9a",

    "DeprovisionedAddresses": [

        "130.137.245.255"

    ]

}
```

Important
You must rerun this command for each IP address in the CIDR range. If your CIDR is a `/24`, you will have to run this command to deprovision each of the 256 IP addresses in the `/24` CIDR.

3. View your BYOIP CIDRs again and ensure there are no more provisioned addresses. When you run the command in this section, the value for `--region` must match the Region of your IPAM.

```
aws ec2 describe-public-ipv4-pools --region us-east-2 --profile member-account
```

In the output, you'll see the IP addresses count in your public IPv4 pool.

```
{
    "PublicIpv4Pools": [
        {
            "PoolId": "ipv4pool-ec2-09037ce61cf068f9a",
            "Description": "",
            "PoolAddressRanges": [],
            "TotalAddressCount": 0,
            "TotalAvailableAddressCount": 0,
            "NetworkBorderGroup": "us-east-2",
            "Tags": []
        }
    ]
}
```

Note
It can take some time for IPAM to discover that public IPv4 pool allocations have been removed. You cannot continue to clean up and deprovision the IPAM pool CIDR until you see that the allocation has been removed from IPAM.

Step 5: Delete the public IPv4 pool

This step must be done by the member account.

- Run the following command to delete the public IPv4 pool the CIDR. When you run the command in this section, the value for `--region` must match the `Locale` option you chose when you created the pool that will be used for the BYOIP CIDR. In this tutorial, that pool is the Regional pool. This step must be done using the AWS CLI.

```
aws ec2 delete-public-ipv4-pool --region us-east-2 --pool-id ipv4pool-
ec2-09037ce61cf068f9a --profile member-account
```

  In the output, you'll see the return value **true**.

```
{
"ReturnValue": true
}
```

  Once you delete the pool, to view the allocation unmanaged by IPAM, open the IPAM console and view the details of the Regional pool under **Allocations**.

Step 6: Delete the RAM share and disable RAM integration with AWS Organizations

This step must be done by the IPAM account and management account respectively. If you are using the AWS CLI to delete the RAM share and disable RAM integration, use the `--profile ipam-account` and `--profile management-account` options.

- Complete the steps in Deleting a resource share in AWS RAM and Disabling resource sharing with AWS Organizations in the *AWS RAM User Guide*, in that order, to delete the RAM share and disable RAM integration with AWS Organizations.

Step 7: Deprovision the CIDRs from the Regional pool and top-level pool

This step must be done by the IPAM account. If you are using the AWS CLI to share the pool, use the `--profile ipam-account` option.

- Complete the steps in Deprovision CIDRs from a pool (p. 20) to deprovision the CIDRs from the Regional pool and then the top-level pool, in that order.

Step 8: Delete the Regional pool and top-level pool

This step must be done by the IPAM account. If you are using the AWS CLI to share the pool, use the `--profile ipam-account` option.

- Complete the steps in Delete a pool (p. 21) to delete the Regional pool and then the top-level pool, in that order.

Bring your own IPv6 CIDR to IPAM using the AWS Management Console

Follow the steps in this tutorial to bring an IPv6 CIDR to IPAM and allocate a VPC with the CIDR using both the AWS Management Console and the AWS CLI.

Important

- This tutorial assumes you have already completed the steps in the following sections:
  - Integrate IPAM with AWS Organizations (p. 5).
  - Create an IPAM (p. 7).

- Each step of this tutorial must be done by one of three AWS Organizations accounts:
  - The management account.
  - The member account configured to be your IPAM administrator in Integrate IPAM with AWS Organizations (p. 5). In this tutorial, this account will be called the IPAM account.
  - The member account in your organization which will allocate CIDRs from an IPAM pool. In this tutorial, this account will be called the member account.

Contents

Step 1: Create a top-level IPAM pool

Since you are going to create a top-level IPAM pool with a Regional pool within it, and we're going to allocate space to a resource (an Elastic IP address) from the Regional pool, you will set the locale on the Regional pool and not the top-level pool. You'll add the locale to the Regional pool when you create the Regional pool in a later step. The IPAM integration with BYOIP requires that the locale is set on whichever pool will be used for the BYOIP CIDR.

This step must be done by the IPAM account.

To create a pool

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. Choose the public scope. For more information about scopes, see How IPAM works (p. 2).
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a **Description** for the pool.
6. Under **Source pool**, choose **No source pool**.
7. Under **Address family**, choose **IPv6**.
8. Ensure **Allow CIDRs in this pool to be publicly advertisable** is selected.
9. Under **Locale**, choose **None**. You will set the locale on the Regional pool.

   The locale is the AWS Region where you want this IPAM pool to be available for allocations. For example, you can only allocate a CIDR for a VPC from an IPAM pool that shares a locale with the VPC's Region. Note that when you have chosen a locale for a pool, you cannot modify it.

   Note
   If you are creating a single pool only and not a top-level pool with Regional pools within it, you would want to choose a Locale for this pool so that the pool is available for allocations.
10. Under **CIDRs to provision**, choose a CIDR to provision for the pool. Note that when provisioning an IPv6 CIDR to a pool within the top-level pool, the minimum IPv6 CIDR you can provision for an advertisable IPAM pool is `/48`; more specific CIDRs (such as `/49`) are not permitted. The minimum

CIDR you can bring in for an non-advertisable IPAM pool is `/56`; more specific CIDRs (such as `/57`) are not permitted. You must include the CIDR and the BYOIP message and certificate signature in the request so we can verify that you own the public space. For a list of BYOIP prerequisites including how to get this BYOIP message and certificate signature, see Bring your own public IPv4 CIDR to IPAM using both the AWS Management Console and the AWS CLI (p. 50).

11. Leave **Use this pool to allocate CIDRs to resources such as VPCs** unchecked.
12. (Optional) Choose **Tags** for the pool.
13. Choose **Create pool**.

Ensure that this CIDR has been provisioned before you continue. You can see the state of provisioning in the **CIDRs** tab in the pool details page. Note that it can take up to one week for the BYOIP CIDR to be provisioned.

Step 2. Create a Regional pool within the top-level pool

Create a Regional pool within the top-level pool. A Locale is required on the pool and it must be one of the operating Regions you configured when you created the IPAM.

This step must be done by the IPAM account.

To create a Regional pool within a top-level pool

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. If you don't want to use the default private scope, from the dropdown menu at the top of the content pane, choose the scope you want to use. For more information about scopes, see How IPAM works (p. 2).
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a description for the pool.
6. Under **Source pool**, choose the top-level pool that you created in the previous section.
7. Choose the locale for the pool. Choosing a locale ensures there are no cross-region dependencies between your pool and the resources allocating from it. The available options come from the operating Regions that you chose when you created your IPAM. In this tutorial, we'll use `us-east-2` as the locale for the Regional pool.

   The locale is the AWS Region where you want this IPAM pool to be available for allocations. For example, you can only allocate a CIDR for a VPC from an IPAM pool that shares a locale with the VPC's Region. Note that when you have chosen a locale for a pool, you cannot modify it.
8. Under **Service**, choose **EC2 (EIP/VPC)**.
9. Under **CIDRs to provision**, choose a CIDR to provision for the pool. Note that when provisioning an IPv6 CIDR to a pool within the top-level pool, the minimum IPv6 CIDR you can provision for an advertisable IPAM pool is `/48`; more specific CIDRs (such as `/49`) are not permitted. The minimum CIDR you can bring in for an non-advertisable IPAM pool is `/56`; more specific CIDRs (such as `/57`) are not permitted.
10. Choose **Use this pool to allocate CIDRs to resources such as VPCs** and choose optional allocation rules for this pool:

    - **Automatically import discovered resources**: This option is not available if the **Locale** is set to **None**. If selected, IPAM will continuously look for resources within the CIDR range of this pool and automatically import them as allocations into your IPAM. Note the following:
      - The CIDRs that will be allocated for these resources must not already be allocated to other resources in order for the import to succeed.

- IPAM will import a CIDR regardless of its compliance with the pool's allocation rules, so a resource might be imported and subsequently marked as noncompliant.
- If IPAM discovers multiple CIDRs that overlap, IPAM will import the largest CIDR only.
- If IPAM discovers multiple CIDRs with matching CIDRs, IPAM will randomly import one of them only.

- **Minimum netmask length**: The minimum netmask length required for CIDR allocations in this IPAM pool to be compliant and the largest size CIDR block that can be allocated from the pool. The minimum netmask length must be less than the maximum netmask length. Possible netmask lengths for IPv4 addresses are `0 - 32`. Possible netmask lengths for IPv6 addresses are `0 - 128`.
- **Default netmask length**: A default netmask length for allocations added to this pool.
- **Maximum netmask length**: The maximum netmask length that will be required for CIDR allocations in this pool. This value dictates the smallest size CIDR block that can be allocated from the pool. Ensure that this value is minimum **`/48`**.
- **Tagging requirements**: The tags that are required for resources to allocate space from the pool. If the resources have their tags changed after they have allocated space or if the allocation tagging rules are changed on the pool, the resource may be marked as noncompliant.
- **Locale**: The locale that will be required for resources that use CIDRs from this pool. Automatically imported resources that do not have this locale will be marked noncompliant. Resources that are not automatically imported into the pool will not be allowed to allocate space from the pool unless they are in this locale.

11. (Optional) Choose **Tags** for the pool.
12. When you've finished configuring your pool, choose **Create pool**.

Ensure that this CIDR has been provisioned before you continue. You can see the state of provisioning in the **CIDRs** tab in the pool details page.

Step 3: Enable resource sharing with AWS Organizations using AWS RAM

You will use AWS RAM to share your Regional pool with the AWS Organizations member account who would like to allocate a CIDR from the Regional pool for a VPC. Before you can do that, you must enable RAM integration with AWS Organizations.

The management account must complete the steps in Enable resource sharing within AWS Organizations in the *AWS RAM User Guide* before you continue with this tutorial. Once resource sharing is enabled in RAM, go to the next step in this tutorial.

Step 4: Share your Regional pool with an AWS Organizations member account using AWS RAM

Complete the process in Share an IPAM pool using AWS RAM (p. 18) and share the Regional pool with the AWS Organizations member account.

This step must be done by the IPAM account.

Important
When you create the resource share, ensure the following:

- The principal is the account ID of the member account who will be allocating a CIDR from the pool.
- You assign the *AWSRAMPermissionIpamPoolByoipCidrImport* permission to the pool.

Step 5: Create a VPC

Complete the steps in Creating a VPC in the *Amazon VPC User Guide*.

This step must be done by the member account.

Note

- When you open VPC in the AWS Management console, the AWS Region you create the VPC in must match the `Locale` option you chose when you created the pool that will be used for the BYOIP CIDR.
- When you reach the step to choose a CIDR for the VPC, you will have an option to use a CIDR from an IPAM pool. Choose the Regional pool you created in this tutorial.

When you create the VPC, AWS allocates a CIDR in the IPAM pool to the VPC. You can view the allocation in IPAM by choosing a pool in the content pane of the IPAM console and viewing the **Allocations** tab for the pool.

Step 6: Advertise the CIDR

The steps in this section must be done by the IPAM account. Once you create the VPC, you can then start advertising the CIDR you brought to AWS that is in the pool that has the **Service EC2 (EIP/VPC)** configured. In this tutorial, that's your Regional pool. By default the CIDR is not advertised, which means it's not publicly accessible over the internet.

This step must be done by the IPAM account.

To advertise the CIDR

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. Choose the public scope. For more information about scopes, see How IPAM works (p. 2).
4. Choose the Regional pool you created in this tutorial.
5. Choose the **CIDRs** tab.
6. Select the BYOIP CIDR and choose **Actions** > **Advertise**.
7. Choose **Advertise CIDR**.

As a result, the BYOIP CIDR is advertised and the value in the **Advertising** column changes from **Withdrawn** to **Advertised**.

Step 7: Cleanup

Follow the steps in this section to clean up the resources you've provisioned and created in this tutorial.

Step 1: Withdraw the CIDR from advertising

This step must be done by the IPAM account.

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. By default, when you create a pool, the default private scope is selected. Choose the public scope.
4. Choose the Regional pool you created in this tutorial.
5. Choose the **CIDRs** tab.
6. Select the BYOIP CIDR and choose **Actions** > **Withdraw from advertising**.

7. Choose **Withdraw CIDR**.

As a result, the BYOIP CIDR is no longer advertised and the value in the **Advertising** column changes from **Advertised** to **Withdrawn**.

Step 2: Delete the VPC

This step must be done by the member account.

- Complete the steps in Deleting a VPC in the *Amazon VPC User Guide* to delete the VPC. When you open VPC in the AWS Management console, the AWS Region delete the VPC from must match the `Locale` option you chose when you created the pool that will be used for the BYOIP CIDR. In this tutorial, that pool is the Regional pool.

  When you delete the VPC, it takes time for IPAM to discover that the resource has been deleted and to deallocate the CIDR allocated to the VPC. You cannot continue to the next step in the cleanup until you see that IPAM has removed the allocation from the pool in the pool details **Allocations** tab.

Step 3: Delete the RAM share and disable RAM integration with AWS Organizations

This step must be done by the IPAM account and management account respectively.

- Complete the steps in Deleting a resource share in AWS RAM and Disabling resource sharing with AWS Organizations in the *AWS RAM User Guide*, in that order, to delete the RAM share and disable RAM integration with AWS Organizations.

Step 4: Deprovision the CIDRs from the Regional pool and top-level pool

This step must be done by the IPAM account.

- Complete the steps in Deprovision CIDRs from a pool (p. 20) to deprovision the CIDRs from the Regional pool and then the top-level pool, in that order.

Step 5: Delete the Regional pool and top-level pool

This step must be done by the IPAM account.

- Complete the steps in Delete a pool (p. 21) to delete the Regional pool and then the top-level pool, in that order.

Bring your own public IPv4 CIDR to IPAM using only the AWS CLI

Follow these steps to bring an IPv4 or IPv6 CIDR to IPAM using only the AWS CLI.

Important
To complete the steps in this tutorial, you first need to complete the following steps using the *Amazon EC2 User Guide for Linux Instances* for the CIDR range you want to bring to AWS and IPAM. Once you complete these steps, continue with this tutorial:

1. Create a key pair and certificate.
2. Create an ROA object in your RIR.

   When you create the ROAs, for IPv4 CIDRs you must set the maximum length of an IP address prefix to `/24`. For IPv6 CIDRs, if you are adding them to an advertisable pool,

the maximum length of an IP address prefix must be `/48`. This ensures that you have full flexibility to divide your public IP address across AWS Regions. IPAM enforces the maximum length you set. The maximum length is the smallest prefix length announcement you will allow for this route. For example, if you bring a `/20` CIDR block to AWS, by setting the maximum length to `/24`, you can divide the larger block any way you like (such as with `/21`, `/22`, or `/24`) and distribute those smaller CIDR blocks to any Region. If you were to set the maximum length to `/23`,you would not be able to divide and advertise a `/24` from the larger block. Also, note that `/24` is the smallest IPv4 block and `/48` is the smallest IPv6 block you can advertise from a Region to the internet.

3. Update the RDAP record in your RIR.

Contents

Bring your own public IPv4 CIDR to IPAM using only the AWS CLI

Follow these steps to bring an IPv4 CIDR to IPAM and allocate an Elastic IP address (EIP) with the CIDR using only the AWS CLI.

Important

- This tutorial assumes you have already completed the steps in the following sections:
  - Integrate IPAM with AWS Organizations (p. 5).
  - Create an IPAM (p. 7).
- Each step of this tutorial must be done by one of three AWS Organizations accounts:
  - The management account.
  - The member account configured to be your IPAM administrator in Integrate IPAM with AWS Organizations (p. 5). In this tutorial, this account will be called the IPAM account.
  - The member account in your organization which will allocate CIDRs from an IPAM pool. In this tutorial, this account will be called the member account.

Contents

Step 1: Create AWS CLI named profiles

To complete this tutorial as a single AWS user, you can use AWS CLI named profiles to switch from one AWS account to another. Named profiles are collections of IAM access key IDs and secret access keys that you store locally and then refer to using the `--profile` option when you use the AWS CLI. For more information about how to create or retrieve IAM access keys for AWS accounts, see Managing access keys for IAM users in the *AWS Identity and Access Management User Guide*.

Complete the steps in Creating named profiles in the *AWS Command Line Interface User Guide* to create one named profiles for each of the three AWS accounts you will use in this tutorial:

- A profile called `management-account` for the AWS Organizations management account.
- A profile called `ipam-account` for the AWS Organizations member account that is configured to be your IPAM administrator.
- A profile called `member-account` for the AWS Organizations member account in your organization which will allocate CIDRs from an IPAM pool.

Once you have created the named profiles, return to this page and go to the next step. You will notice throughout the rest of this tutorial that the sample AWS CLI commands use the `--profile` option with one of the named profiles to indicate which account must run the command.

Step 2: Create an IPAM

This step is optional. If you already have an IPAM created with operating Regions of `us-east-1` and `us-west-2` created, you can skip this step. Create an IPAM and specify an operating region of `us-east-1` and `us-west-2` . You must select an operating region so that you can use the locale option when you create your IPAM pool. The IPAM integration with BYOIP requires that the locale is set on whichever pool will be used for the BYOIP CIDR.

This step must be done by the IPAM account.

Run the following command:

```
aws ec2 create-ipam --description my-ipam --region us-east-1 --operating-
regions RegionName=us-west-2 --profile ipam-account
```

In the output, you'll see the IPAM you've created. Note the value for `PublicDefaultScopeId`. You will need your public scope ID in the next step. You are using the public scope because BYOIP CIDRs are public IP addresses, which is what the public scope is meant for.

```
{
 "Ipam": {
        "OwnerId": "123456789012",
        "IpamId": "ipam-090e48e75758de279",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "PublicDefaultScopeId": "ipam-scope-0087d83896280b594",
        "PrivateDefaultScopeId": "ipam-scope-08b70b04fbd524f8d",
        "ScopeCount": 2,
        "Description": "my-ipam",
        "OperatingRegions": [
            {
                "RegionName": "us-east-1"
            },
            {
                "RegionName": "us-west-2"
            }
        ],
        "Tags": []
    }
```

```
}
```

Step 3: Create a top-level IPAM pool

Complete the steps in this section to create a top-level IPAM pool.

This step must be done by the IPAM account.

To create an IPv4 address pool for all of your AWS resources using the AWS CLI

1. Run the following command to create an IPAM pool. Use the ID of the public scope of the IPAM that you created in the previous step.

   This step must be done by the IPAM account.

   ```
   aws ec2 create-ipam-pool --region us-east-1 --ipam-scope-id ipam-
   scope-0087d83896280b594 --description "top-level-IPv4-pool" --address-family ipv4 --
   profile ipam-account
   ```

   In the output, you'll see `create-in-progress`, which indicates that pool creation is in progress.

   ```
   {
       "IpamPool": {
           "OwnerId": "123456789012",
           "IpamPoolId": "ipam-pool-0a03d430ca3f5c035",
           "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
   pool-0a03d430ca3f5c035",
           "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
   scope-0087d83896280b594",
           "IpamScopeType": "public",
           "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
           "Locale": "None",
           "PoolDepth": 1,
           "State": "create-in-progress",
           "Description": "top-level-pool",
           "AutoImport": false,
           "AddressFamily": "ipv4",
           "Tags": []
       }
   }
   ```

2. Run the following command until you see a state of `create-complete` in the output.

   ```
   aws ec2 describe-ipam-pools --region us-east-1 --profile ipam-account
   ```

   The following example output shows the state of the pool.

   ```
   {
       "IpamPools": [
           {
               "OwnerId": "123456789012",
               "IpamPoolId": "ipam-pool-0a03d430ca3f5c035",
               "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
   pool-0a03d430ca3f5c035",
               "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
   scope-0087d83896280b594",
               "IpamScopeType": "public",
               "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
               "Locale": "None",
               "PoolDepth": 1,
               "State": "create-complete",
   ```

```
            "Description": "top-level-IPV4-pool",
            "AutoImport": false,
            "AddressFamily": "ipv4",
            "Tags": []
        }
    ]
}
```

Step 4: Provision a CIDR to the top-level pool

Provision a CIDR block to the top-level pool. Note that when provisioning an IPv4 CIDR to a pool within the top-level pool, the minimum IPv4 CIDR you can provision is `/24`; more specific CIDRs (such as `/25`) are not permitted. You must include the CIDR and the BYOIP message and certificate signature in the request so we can verify that you own the public space. For a list of BYOIP prerequisites including how to get this BYOIP message and certificate signature, see Bring your own public IPv4 CIDR to IPAM using only the AWS CLI (p. 64).

This step must be done by the IPAM account.

Important
You only need to add `--cidr-authorization-context` when you provision the BYOIP CIDR to the top-level pool. For the Regional pool within the top-level pool, you can omit the `--cidr-authorization-context` option. Once you onboard your BYOIP to IPAM, you are not required to perform ownership validation when you divide the BYOIP across Regions and accounts.

To provision a CIDR block to the pool using the AWS CLI

1. Run the following command to provision the CIDR.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-0a03d430ca3f5c035 --cidr 130.137.245.0/24 --cidr-authorization-
context Message="1|aws|470889052444|130.137.245.0/24|20250101|SHA256|
RSAPSS",Signature="W3gdQ9PZHLjPmrnGM~cvGx~KCIsMaU0P7ENO7VRnfSuf9NuJU5RUveQzus~QmF~Nx42j3z7d65uyZZiD
hApR89Kt6GxRYOdRaNx8yt~uoZWzxct2yIhWngy-
du9pnEHBOX6WhoGYjWszPw0iV4cmaAX9DuMs8ASR83K127VvcBcRXElT5URr3gWEB1CQe3rmuyQk~gAdbXiDN~94-
oS9AZlafBbrFxRjFWRCTJhc7Cg3ASbRO~VWNci-
C~bWAPczbX3wPQSjtWGV3k1bGuD26ohUc02o8oJZQyYXRpgqcWGVJdQ__" --profile ipam-account
```

   In the output, you'll see the CIDR pending provision.

```
{
    "IpamPoolCidr": {

        "Cidr": "130.137.245.0/24",

        "State": "pending-provision"

    }

}
```

2. Ensure that this CIDR has been provisioned before you continue. Note that it can take up to one week for the BYOIP CIDR to be provisioned. Run the following command until you see a state of `provisioned` in the output.

```
aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-
pool-0a03d430ca3f5c035 --profile ipam-account
```

   The following example output shows the state.

```
{
    "IpamPoolCidrs": [
        {
            "Cidr": "130.137.245.0/24",
            "State": "provisioned"
        }
    ]
}
```

Step 5: Create a Regional pool within the top-level pool

Create a Regional pool within the top-level pool. `--locale` is required on the pool and it must be one of the operating Regions you configured when you created the IPAM. The locale is the AWS Region where you want this IPAM pool to be available for allocations. For example, you can only allocate a CIDR for a VPC from an IPAM pool that shares a locale with the VPC's Region. Note that when you have chosen a locale for a pool, you cannot modify it.

This step must be done by the IPAM account.

Choosing a locale ensures there are no cross-region dependencies between your pool and the resources allocating from it. The available options come from the operating Regions that you chose when you created your IPAM. In this tutorial, we'll use `us-west-2` as the locale for the Regional pool.

To create a Regional pool using the AWS CLI

1. Run the following command to create the pool.

```
aws ec2 create-ipam-pool --description "Regional-IPv4-pool" --region us-east-1 --ipam-
scope-id ipam-scope-0087d83896280b594 --source-ipam-pool-id ipam-pool-0a03d430ca3f5c035
 --locale us-west-2 --address-family ipv4 --aws-service ec2 --profile ipam-account
```

   In the output, you'll see IPAM creating the pool.

```
{
     "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0d8f3646b61ca5987",
        "SourceIpamPoolId": "ipam-pool-0a03d430ca3f5c035",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0d8f3646b61ca5987",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
        "IpamScopeType": "public",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "Locale": "us-west-2",
        "PoolDepth": 2,
        "State": "create-in-progress",
        "Description": "Regional--pool",
        "AutoImport": false,
        "AddressFamily": "ipv4",
        "Tags": [],
        "ServiceType": "ec2"
    }
}
```

2. Run the following command until you see a state of `create-complete` in the output.

```
aws ec2 describe-ipam-pools --region us-east-1 --profile ipam-account
```

In the output, you see the pools that you have in your IPAM. In this tutorial, we created a top-level and a Regional pool, so you'll see them both.

Step 6: Provision a CIDR to the Regional pool

Provision a CIDR block to the Regional pool. Note that when provisioning a CIDR to a pool within the top-level pool, the minimum IPv4 CIDR you can provision is `/24`; more specific CIDRs (such as `/25`) are not permitted.

This step must be done by the IPAM account.

To assign a CIDR block to the Regional pool using the AWS CLI

1. Run the following command to provision the CIDR.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-0d8f3646b61ca5987 --cidr 130.137.245.0/24 --profile ipam-account
```

   In the output, you'll see the CIDR pending provision.

```
{
    "IpamPoolCidr": {

        "Cidr": "130.137.245.0/24",

        "State": "pending-provision"

    }

}
```

2. Run the following command until you see the state of `provisioned` in the output.

```
aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-
pool-0d8f3646b61ca5987 --profile ipam-account
```

   The following example output shows the correct state.

```
{
    "IpamPoolCidrs": [
        {
            "Cidr": "130.137.245.0/24",
            "State": "provisioned"
        }
    ]
}
```

Step 7: Enable resource sharing with AWS Organizations using AWS RAM

You will use AWS RAM to share your Regional pool with the AWS Organizations member account who would like to allocate a CIDR from the Regional pool for a VPC. Before you can do that, you must enable RAM integration with AWS Organizations.

Complete the steps in Enable resource sharing within AWS Organizations in the *AWS RAM User Guide* using the management account. If you are using the AWS CLI to enable resource sharing, use the ––

`profile` **`management-account`** option. Once resource sharing is enabled in RAM, go to the next step in this tutorial.

Step 8: Share your Regional pool with an AWS Organizations member account using AWS RAM

Complete the process in Share an IPAM pool using AWS RAM (p. 18) and share the Regional pool with the AWS Organizations member account.

This step must be done by the IPAM account. If you are using the AWS CLI to share the pool, use the `--profile` **`ipam-account`** option.

Important
When you create the resource share, ensure the following:

- The principal is the account ID of the member account who will be allocating a CIDR from the pool for the Elastic IP address.
- You assign the *AWSRAMPermissionIpamPoolByoipCidrImport* permission to the pool.

Step 9: Create a public IPv4 pool

Creating a public IPv4 pool is a required step for bringing a public IPv4 address to AWS to be managed with IPAM. This step would typically be done by a different AWS account which wants to provision an Elastic IP address.

This step must be done by the member account.

Important
Public IPv4 pools and IPAM pools are managed by distinct resources in AWS. Public IPv4 pools are single account resources that enable you to convert your publicly-owned CIDRs to Elastic IP addresses. IPAM pools can be used to allocate your public space to public IPv4 pools.

To create a public IPv4 pool using the AWS CLI

- Run the following command to provision the CIDR. When you run the command in this section, the value for `--region` must match the `--locale` option you entered when you created the pool that will be used for the BYOIP CIDR.

```
aws ec2 create-public-ipv4-pool --region us-west-2 --profile member-account
```

In the output, you'll see the public IPv4 pool ID. You will need this ID in the next step.

```
{
    "PoolId": "ipv4pool-ec2-0019eed22a684e0b2"
}
```

Step 10: Provision the public IPv4 CIDR to your public IPv4 pool

Provision the public IPv4 CIDR to your public IPv4 pool. The value for `--region` must match the `--locale` value you entered when you created the pool that will be used for the BYOIP CIDR.

This step must be done by the member account.

To create a public IPv4 pool using the AWS CLI

1. Run the following command to provision the CIDR.

```
aws ec2 provision-public-ipv4-pool-cidr --region us-west-2 --ipam-pool-id ipam-
pool-0d8f3646b61ca5987 --pool-id ipv4pool-ec2-0019eed22a684e0b2 --netmask-length 24 --
profile member-account
```

In the output, you'll see the provisioned CIDR.

```
{
    "PoolId": "ipv4pool-ec2-0019eed22a684e0b2",
    "PoolAddressRange": {
        "FirstAddress": "130.137.245.0",
        "LastAddress": "130.137.245.255",
        "AddressCount": 256,
        "AvailableAddressCount": 256
    }
}
```

2. Run the following command to view the CIDR provisioned in the public IPv4 pool.

```
aws ec2 describe-byoip-cidrs --region us-west-2 --max-results 10 --profile member-
account
```

In the output, you'll see the provisioned CIDR. By default the CIDR is not advertised, which means it's not publicly accessible over the internet. You will have the chance to set this CIDR to advertised in the last step of this tutorial.

```
{
    "ByoipCidrs": [
        {
            "Cidr": "130.137.245.0/24",
            "StatusMessage": "Cidr successfully provisioned",
            "State": "provisioned"
        }
    ]
}
```

Step 11: Create an Elastic IP address from the public IPv4 pool

Create an Elastic IP address (EIP) from the public IPv4 pool. When you run the commands in this section, the value for `--region` must match the `--locale` option you entered when you created the pool that will be used for the BYOIP CIDR.

This step must be done by the member account.

To create an EIP from the public IPv4 pool using the AWS CLI

1. Run the following command to create the EIP.

```
aws ec2 allocate-address  --region us-west-2 --public-ipv4-pool ipv4pool-
ec2-0019eed22a684e0b2 --profile member-account
```

In the output, you'll see the allocation.

```
{
    "PublicIp": "130.137.245.100",
    "AllocationId": "eipalloc-0db3405026756dbf6",
    "PublicIpv4Pool": "ipv4pool-ec2-0019eed22a684e0b2",
```

```
    "NetworkBorderGroup": "us-east-1",
    "Domain": "vpc"
}
```

2. Run the following command to view the EIP allocation managed in IPAM.

   This step must be done by the IPAM account.

```
aws ec2 get-ipam-pool-allocations --region us-west-2 --ipam-pool-id ipam-
pool-0d8f3646b61ca5987 --profile ipam-account
```

   The output shows the allocation in IPAM.

```
{
    "IpamPoolAllocations": [
        {
            "Cidr": "130.137.245.0/24",
            "IpamPoolAllocationId": "ipam-pool-alloc-5dedc8e7937c4261b56dc3e3eb53dc45",
            "ResourceId": "ipv4pool-ec2-0019eed22a684e0b2",
            "ResourceType": "ec2-public-ipv4-pool",
            "ResourceOwner": "123456789012"
        }
    ]
}
```

Step 12: Advertise the CIDR

The steps in this section must be done by the IPAM account. Once you associate the Elastic IP address (EIP) with an instance or Elastic Load Balancer, you can then start advertising the CIDR you brought to AWS that is in pool that has `--aws-service ec2` defined. In this tutorial, that's your Regional pool. By default the CIDR is not advertised, which means it's not publicly accessible over the internet. When you run the command in this section, the value for `--region` must match the `--locale` option you entered when you created the pool that will be used for the BYOIP CIDR.

This step must be done by the IPAM account.

Start advertising the CIDR using the AWS CLI

- Run the following command to advertise the CIDR.

```
aws ec2 advertise-byoip-cidr --region us-west-2 --cidr 130.137.245.0/24 --profile ipam-
account
```

  In the output, you'll see the CIDR is advertised.

```
{
    "ByoipCidr": {
        "Cidr": "130.137.245.0/24",
        "State": "advertised"
    }
}
```

Step 13: Cleanup

Follow the steps in this section to clean up the resources you've provisioned and created in this tutorial. When you run the commands in this section, the value for `--region` must match the `--locale` option you entered when you created the pool that will be used for the BYOIP CIDR.

Clean up using the AWS CLI

1. View the EIP allocation managed in IPAM.

   This step must be done by the IPAM account.

   ```
   aws ec2 get-ipam-pool-allocations --region us-west-2 --ipam-pool-id ipam-
   pool-0d8f3646b61ca5987 --profile ipam-account
   ```

   The output shows the allocation in IPAM.

   ```
   {
       "IpamPoolAllocations": [
           {
               "Cidr": "130.137.245.0/24",
               "IpamPoolAllocationId": "ipam-pool-alloc-5dedc8e7937c4261b56dc3e3eb53dc45",
               "ResourceId": "ipv4pool-ec2-0019eed22a684e0b2",
               "ResourceType": "ec2-public-ipv4-pool",
               "ResourceOwner": "123456789012"
           }
       ]
   }
   ```

2. Stop advertising the IPv4 CIDR.

   This step must be done by the IPAM account.

   ```
   aws ec2 withdraw-byoip-cidr --region us-west-2 --cidr 130.137.245.0/24 --profile ipam-
   account
   ```

   In the output, you'll see the CIDR State has changed from **advertised** to **provisioned**.

   ```
   {
       "ByoipCidr": {
           "Cidr": "130.137.245.0/24",
           "State": "provisioned"
       }
   }
   ```

3. Release the Elastic IP address.

   This step must be done by the member account.

   ```
   aws ec2 release-address --region us-west-2 --allocation-id eipalloc-0db3405026756dbf6
    --profile member-account
   ```

   You will not see any output when you run this command.

4. View your BYOIP CIDRs.

   This step must be done by the member account.

```
aws ec2 describe-public-ipv4-pools --region us-west-2 --profile member-account
```

In the output, you'll see the IP addresses in your BYOIP CIDR.

```
{
    "PublicIpv4Pools": [
        {
            "PoolId": "ipv4pool-ec2-0019eed22a684e0b2",
            "Description": "",
            "PoolAddressRanges": [
                {
                    "FirstAddress": "130.137.245.0",
                    "LastAddress": "130.137.245.255",
                    "AddressCount": 256,
                    "AvailableAddressCount": 256
                }
            ],
            "TotalAddressCount": 256,
            "TotalAvailableAddressCount": 256,
            "NetworkBorderGroup": "us-east-1",
            "Tags": []
        }
    ]
}
```

5. Release the last IP address in the CIDR from the public IPv4 pool. Enter the IP address with a netmask of /32. You must rerun this command for each IP address in the CIDR range. If your CIDR is a `/24`, you will have to run this command to deprovision each of the 256 IP addresses in the `/24` CIDR. When you run the command in this section, the value for `--region` must match the Region of your IPAM.

   This step must be done by the member account.

```
aws ec2 deprovision-public-ipv4-pool-cidr --region us-east-1 --pool-id ipv4pool-
ec2-0019eed22a684e0b2 --cidr 130.137.245.255/32 --profile member-account
```

   In the output, you'll see the deprovisioned CIDR.

```
{
    "PoolId": "ipv4pool-ec2-0019eed22a684e0b2",

    "DeprovisionedAddresses": [

        "130.137.245.255"

    ]

}
```

6. View your BYOIP CIDRs again and ensure there are no more provisioned addresses. When you run the command in this section, the value for `--region` must match the Region of your IPAM.

   This step must be done by the member account.

```
aws ec2 describe-public-ipv4-pools --region us-east-1 --profile member-account
```

   In the output, you'll see the IP addresses count in your public IPv4 pool.

```
{
    "PublicIpv4Pools": [
        {
            "PoolId": "ipv4pool-ec2-0019eed22a684e0b2",
            "Description": "",
            "PoolAddressRanges": [],
            "TotalAddressCount": 0,
            "TotalAvailableAddressCount": 0,
            "NetworkBorderGroup": "us-east-1",
            "Tags": []
        }
    ]
}
```

7. View the EIP allocation is no longer managed in IPAM. It can take some time for IPAM to discover that the Elastic IP address has been removed. You cannot continue to clean up and deprovision the IPAM pool CIDR until you see that the allocation has been removed from IPAM. When you run the command in this section, the value for `--region` must match the --locale option you entered when you created the pool that will be used for the BYOIP CIDR.

   This step must be done by the IPAM account.

```
aws ec2 get-ipam-pool-allocations --region us-west-2 --ipam-pool-id ipam-
pool-0d8f3646b61ca5987 --profile ipam-account
```

   The output shows the allocation in IPAM.

```
{
    "IpamPoolAllocations": []
}
```

8. Deprovision the Regional pool CIDR. When you run the commands in this step, the value for `--region` must match the Region of your IPAM.

   This step must be done by the IPAM account.

```
aws ec2 deprovision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-0d8f3646b61ca5987 --cidr 130.137.245.0/24 --profile ipam-account
```

   In the output, you'll see the CIDR pending deprovision.

```
{
    "IpamPoolCidr": {

        "Cidr": "130.137.245.0/24",

        "State": "pending-deprovision"

    }

}
```

   Deprovisioning takes time to complete. Check the status of deprovisioning.

```
aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-
pool-0d8f3646b61ca5987 --profile ipam-account
```

Wait until you see **deprovisioned** before you continue to the next step.

```
{
    "IpamPoolCidr": {

        "Cidr": "130.137.245.0/24",

        "State": "deprovisioned"

    }

}
```

9. Delete the RAM share and disable RAM integration with AWS Organizations. Complete the steps in Deleting a resource share in AWS RAM and Disabling resource sharing with AWS Organizations in the *AWS RAM User Guide*, in that order, to delete the RAM share and disable RAM integration with AWS Organizations.

   This step must be done by the IPAM account and management account respectively. If you are using the AWS CLI to delete the RAM share and disable RAM integration, use the `--profile` **`ipam-account`** and `--profile` **`management-account`** options.

10. Delete the Regional pool. When you run the command in this step, the value for `--region` must match the Region of your IPAM.

    This step must be done by the IPAM account.

```
aws ec2 delete-ipam-pool --region us-east-1 --ipam-pool-id ipam-pool-0d8f3646b61ca5987
 --profile ipam-account
```

    In the output, you can see the delete state.

```
{
   "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0d8f3646b61ca5987",
        "SourceIpamPoolId": "ipam-pool-0a03d430ca3f5c035",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0d8f3646b61ca5987",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
        "IpamScopeType": "public",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "Locale": "us-east-1",
        "PoolDepth": 2,
        "State": "delete-in-progress",
        "Description": "reg-ipv4-pool",
        "AutoImport": false,
        "Advertisable": true,
        "AddressFamily": "ipv4"
    }
}
```

11. Deprovision the top-level pool CIDR. When you run the commands in this step, the value for `--region` must match the Region of your IPAM.

    This step must be done by the IPAM account.

```
aws ec2 deprovision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-0a03d430ca3f5c035 --cidr 130.137.245.0/24 --profile ipam-account
```

In the output, you'll see the CIDR pending deprovision.

```
{
    "IpamPoolCidr": {

        "Cidr": "130.137.245.0/24",

        "State": "pending-deprovision"

    }

}
```

Deprovisioning takes time to complete. Run the following command to check the status of deprovisioning.

```
aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-
pool-0a03d430ca3f5c035 --profile ipam-account
```

Wait until you see **deprovisioned** before you continue to the next step.

```
{
    "IpamPoolCidr": {

        "Cidr": "130.137.245.0/24",

        "State": "deprovisioned"

    }

}
```

12. Delete the top-level pool. When you run the command in this step, the value for `--region` must match the Region of your IPAM.

    This step must be done by the IPAM account.

```
aws ec2 delete-ipam-pool --region us-east-1 --ipam-pool-id ipam-pool-0a03d430ca3f5c035
 --profile ipam-account
```

In the output, you can see the delete state.

```
{
  "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0a03d430ca3f5c035",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0a03d430ca3f5c035",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
        "IpamScopeType": "public",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "Locale": "us-east-1",
        "PoolDepth": 2,
        "State": "delete-in-progress",
        "Description": "top-level-pool",
        "AutoImport": false,
        "Advertisable": true,
```

```
        "AddressFamily": "ipv4"
    }
}
```

13. Delete the IPAM. When you run the command in this step, the value for `--region` must match the Region of your IPAM.

    This step must be done by the IPAM account.

```
aws ec2 delete-ipam --region us-east-1 --ipam-id ipam-090e48e75758de279 --profile ipam-account
```

    In the output, you'll see the IPAM response. This means that the IPAM was deleted.

```
{
    "Ipam": {
        "OwnerId": "123456789012",
        "IpamId": "ipam-090e48e75758de279",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "PublicDefaultScopeId": "ipam-scope-0087d83896280b594",
        "PrivateDefaultScopeId": "ipam-scope-08b70b04fbd524f8d",
        "ScopeCount": 2,

        "OperatingRegions": [
            {
                "RegionName": "us-east-1"
            },
            {
                "RegionName": "us-west-2"
            }
        ],
    }
}
```

Bring your own IPv6 CIDR to IPAM using only the AWS CLI

Follow these steps to bring an IPv6 CIDR to IPAM and allocate a VPC using only the AWS CLI.

Important

- This tutorial assumes you have already completed the steps in the following sections:
  - Integrate IPAM with AWS Organizations (p. 5).
  - Create an IPAM (p. 7).
- Each step of this tutorial must be done by one of three AWS Organizations accounts:
  - The management account.
  - The member account configured to be your IPAM administrator in Integrate IPAM with AWS Organizations (p. 5). In this tutorial, this account will be called the IPAM account.
  - The member account in your organization which will allocate CIDRs from an IPAM pool. In this tutorial, this account will be called the member account.

Contents

Step 1: Create AWS CLI named profiles

To complete this tutorial as a single AWS user, you can use AWS CLI named profiles to switch from one AWS account to another. Named profiles are collections of IAM access key IDs and secret access keys that you store locally and then refer to using the `--profile` option when you use the AWS CLI. For more information about how to create or retrieve IAM access keys for AWS accounts, see Managing access keys for IAM users in the *AWS Identity and Access Management User Guide*.

Complete the steps in Creating named profiles in the *AWS Command Line Interface User Guide* to create one named profiles for each of the three AWS accounts you will use in this tutorial:

- A profile called `management-account` for the AWS Organizations management account.
- A profile called `ipam-account` for the AWS Organizations member account that is configured to be your IPAM administrator.
- A profile called `member-account` for the AWS Organizations member account in your organization which will allocate CIDRs from an IPAM pool.

Once you have created the named profiles, return to this page and go to the next step. You will notice throughout the rest of this tutorial that the sample AWS CLI commands use the `--profile` option with one of the named profiles to indicate which account must run the command.

Step 2: Create an IPAM

This step is optional. If you already have an IPAM created with operating Regions of `us-east-1` and `us-west-2` created, you can skip this step. Create an IPAM and specify an operating region of `us-east-1` and `us-west-2` . You must select an operating region so that you can use the locale option when you create your IPAM pool. The IPAM integration with BYOIP requires that the locale is set on whichever pool will be used for the BYOIP CIDR.

This step must be done by the IPAM account.

Run the following command:

```
aws ec2 create-ipam --description my-ipam --region us-east-1 --operating-
regions RegionName=us-west-2 --profile ipam-account
```

In the output, you'll see the IPAM you've created. Note the value for `PublicDefaultScopeId`. You will need your public scope ID in the next step.

```
{
 "Ipam": {
        "OwnerId": "123456789012",
        "IpamId": "ipam-090e48e75758de279",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "PublicDefaultScopeId": "ipam-scope-0087d83896280b594",
        "PrivateDefaultScopeId": "ipam-scope-08b70b04fbd524f8d",
```

```
        "ScopeCount": 2,
        "Description": "my-ipam",
        "OperatingRegions": [
            {
                "RegionName": "us-east-1"
            },
            {
                "RegionName": "us-west-2"
            }
        ],
        "Tags": []
    }
}
```

Step 3: Create an IPAM pool

Since you are going to create a top-level IPAM pool with a Regional pool within it, and we're going to allocate space to a resource (a VPC) from the Regional pool, you will set the locale on the Regional pool and not the top-level pool. You'll add the locale to the Regional pool when you create the Regional pool in a later step. The IPAM integration with BYOIP requires that the locale is set on whichever pool will be used for the BYOIP CIDR.

This step must be done by the IPAM account.

Choose if you want this IPAM pool CIDR to be advertisable by AWS over the public internet (`--publicly-advertisable` or `--no-publicly-advertisable`).

Note
Note that the scope ID must be the ID for the public scope and the address family must be `ipv6`.

To create an IPv6 address pool for all of your AWS resources using the AWS CLI

1. Run the following command to create an IPAM pool. Use the ID of the public scope of the IPAM that you created in the previous step.

```
aws ec2 create-ipam-pool --region us-east-1 --ipam-scope-id ipam-
scope-0087d83896280b594 --description "top-level-IPv6-pool" --address-family ipv6 --
publicly-advertisable --profile ipam-account
```

   In the output, you'll see `create-in-progress`, which indicates that pool creation is in progress.

```
{
    "IpamPool": {

        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-07f2466c7158b50c4",

        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-07f2466c7158b50c4",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
        "IpamScopeType": "public",

        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",

        "Locale": "None",

        "PoolDepth": 1,

        "State": "create-in-progress",
```

```
        "Description": "top-level-Ipv6-pool",

        "AutoImport": false,

        "Advertisable": true,

        "AddressFamily": "ipv6",

        "Tags": []

    }

}
```

2. Run the following command until you see a state of `create-complete` in the output.

```
aws ec2 describe-ipam-pools --region us-east-1 --profile ipam-account
```

The following example output shows the state of the pool.

```
{
    "IpamPool": {

        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-07f2466c7158b50c4",

        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-07f2466c7158b50c4",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
        "IpamScopeType": "public",

        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",

        "Locale": "None",

        "PoolDepth": 1,

        "State": "create-complete",

        "Description": "top-level-Ipv6-pool",

        "AutoImport": false,

        "Advertisable": true,

        "AddressFamily": "ipv6",

        "Tags": []

    }

}
```

Step 4: Provision a CIDR to the top-level pool

Provision a CIDR block to the top-level pool. Note that when provisioning an IPv6 CIDR to a pool within the top-level pool, the minimum IPv6 CIDR you can provision for an advertisable IPAM pool is `/48`; more specific CIDRs (such as `/49`) are not permitted. The minimum CIDR you can bring in for an non-advertisable IPAM pool is `/56`; more specific CIDRs (such as `/57`) are not permitted. You must include the CIDR and the BYOIP message and certificate signature in the request so we can verify that you own the

public space. For a list of BYOIP prerequisites including how to get this BYOIP message and certificate signature, see Bring your own public IPv4 CIDR to IPAM using only the AWS CLI (p. 64).

You only need to add `--cidr-authorization-context` when you provision the BYOIP CIDR to the top-level pool. For the Regional pool within the top-level pool, you can omit the `--cidr-authorization-context` option.

This step must be done by the IPAM account.

To provision a CIDR block to the pool using the AWS CLI

1. Run the following command to provision the CIDR.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-07f2466c7158b50c4 --cidr 2605:9cc0:409::/48 --cidr-authorization-
context Message="1|aws|470889052444|2605:9cc0:409::/48|20250101|
SHA256|RSAPSS",Signature="FU26~vRG~NUGXa~akxd6dvdcCfvL88g8d~YAuai-
CR7HqMwzcgdS9RlpBGtfIdsRGyr77LmWyWqU9Xp1g2R1kSkfD00NiLKLcv9F63k6wdEkyFxNp7RAJDvF1mBwxmSgH~Crt-
Vp6LON3yOOXMp4JENB9uM7sMlu6oeoutGyyhXFeYPzlGSRdcdfKNKaimvPCqVsxGN5AwSilKQ8byNqoa~G3dvs8ueSaDcT~tW4C
wispI~r69fq515UR19TA~fmmxBDh1huQ8DkM1rqcwveWow__" --profile ipam-account
```

   In the output, you'll see the CIDR pending provision.

```
{
    "IpamPoolCidr": {

        "Cidr": "2605:9cc0:409::/48",

        "State": "pending-provision"

    }

}
```

2. Ensure that this CIDR has been provisioned before you continue. Note that it can take up to one week for the BYOIP CIDR to be provisioned. Run the following command until you see a state of `provisioned` in the output.

```
aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-
pool-07f2466c7158b50c4 --profile ipam-account
```

   The following example output shows the state.

```
{
    "IpamPoolCidrs": [
        {
            "Cidr": "2605:9cc0:409::/48",
            "State": "provisioned"
        }
    ]
}
```

Step 5: Create a Regional pool within the top-level pool

Create a Regional pool within the top-level pool. `--locale` is required on the pool and it must be one of the operating Regions you configured when you created the IPAM.

This step must be done by the IPAM account.

To create a Regional pool using the AWS CLI

1. Run the following command to create the pool.

```
aws ec2 create-ipam-pool --description "Regional-IPv6-pool" --region us-east-1 --ipam-
scope-id ipam-scope-0087d83896280b594 --source-ipam-pool-id ipam-pool-07f2466c7158b50c4
 --locale us-west-2 --address-family ipv6 --aws-service ec2 --profile ipam-account
```

   In the output, you'll see IPAM creating the pool.

```
{
    "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0053b7d2b4fc3f730",
        "SourceIpamPoolId": "ipam-pool-07f2466c7158b50c4",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0053b7d2b4fc3f730",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
        "IpamScopeType": "public",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "Locale": "us-west-2",
        "PoolDepth": 2,
        "State": "create-in-progress",
        "Description": "reg-ipv6-pool",
        "AutoImport": false,
        "Advertisable": true,
        "AddressFamily": "ipv6",
        "Tags": [],
        "ServiceType": "ec2"
    }
}
```

2. Run the following command until you see a state of `create-complete` in the output.

```
aws ec2 describe-ipam-pools --region us-east-1 --profile ipam-account
```

   In the output, you see the pools that you have in your IPAM. In this tutorial, we created a top-level and a Regional pool, so you'll see them both.

Step 6: Provision a CIDR to the Regional pool

Provision a CIDR block to the Regional pool. Note that when provisioning the CIDR to a pool within the top-level pool, the minimum IPv6 CIDR you can provision for an advertisable IPAM pool is `/48`; more specific CIDRs (such as `/49`) are not permitted. The minimum CIDR you can bring in for an non-advertisable IPAM pool is `/56`; more specific CIDRs (such as `/57`) are not permitted.

This step must be done by the IPAM account.

To assign a CIDR block to the Regional pool using the AWS CLI

1. Run the following command to provision the CIDR.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-0053b7d2b4fc3f730 --cidr 2605:9cc0:409::/48 --profile ipam-account
```

   In the output, you'll see the CIDR pending provision.

```
{
    "IpamPoolCidr": {
        "Cidr": "2605:9cc0:409::/48",
        "State": "pending-provision"
    }
}
```

2. Run the following command until you see the state of `provisioned` in the output.

```
aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-
pool-0053b7d2b4fc3f730 --profile ipam-account
```

The following example output shows the correct state.

```
{
    "IpamPoolCidrs": [
        {
            "Cidr": "2605:9cc0:409::/48",
            "State": "provisioned"
        }
    ]
}
```

Step 7: Enable resource sharing with AWS Organizations using AWS RAM

You will use AWS RAM to share your Regional pool with the AWS Organizations member account who would like to allocate a CIDR from the Regional pool for a VPC. Before you can do that, you must enable RAM integration with AWS Organizations.

Complete the steps in Enable resource sharing within AWS Organizations in the *AWS RAM User Guide* using the management account. If you are using the AWS CLI to enable resource sharing, use the `--profile` **`management-account`** option. Once resource sharing is enabled in RAM, go to the next step in this tutorial.

Step 8: Share your Regional pool with an AWS Organizations member account using AWS RAM

Complete the process in Share an IPAM pool using AWS RAM (p. 18) and share the Regional pool with the AWS Organizations member account.

This step must be done by the IPAM account. If you are using the AWS CLI to share the pool, use the `--profile` **`ipam-account`** option.

Important
When you create the resource share, ensure the following:

- The principal is the account ID of the member account who will be allocating a CIDR from the pool for the Elastic IP address.
- You assign the *AWSRAMPermissionIpamPoolByoipCidrImport* permission to the pool.

Step 9: Create a VPC using the IPv6 CIDR

Create a VPC using the IPAM pool ID. You must associate an IPv4 CIDR block to the VPC as well using the `--cidr-block` option or the request will fail. When you run the command in this section, the value for `--region` must match the `--locale` option you entered when you created the pool that will be used for the BYOIP CIDR.

This step must be done by the member account.

To create a VPC with the IPv6 CIDR using the AWS CLI

1. Run the following command to provision the CIDR.

```
aws ec2 create-vpc --region us-west-2 --ipv6-ipam-pool-id ipam-pool-0053b7d2b4fc3f730
 --cidr-block 10.0.0.0/16 --ipv6-netmask-length 56 --profile member-account
```

   In the output, you'll see the VPC being created.

```
{
    "Vpc": {
        "CidrBlock": "10.0.0.0/16",
        "DhcpOptionsId": "dopt-2afccf50",
        "State": "pending",
        "VpcId": "vpc-00b5573ffc3b31a29",
        "OwnerId": "123456789012",
        "InstanceTenancy": "default",
        "Ipv6CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-01b5703d6cc695b5b",
                "Ipv6CidrBlock": "2605:9cc0:409::/56",
                "Ipv6CidrBlockState": {
                    "State": "associating"
                },
                "NetworkBorderGroup": "us-east-1",
                "Ipv6Pool": "ipam-pool-0053b7d2b4fc3f730"
            }
        ],
        "CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-09cccb07d4e9a0e0e",
                "CidrBlock": "10.0.0.0/16",
                "CidrBlockState": {
                    "State": "associated"
                }
            }
        ],
        "IsDefault": false
    }
}
```

2. View the VPC allocation in IPAM.

```
aws ec2 get-ipam-pool-allocations --region us-west-2 --ipam-pool-id ipam-
pool-0053b7d2b4fc3f730 --profile ipam-account
```

   In the output, you'll see allocation in IPAM.

```
{
    "IpamPoolAllocations": [
        {
            "Cidr": "2605:9cc0:409::/56",
            "IpamPoolAllocationId": "ipam-pool-alloc-5f8db726fb9e4ff0a33836e649283a52",
            "ResourceId": "vpc-00b5573ffc3b31a29",
            "ResourceType": "vpc",
            "ResourceOwner": "123456789012"
        }
    ]
}
```

Step 10: Advertise the CIDR

Once you create the VPC with CIDR allocated in IPAM, you can then start advertising the CIDR you brought to AWS that is in pool that has `--aws-service ec2` defined. In this tutorial, that's your Regional pool. By default the CIDR is not advertised, which means it's not publicly accessible over the internet. When you run the command in this section, the value for `--region` must match the `--locale` option you entered when you created the Regional pool that will be used for the BYOIP CIDR.

This step must be done by the IPAM account.

Start advertising the CIDR using the AWS CLI

- Run the following command to advertise the CIDR.

```
aws ec2 advertise-byoip-cidr --region us-west-2 --cidr 2605:9cc0:409::/48 --
profile ipam-account
```

In the output, you'll see the CIDR is advertised.

```
{
    "ByoipCidr": {
        "Cidr": "2605:9cc0:409::/48",
        "State": "advertised"
    }
}
```

Step 11: Cleanup

Follow the steps in this section to clean up the resources you've provisioned and created in this tutorial. When you run the commands in this section, the value for `--region` must match the `--locale` option you entered when you created the Regional pool that will be used for the BYOIP CIDR.

Clean up using the AWS CLI

1. Run the following command to view the VPC allocation managed in IPAM.

   This step must be done by the IPAM account.

   ```
   aws ec2 get-ipam-pool-allocations --region us-west-2 --ipam-pool-id ipam-
   pool-0053b7d2b4fc3f730 --profile ipam-account
   ```

   The output shows the allocation in IPAM.

   ```
   {
       "IpamPoolAllocations": [
           {
               "Cidr": "2605:9cc0:409::/56",
               "IpamPoolAllocationId": "ipam-pool-alloc-5f8db726fb9e4ff0a33836e649283a52",
               "ResourceId": "vpc-00b5573ffc3b31a29",
               "ResourceType": "vpc",
               "ResourceOwner": "123456789012"
           }
       ]
   }
   ```

2. Run the following command to stop advertising the CIDR. When you run the command in this step, the value for `--region` must match the `--locale` option you entered when you created the Regional pool that will be used for the BYOIP CIDR.

This step must be done by the IPAM account.

```
aws ec2 withdraw-byoip-cidr --region us-west-2 --cidr 2605:9cc0:409::/48 --
profile ipam-account
```

In the output, you'll see the CIDR State has changed from **advertised** to **provisioned**.

```
{
    "ByoipCidr": {
        "Cidr": "2605:9cc0:409::/48",
        "State": "provisioned"
    }
}
```

3. Run the following command to delete the VPC. When you run the command in this section, the value for `--region` must match the `--locale` option you entered when you created the Regional pool that will be used for the BYOIP CIDR.

   This step must be done by the member account.

```
aws ec2 delete-vpc --region us-west-2 --vpc-id vpc-00b5573ffc3b31a29 --profile member-
account
```

   You will not see any output when you run this command.

4. Run the following command to view the VPC allocation in IPAM. It can take some time for IPAM to discover that the VPC has been deleted and remove this allocation. When you run the commands in this section, the value for `--region` must match the `--locale` option you entered when you created the Regional pool that will be used for the BYOIP CIDR.

   This step must be done by the IPAM account.

```
aws ec2 get-ipam-pool-allocations --region us-west-2 --ipam-pool-id ipam-
pool-0053b7d2b4fc3f730 --profile ipam-account
```

   The output shows the allocation in IPAM.

```
{
   "IpamPoolAllocations": [

        {

            "Cidr": "2605:9cc0:409::/56",

            "IpamPoolAllocationId": "ipam-pool-alloc-5f8db726fb9e4ff0a33836e649283a52",

            "ResourceId": "vpc-00b5573ffc3b31a29",

            "ResourceType": "vpc",

            "ResourceOwner": "123456789012"

        }

    ]

}
```

Rerun the command and look for the allocation to be removed. You cannot continue to clean up and deprovision the IPAM pool CIDR until you see that the allocation has been removed from IPAM.

```
aws ec2 get-ipam-pool-allocations --region us-west-2 --ipam-pool-id ipam-
pool-0053b7d2b4fc3f730 --profile ipam-account
```

The output shows the allocation removed from IPAM.

```
{
    "IpamPoolAllocations": []
}
```

5. Delete the RAM share and disable RAM integration with AWS Organizations. Complete the steps in Deleting a resource share in AWS RAM and Disabling resource sharing with AWS Organizations in the *AWS RAM User Guide*, in that order, to delete the RAM share and disable RAM integration with AWS Organizations.

   This step must be done by the IPAM account and management account respectively. If you are using the AWS CLI to delete the RAM share and disable RAM integration, use the `--profile ipam-account` and `--profile management-account` options.

6. Run the following command to deprovision the Regional pool CIDR.

   This step must be done by the IPAM account.

   ```
   aws ec2 deprovision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
   pool-0053b7d2b4fc3f730 --cidr 2605:9cc0:409::/48 --profile ipam-account
   ```

   In the output, you'll see the CIDR pending deprovision.

   ```
   {
       "IpamPoolCidr": {
           "Cidr": "2605:9cc0:409::/48",
           "State": "pending-deprovision"
       }
   }
   ```

   Deprovisioning takes time to complete. Continue to run the command until you see the CIDR state **deprovisioned**.

   ```
   aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-
   pool-0053b7d2b4fc3f730 --cidr 2605:9cc0:409::/48 --profile ipam-account
   ```

   In the output, you'll see the CIDR pending deprovision.

   ```
   {
       "IpamPoolCidr": {
           "Cidr": "2605:9cc0:409::/48",
           "State": "deprovisioned"
       }
   }
   ```

7. Run the following command to delete the Regional pool.

   This step must be done by the IPAM account.

```
aws ec2 delete-ipam-pool --region us-east-1 --ipam-pool-id ipam-pool-0053b7d2b4fc3f730
 --profile ipam-account
```

In the output, you can see the delete state.

```
{
    "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0053b7d2b4fc3f730",
        "SourceIpamPoolId": "ipam-pool-07f2466c7158b50c4",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0053b7d2b4fc3f730",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
        "IpamScopeType": "public",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "Locale": "us-east-1",
        "PoolDepth": 2,
        "State": "delete-in-progress",
        "Description": "reg-ipv6-pool",
        "AutoImport": false,
        "Advertisable": true,
        "AddressFamily": "ipv6"
    }
}
```

8. Run the following command to deprovision the top-level pool CIDR.

   This step must be done by the IPAM account.

```
aws ec2 deprovision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-07f2466c7158b50c4 --cidr 2605:9cc0:409::/48 --profile ipam-account
```

   In the output, you'll see the CIDR pending deprovision.

```
{
    "IpamPoolCidr": {
        "Cidr": "2605:9cc0:409::/48",
        "State": "pending-deprovision"
    }
}
```

   Deprovisioning takes time to complete. Run the following command to check the status of deprovisioning.

```
aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-
pool-07f2466c7158b50c4 --profile ipam-account
```

   Wait until you see **deprovisioned** before you continue to the next step.

```
{
    "IpamPoolCidr": {

        "Cidr": "2605:9cc0:409::/48",

        "State": "deprovisioned"
```

```
    }

}
```

9. Run the following command to delete the top-level pool.

   This step must be done by the IPAM account.

```
aws ec2 delete-ipam-pool --region us-east-1 --ipam-pool-id ipam-pool-07f2466c7158b50c4
 --profile ipam-account
```

   In the output, you can see the delete state.

```
{
    "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0053b7d2b4fc3f730",
        "SourceIpamPoolId": "ipam-pool-07f2466c7158b50c4",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0053b7d2b4fc3f730",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
        "IpamScopeType": "public",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "Locale": "us-east-1",
        "PoolDepth": 2,
        "State": "delete-in-progress",
        "Description": "reg-ipv6-pool",
        "AutoImport": false,
        "Advertisable": true,
        "AddressFamily": "ipv6"
    }
}
```

10. Run the following command to delete the IPAM.

    This step must be done by the IPAM account.

```
aws ec2 delete-ipam --region us-east-1 --ipam-id ipam-090e48e75758de279 --profile ipam-
account
```

    In the output, you'll see the IPAM response. This means that the IPAM was deleted.

```
{
    "Ipam": {
        "OwnerId": "123456789012",
        "IpamId": "ipam-090e48e75758de279",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "PublicDefaultScopeId": "ipam-scope-0087d83896280b594",
        "PrivateDefaultScopeId": "ipam-scope-08b70b04fbd524f8d",
        "ScopeCount": 2,
        "OperatingRegions": [
            {
                "RegionName": "us-east-1"
            },
            {
                "RegionName": "us-west-2"
            }
        ]
    }
}
```

Tutorial: Transfer existing BYOIP IPv4 CIDRs to IPAM

Follow these steps to transfer an existing IPv4 CIDR to IPAM. If you already have an IPv4 BYOIP CIDR with AWS, you can move the CIDR to IPAM from a public IPv4 pool. You cannot move an IPv6 CIDR to IPAM. If you are bringing a new IP address to AWS for the first time, complete the steps in Tutorial: BYOIP address CIDRs to IPAM (p. 49).

Important

- This tutorial assumes you have already completed the steps in Create an IPAM (p. 7).
- Each step of this tutorial must be done by one of two AWS accounts:
  - The account for the IPAM administrator. In this tutorial, this account will be called the IPAM account.
  - The account in your organization which which owns the BYOIP CIDR. In this tutorial, this account will be called the BYOIP CIDR owner account.

Note
The IPAM account must share the pool with the BYOIP CIDR owner via AWS RAM and include the `AWSRAMPermissionIpamPoolByoipCidrImport` policy on the shared resource. For more information, see Share an IPAM pool using AWS RAM (p. 18). To transfer the BYOIP CIDR to IPAM, the BYOIP CIDR owner must have these permissions in their IAM policy:

- `ec2:MoveByoipCidrToIpam`
- `ec2:ImportByoipCidrToIpam`

Contents

Step 1: Create AWS CLI named profiles

To complete this tutorial as a single AWS user, you can use AWS CLI named profiles to switch from one AWS account to another. Named profiles are collections of IAM access key IDs and secret access keys that you store locally and then refer to using the `--profile` option when you use the AWS CLI. For more information about how to create or retrieve IAM access keys for AWS accounts, see Managing access keys for IAM users in the *AWS Identity and Access Management User Guide*.

Complete the steps in Creating named profiles in the *AWS Command Line Interface User Guide* to create one named profiles for each of the AWS accounts you will use in this tutorial:

- A profile called `ipam-account` for the AWS account that is the IPAM administrator.
- A profile called `byoip-owner-account` for the AWS account in your organization which owns the BYOIP CIDR.

Once you have created the named profiles, return to this page and go to the next step. You will notice throughout the rest of this tutorial that the sample AWS CLI commands use the `--profile` option with one of the named profiles to indicate which account must run the command.

Step 2: Get your IPAM's public scope ID

Follow the steps in this section to get your IPAM's public scope ID. This step should be performed by the IPAM account.

Run the following command to get your public scope ID.

```
aws ec2 describe-ipams --region us-east-1 --profile ipam-account
```

In the output, you'll see your public scope ID. Note the values for `PublicDefaultScopeId`. You will need it in the next step.

```
{
 "Ipams": [
        {
            "OwnerId": "123456789012",
            "IpamId": "ipam-090e48e75758de279",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
            "PublicDefaultScopeId": "ipam-scope-0087d83896280b594",
            "PrivateDefaultScopeId": "ipam-scope-08b70b04fbd524f8d",
            "ScopeCount": 2,
            "Description": "my-ipam",
            "OperatingRegions": [
                {
                    "RegionName": "us-east-1"
                },
                {
                    "RegionName": "us-west-2"
                }
            ],
            "Tags": []
        }
    ]
}
```

Step 3: Create an IPAM pool

Follow the steps in this section to create an IPAM pool. This step should be performed by the IPAM account. The IPAM pool you create must be a top-level pool with the `--locale` option matching the BYOIP CIDR AWS Region and the pool must be created with the `--aws-service ec2` option. You can only transfer a BYOIP to a top-level IPAM pool.

To create an IPv4 address pool for the transferred BYOIP CIDR using the AWS CLI

1. Run the following command to create an IPAM pool. Use the ID of the public scope of the IPAM that you retrieved in the previous step.

   ```
   aws ec2 create-ipam-pool --region us-east-1 --ipam-scope-id ipam-
   scope-0087d83896280b594 --description "top-level-pool" --locale us-west-2 --aws-service
    ec2 --address-family ipv4 --profile ipam-account
   ```

   In the output, you'll see `create-in-progress`, which indicates that pool creation is in progress.

   ```
   {
   ```

```
    "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0a03d430ca3f5c035",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0a03d430ca3f5c035",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
        "IpamScopeType": "public",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "Locale": "us-west-2",
        "PoolDepth": 1,
        "State": "create-in-progress",
        "Description": "top-level-pool",
        "AutoImport": false,
        "AddressFamily": "ipv4",
        "Tags": [],
        "AwsService": "ec2"
    }
}
```

2. Run the following command until you see a state of `create-complete` in the output.

```
aws ec2 describe-ipam-pools --region us-east-1 --profile ipam-account
```

The following example output shows the state of the pool. You will need the **OwnerId** in the next step.

```
{
    "IpamPools": [
        {
            "OwnerId": "123456789012",
            "IpamPoolId": "ipam-pool-0a03d430ca3f5c035",
            "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0a03d430ca3f5c035",
            "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
            "IpamScopeType": "public",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
            "Locale": "us-west-2",
            "PoolDepth": 1,
            "State": "create-complete",
            "Description": "top-level-pool",
            "AutoImport": false,
            "AddressFamily": "ipv4",
            "Tags": [],
            "AwsService": "ec2"
        }
    ]
}
```

Step 4: Transfer an existing BYOIP IPV4 CIDR to IPAM

Follow the steps in this section to transfer an existing BYOIP IPV4 CIDR to IPAM. This step should be performed by the BYOIP CIDR owner account.

To transfer a BYOIP CIDR to the IPAM pool using the AWS CLI

1. Run the following command to transfer the CIDR. Ensure that the `--region` value is the AWS Region of the BYOIP CIDR.

```
aws ec2 move-byoip-cidr-to-ipam --region us-west-2 --ipam-pool-id ipam-
pool-0a03d430ca3f5c035 --ipam-pool-owner 123456789012 --cidr 130.137.249.0/24 --
profile byoip-owner-account
```

In the output, you'll see the CIDR pending provision.

```
{
    "ByoipCidr": {
        "Cidr": "130.137.249.0/24",
        "State": "pending-transfer"

    }
}
```

2. Ensure that the CIDR has been transferred. Run the following command until you see a state of `complete-transfer` in the output.

```
aws ec2 move-byoip-cidr-to-ipam --region us-west-2 --ipam-pool-id ipam-
pool-0a03d430ca3f5c035 --ipam-pool-owner 123456789012 --cidr 130.137.249.0/24 --
profile byoip-owner-account
```

The following example output shows the state.

```
{
    "ByoipCidr": {
        "Cidr": "130.137.249.0/24",
        "State": "complete-transfer"

    }
}
```

Step 5: View the CIDR in IPAM

Follow the steps in this section to view the CIDR in IPAM. This step should be performed by the IPAM account.

To view the transferred BYOIP CIDR in IPAM pool using the AWS CLI

- Run the following command to view the allocation managed in IPAM. Ensure that the `--region` value is the AWS Region of the BYOIP CIDR.

```
aws ec2 get-ipam-pool-allocations --region us-west-2 --ipam-pool-id ipam-
pool-0d8f3646b61ca5987 --profile ipam-account
```

The output shows the allocation in IPAM.

```
{
    "IpamPoolAllocations": [
        {
            "Cidr": "130.137.249.0/24",
            "IpamPoolAllocationId": "ipam-pool-alloc-5dedc8e7937c4261b56dc3e3eb53dc46",
            "ResourceId": "ipv4pool-ec2-0019eed22a684e0b3",
            "ResourceType": "ec2-public-ipv4-pool",
            "ResourceOwner": "470889052924"
        }
    ]
```

```
}
```

Step 6: Cleanup

Follow the steps in this section to remove the resources you created in this tutorial. This step should be performed by the IPAM account.

To cleanup the resources created in this tutorial using the AWS CLI

1. Run the following command to get the allocation ID for the BYOIP CIDR. Ensure that the `--region` value matches the AWS Region of the BYOIP CIDR.

```
aws ec2 get-ipam-pool-allocations --region us-west-2 --ipam-pool-id ipam-
pool-0d8f3646b61ca5987 --profile ipam-account
```

   The output shows the allocation in IPAM.

```
{
    "IpamPoolAllocations": [
        {
            "Cidr": "130.137.249.0/24",
            "IpamPoolAllocationId": "ipam-pool-alloc-5dedc8e7937c4261b56dc3e3eb53dc46",
            "ResourceId": "ipv4pool-ec2-0019eed22a684e0b3",
            "ResourceType": "ec2-public-ipv4-pool",
            "ResourceOwner": "470889052924"
        }
    ]
}
```

2. Run the following command to deallocate the BYOIP CIDR. It can take some time for IPAM to discover that the VPC has been deleted and remove this allocation. Ensure that the `--region` value is the AWS Region of the BYOIP CIDR.

```
aws ec2 release-ipam-pool-allocation --region us-west-2 --ipam-pool-id ipam-
pool-0a03d430ca3f5c035 --cidr 130.137.249.0/24 --ipam-pool-allocation-id ipam-pool-
alloc-5dedc8e7937c4261b56dc3e3eb53dc46 --profile ipam-account
```

   The output shows the allocation removed from IPAM.

```
{
    "IpamPoolAllocations": []
}
```

3. Run the following command to delete the top-level pool.

```
aws ec2 delete-ipam-pool --region us-east-1 --ipam-pool-id ipam-pool-0a03d430ca3f5c035
 --profile ipam-account
```

   In the output, you can see the delete state.

```
{
  "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0a03d430ca3f5c035",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0a03d430ca3f5c035",
```

```
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0087d83896280b594",
        "IpamScopeType": "public",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-090e48e75758de279",
        "Locale": "us-east-1",
        "PoolDepth": 2,
        "State": "delete-in-progress",
        "Description": "top-level-pool",
        "AutoImport": false,
        "Advertisable": true,
        "AddressFamily": "ipv4",
        "AwsService": "ec2"
    }
}
```

Identity and access management in IPAM

AWS uses security credentials to identify you and to grant you access to your AWS resources. You can use features of AWS Identity and Access Management (IAM) to allow other users, services, and applications to use your AWS resources fully or in a limited way, without sharing your security credentials.

This section describes the AWS service-linked roles that are created specifically for IPAM and the managed policies attached to the IPAM service-linked roles. For more information about AWS IAM roles and policies, see Roles terms and concepts in the *IAM User Guide*.

For more information on identity and access management for VPC, see Identity and access management for Amazon VPC in the *Amazon VPC User Guide*.

Contents

Service-linked roles for IPAM

Service-linked roles in AWS Identity and Access Management (IAM) enable AWS services to call other AWS services on your behalf. For more information about service-linked roles, see Using service-linked roles in the *IAM User Guide*.

There is currently only one service-linked role for IPAM: **AWSServiceRoleForIPAM**.

Permissions granted to the service-linked role

IPAM uses the **AWSServiceRoleForIPAM** service-linked role to call the actions in the attached **AWSIPAMServiceRolePolicy** managed policy. For more information on the allowed actions in that policy, see AWS managed policies for IPAM (p. 99).

Also attached to the service-linked role is an IAM trusted policy that allows the `ipam.amazonaws.com` service to assume the service-linked role.

Create the service-linked role

IPAM monitors the IP address usage in one or more accounts by assuming the service-linked role in an account, discovering the resources and their CIDRs, and integrating the resources with IPAM.

The service-linked role is created in one of two ways:

- **When you integrate with AWS Organizations**

  If you Integrate IPAM with AWS Organizations (p. 5) using the IPAM console or using the `enable-ipam-organization-admin-account` AWS CLI command, the **AWSServiceRoleForIPAM** service-linked role is automatically created in each of your AWS Organizations member accounts. As a result, the resources within all member accounts are discoverable by IPAM.

  Important
  For IPAM to create the service-linked role on your behalf:

  - The AWS Organizations management account that enables IPAM integration with AWS Organizations must have an IAM policy attached to it that permits the following actions:
    - `ec2:EnableIpamOrganizationAdminAccount`
    - `organizations:EnableAwsServiceAccess`
    - `organizations:RegisterDelegatedAdministrator`
    - `iam:CreateServiceLinkedRole`
  - The IPAM account must have an IAM policy attached to it that permits the `iam:CreateServiceLinkedRole` action.
- **When you create an IPAM using a single AWS account**

  If you Use IPAM with a single account (p. 6), the **AWSServiceRoleForIPAM** service-linked role is automatically created when you create an IPAM as that account.

  Important
  If you use IPAM with a single AWS account, before you create an IPAM, you must ensure that the AWS account you are using has an IAM policy attached to it that permits the `iam:CreateServiceLinkedRole` action. When you create the IPAM, you automatically create the **AWSServiceRoleForIPAM** service-linked role. For more information on managing IAM policies, see Editing IAM policies in the *IAM User Guide*.

Edit the service-linked role

You cannot edit the **AWSServiceRoleForIPAM** service-linked role.

Delete the service-linked role

If you no longer need to use IPAM, we recommend that you delete the **AWSServiceRoleForIPAM** service-linked role.

Note
You can delete the service-linked role only after you delete all IPAM resources in your AWS account. This ensures that you can't inadvertently remove the monitoring capability of IPAM.

Follow these steps to delete the service-linked role via the AWS CLI:

1. Delete your IPAM resources using deprovision-ipam-pool-cidr and delete-ipam. For more information, see Deprovision CIDRs from a pool (p. 20) and Delete an IPAM (p. 26).
2. Disable the IPAM account with disable-ipam-organization-admin-account.
3. Disable the IPAM service with disable-aws-service-access using the `--service-principal ipam.amazonaws.com` option.
4. Delete the service-linked role: delete-service-linked-role. When you delete the service-linked role, the IPAM managed policy is also deleted. For more information, see Deleting a service-linked role in the *IAM User Guide*.

AWS managed policies for IPAM

If you are using IPAM with a single AWS account and you create an IPAM, the **AWSIPAMServiceRolePolicy** managed policy is automatically created in your IAM account and attached to the **AWSServiceRoleForIPAM** service-linked role.

If you enable IPAM integration with AWS Organizations, the **AWSIPAMServiceRolePolicy** managed policy is automatically created in your IAM account and in each of your AWS Organizations member accounts, and the managed policy is attached to the **AWSServiceRoleForIPAM** service-linked role.

This managed policy enables IPAM to do the following:

- Monitor CIDRs associated with EC2 networking resources across all members of your AWS Organization.
- Store metrics related to IPAM in Amazon CloudWatch, such as the IP address space available in your IPAM pools and the number of resource CIDRs that comply with allocation rules.

The following example shows the details of the managed policy that's created.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeAddresses",
                "ec2:DescribeByoipCidrs",
                "ec2:DescribeIpv6Pools",
                "ec2:DescribePublicIpv4Pools",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "organizations:DescribeAccount",
                "organizations:DescribeOrganization",
                "organizations:ListAccounts",
                "organizations:ListDelegatedAdministrators"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "cloudwatch:PutMetricData",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "cloudwatch:namespace": "AWS/IPAM"
                }
            }
        }
    ]
}
```

The first statement in the preceding example enables IPAM to monitor the CIDRs used by your single AWS account or by the members of your AWS Organization.

The second statement in the preceding example uses the `cloudwatch:PutMetricData` condition key to allow IPAM to store IPAM metrics in your `AWS/IPAM` Amazon CloudWatch namespace. These metrics are used by the AWS Management Console to display data about the allocations in your IPAM pools and scopes. For more information, see Monitor CIDR usage with the IPAM dashboard (p. 28).

Updates to the AWS managed policy

View details about updates to AWS managed policies for IPAM since this service began tracking these changes.

Change	Description	Date
IPAM started tracking changes	IPAM started tracking changes for its AWS managed policies.	December 2, 2021

Quotas for your IPAM

This section lists the quotas related to IPAM. The Service Quotas console also provides information about IPAM quotas. You can use the Service Quotas console to view default quotas and request quota increases for adjustable quotas. For more information, see Requesting a quota increase in the *Service Quotas User Guide*.

Name	Default	Adjustable
IPAM administrators per organization	1	No
IPAMs per Region	1	No
Scopes per IPAM	5	Yes
Pools per scope	50	Yes
CIDRs per pool	50	Yes
Pool depth (the number of pools within pools)	10	Yes

Note
You cannot use IPAM to manage IP addresses across multiple AWS Organizations.

Pricing

You are charged hourly for each active IP address that IPAM monitors. An active IP address is defined as an IP address assigned to a resource such as an EC2 instance or an Elastic Network Interface (ENI). For more information, see IPAM pricing.

Document history for IPAM

The following table describes the releases for IPAM.

Feature	Description	Release Date
Initial release	This release introduces Amazon VPC IP Address Manager.	December 2, 2021