IP Address Manager

Amazon Virtual Private Cloud

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon Virtual Private Cloud: IP Address Manager

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

What is IPAM? 1
How IPAM works 2
Getting started with IPAM 4
Access IPAM 4
Configure integration options for your IPAM 5
Integrate IPAM with accounts in an AWS Organization 6
Integrate IPAM with accounts outside of your organization 8
Use IPAM with a single account 11
Create an IPAM 11
Plan for IP address provisioning 14
Example IPAM pool plans 15
Create IPv4 pools 17
Create IPv6 pools 27
Allocate CIDRs 35
Create a VPC that uses an IPAM pool CIDR 36
Manually allocate a CIDR to a pool to reserve IP address space 36
Managing IP address space in IPAM 38
Change the monitoring state of VPC CIDRs 39
Create additional scopes 40
Delete an IPAM 41
Delete a pool 43
Delete a scope 44
Deprovision CIDRs from a pool 45
Edit an IPAM pool 47
Enable cost distribution 48
Enable provisioning private IPv6 GUA CIDRs 49
Enforce IPAM use for VPC creation with SCPs 51
Enforce IPAM when creating VPCs 51
Enforce an IPAM pool when creating VPCs 52
Enforce IPAM for all but a given list of OUs 52
Exclude organizational units from IPAM 53
How OU exclusions work 54
Add or remove OU exclusions 55
Modify IPAM tier 61

Modify IPAM operating Regions ... 62
Provision CIDRs to a pool ... 63
Move VPC CIDRs between scopes ... 65
Release an allocation ... 66
Share an IPAM pool using AWS RAM ... 68
Work with resource discoveries ... 71
Create a resource discovery ... 71
View resource discovery details ... 73
Share a resource discovery ... 75
Associate a resource discovery with an IPAM ... 77
Disassociate a resource discovery ... 79
Delete a resource discovery ... 79
Tracking IP address usage in IPAM ... 81
Monitor CIDR usage with the IPAM dashboard ... 81
Monitor CIDR usage by resource ... 85
Monitor IPAM with Amazon CloudWatch ... 89
Pool and scope metrics ... 89
Resource utilization metrics ... 93
View IP address history ... 98
View public IP insights ... 101
Tutorials ... 106
Create an IPAM and pools using the console ... 106
Prerequisites ... 107
How AWS Organizations integrates with IPAM ... 107
Step 1: Delegate an IPAM administrator ... 108
Step 2: Create an IPAM ... 110
Step 3: Create a top-level IPAM pool ... 112
Step 4: Create Regional IPAM pools ... 117
Step 5: Create a pre-production development pool ... 121
Step 6: Share the IPAM pool ... 125
Step 7: Create a VPC with a CIDR allocated from an IPAM pool ... 130
Step 8: Cleanup ... 134
Create an IPAM and pools using the AWS CLI ... 136
Step 1: Enable IPAM in your organization ... 137
Step 2: Create an IPAM ... 137
Step 3: Create an IPv4 address pool ... 139

Step 4: Provision a CIDR to the top-level pool ... 141
Step 5. Create a Regional pool with CIDR sourced from the top-level pool ... 142
Step 6: Provision a CIDR to the Regional pool ... 144
Step 7. Create a RAM share for enabling IP assignments across accounts ... 145
Step 8. Create a VPC ... 146
Step 9. Cleanup ... 147
View IP address history using the AWS CLI ... 148
Overview ... 148
Scenarios ... 149
Bring your ASN to IPAM ... 156
Onboarding prerequisites for your ASN ... 157
Tutorial steps ... 158
Bring your IP addresses to IPAM ... 162
Verify domain control ... 163
BYOIP with AWS console and CLI ... 169
BYOIP with AWS CLI only ... 196
Transfer a BYOIP IPv4 CIDR to IPAM ... 243
Step 1: Create AWS CLI named profiles and IAM roles ... 244
Step 2: Get your IPAM's public scope ID ... 245
Step 3: Create an IPAM pool ... 246
Step 4: Share the IPAM pool using AWS RAM ... 248
Step 5: Transfer an existing BYOIP IPV4 CIDR to IPAM ... 250
Step 6: View the CIDR in IPAM ... 253
Step 7: Cleanup ... 253
Plan VPC IP address space for subnet IP allocations ... 257
Step 1: Create a VPC ... 258
Step 2: Create a resource planning pool ... 259
Step 3: Create subnet pools ... 260
Step 4: Create subnets ... 260
Step 5: Cleanup ... 261
Allocate sequential Elastic IP addresses from an IPAM pool ... 262
Step 1: Create an IPAM ... 263
Step 2: Create an IPAM pool and provision a CIDR ... 265
Step 3: Allocate an Elastic IP address from the pool ... 270
Step 4: Associate the Elastic IP address with an EC2 instance ... 271
Step 5: Track and monitor pool usage ... 272

Cleanup ... 273
Identity and access management in IPAM ... 275
Service-linked roles for IPAM ... 275
Service-linked role permissions ... 275
Create the service-linked role ... 276
Edit the service-linked role ... 277
Delete the service-linked role ... 277
Managed policies for IPAM ... 277
Updates to the AWS managed policy ... 279
Example policy ... 281
Quotas ... 284
Pricing ... 287
View pricing information ... 287
View your current costs and usage using AWS Cost Explorer ... 287
Related information ... 289
Document history ... 290

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 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
- Provision Amazon-provided contiguous IPv6 CIDR blocks to pools for VPC creation

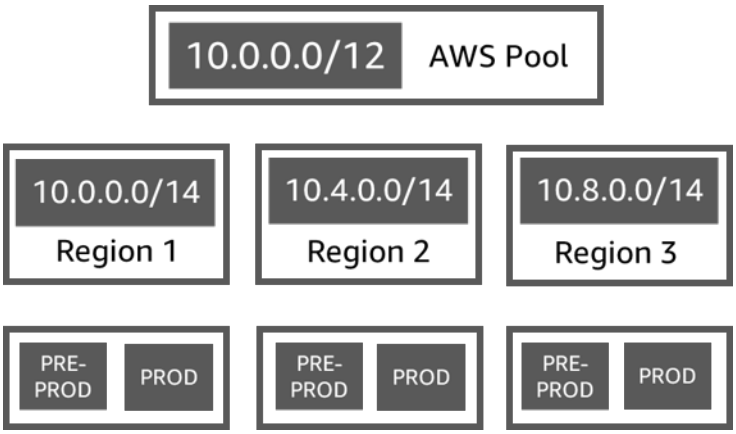
This guide consists of the following sections:

- How IPAM works: IPAM concepts and terminology.
- Getting started with IPAM: 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: Steps to manage your IPAM, scopes, pools, and allocations.
- Tracking IP address usage in IPAM: Steps to monitor and track IP address usage with IPAM.
- Tutorials for Amazon VPC IP Address Manager: 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. When you create IPAM, a default public scope and a default private scope are automatically created for you. Each scope represents the IP space for a single network. The **private scope** is intended for all the IP addresses that can't be advertised to the internet. The **public scope** is generally intended for all the IP addresses that can be advertised to the internet from AWS. Note that when provisioning BYOIPv6 addresses to an IPAM pool, you can configure the addresses to not be publicly advertisable though they are in the public scope. 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 public and private IPv6 space. For more information about public and private IPv6 addresses, see IPv6 addresses in the *Amazon VPC User Guide*.

To get started and create an IPAM, see Getting started with IPAM.

Getting started with IPAM

Follow the steps in this section to get started with IPAM. This section is intended to get you started quickly with IPAM, but you may find that what you can achieve with the steps in this section doesn't fit your needs. For information about different ways you can use IPAM, see Plan for IP address provisioning and Tutorials for Amazon VPC IP Address Manager.

In this section, 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.

Tasks

- Access IPAM
- Configure integration options for your IPAM
- Create an IPAM
- Plan for IP address provisioning
- Allocate CIDRs from an IPAM pool

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 to learn about the role of IPAM in Amazon VPC and then continue with the instructions in Configure integration options for your IPAM.

Configure integration options for your IPAM

This section describes your options for how you can integrate IPAM with AWS Organizations, other AWS accounts, or use it with a single AWS account.

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 accounts in an AWS Organization.
- After you integrate with AWS Organizations, to integrate IPAM with accounts outside of your organization, see Integrate IPAM with accounts outside of your organization.
- 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.

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 accounts in an AWS Organization
- Integrate IPAM with accounts outside of your organization
- Use IPAM with a single account

Integrate IPAM with accounts in an AWS Organization

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.

Note

When integrating with AWS Organizations:

- 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.
- **The IPAM account must be an AWS Organizations member account.** You cannot use the AWS Organizations management account as the IPAM account. To check whether your IPAM is already integrated with AWS Organizations, use the steps below and view the details of the integration in *Organization settings*.

- 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 use an IAM role that 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.
- The user associated with the AWS Organizations management account must use an IAM role that has the following IAM policy actions attached:
  - `ec2:EnableIpamOrganizationAdminAccount`
  - `organizations:EnableAwsServiceAccess`
  - `organizations:RegisterDelegatedAdministrator`
  - `iam:CreateServiceLinkedRole`

  For more information on creating IAM roles, see Creating a role to delegate permissions to an IAM user in the *IAM User Guide*.
- The user associated with the AWS Organizations management account may use an IAM role that has the following IAM policy actions attached to list your current AWS Orgs delegated administrators: `organizations:ListDelegatedAdministrators`

AWS Management Console

To select an IPAM account

1. Using the AWS Organizations management account, 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 **Organization settings**.
4. The **Delegate** option is only available if you've logged in to the console as the AWS Organizations management account. Choose **Delegate**.

5. Enter the AWS account ID for an IPAM account. The IPAM administrator must be an AWS Organizations member account.
6. Choose **Save changes**.

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.

Integrate IPAM with accounts outside of your organization

This section describes how to integrate your IPAM with AWS accounts outside of your organization. To complete steps in this section, you must have already completed the steps in Integrate IPAM with accounts in an AWS Organization and delegated an IPAM account.

Integrating IPAM with AWS accounts outside of your organization enables you to do the following:

- Manage IP addresses outside of your organization from a single IPAM account.

- Share IPAM pools with third-party services hosted by other AWS accounts in other AWS Organizations.

After you integrate IPAM with AWS accounts outside of your organization, you can share an IPAM pool directly with the desired accounts of other organizations.

Contents

- Considerations and limitations
- Process overview

Considerations and limitations

This section contains considerations and limitations for integrating IPAM with accounts outside of your organization:

- When you share a resource discovery with another account, the only data that is exchanged is IP address and account status monitoring data. You can view this data before sharing using the get-ipam-discovered-resource-cidrs and get-ipam-discovered-accounts CLI commands or GetIpamDiscoveredResourceCidrs and GetIpamDiscoveredAccounts APIs. For resource discoveries that monitor resources across an organization, no organization data (such as the names of Organizational Units in your organization) are shared.
- When you create a resource discovery, the resource discovery monitors all visible resources in the owner account. If the owner account is a third-party service AWS account that creates resources for multiple of their own customers, those resources will be discovered by the resource discovery. If the third-party AWS service account shares the resource discovery with an end-user AWS account, the end-user will have visibility into the resources of the other customers of the third-party AWS service. For that reason, the third-party AWS service should exercise caution creating and sharing resource discoveries or use a separate AWS account for each customer.

Process overview

This section explains how to integrate your IPAM with AWS accounts outside of your organization. It refers to topics that are covered in other sections of this guide. Keep this page visible, and open the topics linked below in a new window so that you can return to this page for guidance.

When you integrate IPAM with AWS accounts outside of your organization, there are 4 AWS accounts involved in the process:

- **Primary Org Owner** - The AWS Organizations management account for organization 1.
- **Primary Org IPAM Account** - The IPAM delegated administrator account for organization 1.
- **Secondary Org Owner** - The AWS Organizations management account for organization 2.
- **Secondary Org Admin Account** - The IPAM delegated administrator account for organization 2.

Steps

1. Primary Org Owner delegates a member of their organization as the Primary Org IPAM Account (see Integrate IPAM with accounts in an AWS Organization).
2. Primary Org IPAM Account creates an IPAM (see Create an IPAM).
3. Secondary Org Owner delegates a member of their organization as the Secondary Org Admin Account (see Integrate IPAM with accounts in an AWS Organization).
4. Secondary Org Admin Account creates a resource discovery and shares it with the Primary Org IPAM Account using AWS RAM (see Create a resource discovery to integrate with another IPAM and Share a resource discovery with another AWS account). The resource discovery must be created in the same home Region as the Primary Org IPAM.
5. Primary Org IPAM Account accepts the resource share invitation using AWS RAM (see Accepting and rejecting resource share invitations in the *AWS RAM User Guide*).
6. Primary Org IPAM Account associates the resource discovery with their IPAM (see Associate a resource discovery with an IPAM).
7. Primary Org IPAM Account can now monitor and/or manage IPAM resources created by the accounts in Secondary Org.
8. (Optional) Primary Org IPAM Account shares IPAM pools with member accounts in Secondary Org (see Share an IPAM pool using AWS RAM).
9. (Optional) If Primary Org IPAM Account wants to stop discovering resources in Secondary Org, it can disassociate the resource discovery from the IPAM (see Disassociate a resource discovery).
10. (Optional) If the Secondary Org Admin Account wants to stop participating in the Primary Org's IPAM, they can unshare the shared resource discovery (see Update a resource share in AWS RAM in the *AWS RAM User Guide*) or delete the resource discovery (see Delete a resource discovery).

Use IPAM with a single account

If you choose not to Integrate IPAM with accounts in an AWS Organization, 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 (IAM).

Service-linked roles are a type of IAM role that allows AWS services to access other AWS services on your behalf. They simplify the permission management process by automatically creating and managing the necessary permissions for specific AWS services to perform their required actions, streamlining the setup and administration of these services.

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.

> **Important**
>
> If you use IPAM with a single AWS account, you must ensure that the AWS account you use to create the IPAM uses a IAM role with a 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.

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.

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. Create the IPAM in your main Region of operations.
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. Choose an **IPAM tier**. For more information about the features available in each tier and the costs associated with the tiers, see the IPAM tab on the Amazon VPC pricing page.
6. 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.

   > **Note**
   >
   > If you are creating an IPAM in the Free Tier, you can select multiple operating Regions for your IPAM, but the only IPAM feature that will be available across operating Regions is Public IP insights. You cannot use other features in the Free Tier, like BYOIP, across the IPAM's operating Regions. You can only use them in the IPAM's home Region. To use all IPAM features across operating Regions, create an IPAM in the Advanced Tier.

7. Choose if you want to enable **Private IPv6 GUA CIDRs**. For more information about this option, see Enable provisioning private IPv6 GUA CIDRs.
8. Choose if you want to enable **Metering mode**. For more information about this option, see Enable cost distribution.
9. Choose **Create IPAM**.

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.

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. The pool creation process is different for pools in public and private scopes. This section includes steps for creating a regional pool in the private scope. For BYOIP and BYOASN tutorials, see Tutorials.

> **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 accounts in an AWS Organization.

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.

> **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. For more information on best practices, see Amazon VPC IP Address Manager Best Practices.

If you are creating a single IPAM pool, complete the steps in Create a top-level IPv4 pool and then skip to Allocate CIDRs from an IPAM pool.

Contents

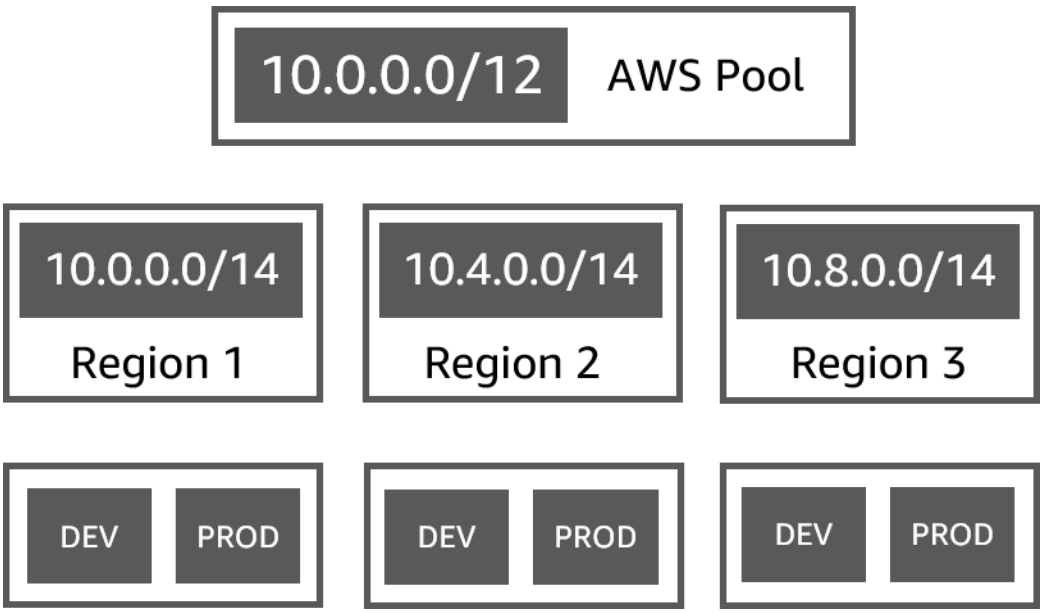
- Example IPAM pool plans
- Create IPv4 pools
- Create IPv6 address pools in your IPAM

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.

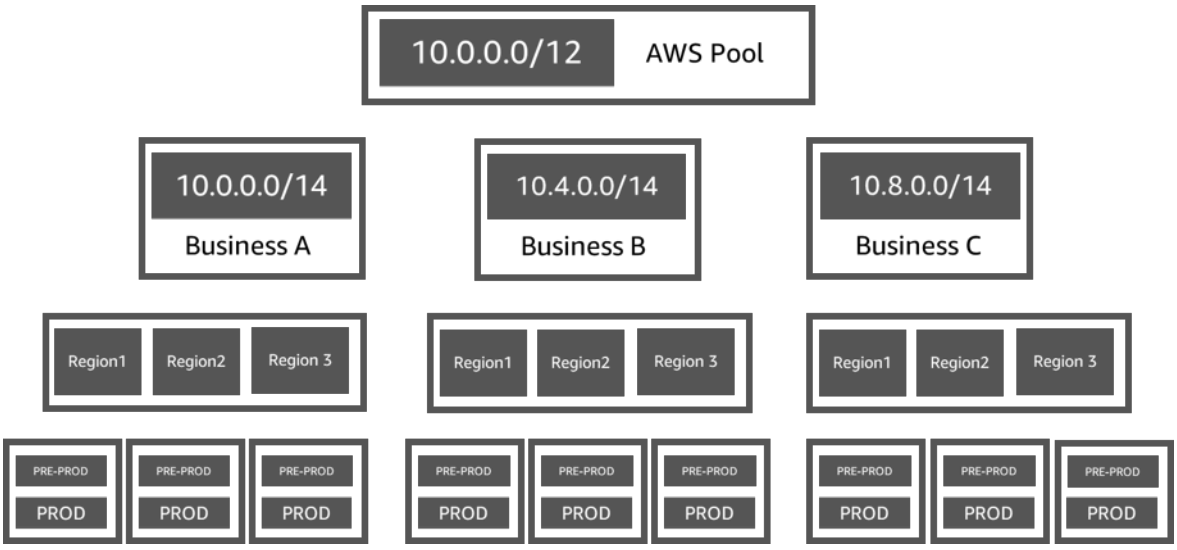
IPv4 pools in multiple AWS Regions

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 development resources and one pool for production resources.



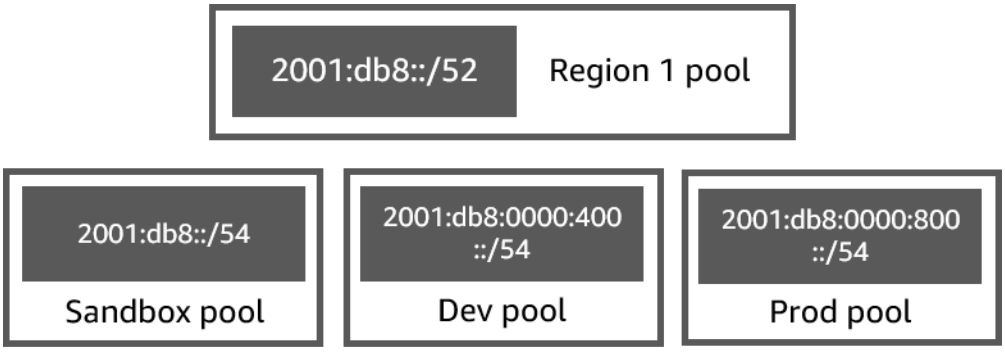
IPv4 pools for multiple lines of business

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.



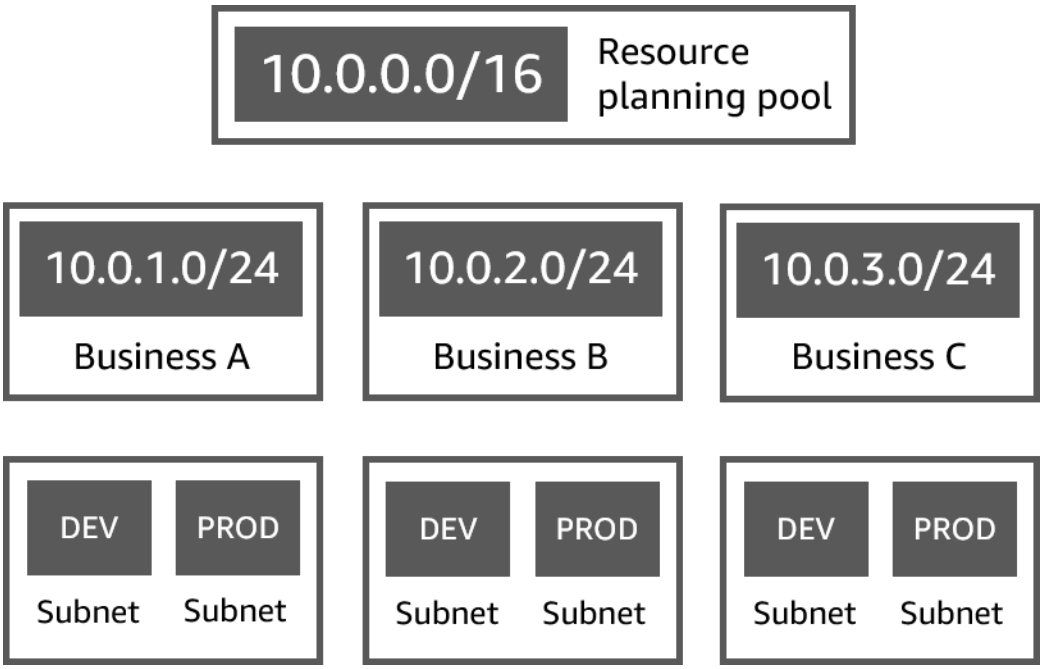
IPv6 pools in an AWS Region

The following example shows an IPAM IPv6 pool hierarchy for multiple lines of business within a Regional pool. Each Regional pool has three IPAM pools within it, one pool for sandbox resources, one pool for development resources, and one pool for production resources.



Subnet pools for multiple lines of business

The following example shows a resource planning pool hierarchy for multiple lines of business and dev/ prod subnet pools. For more information on subnet IP address space planning using IPAM, see Tutorial: Plan VPC IP address space for subnet IP allocations.



Create IPv4 pools

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

The following example shows the hierarchy of the pool structure that you can create with instructions in this guide. In this section, you are creating an IPv4 IPAM pool hierarchy:

- 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.

Contents

- Create a top-level IPv4 pool
- Create a Regional IPv4 pool
- Create a development IPv4 pool

Create a top-level IPv4 pool

Follow the steps in this section to create an IPv4 top-level IPAM pool. When you create the pool, you provision a CIDR for the pool to use. You then assign that space to an allocation. An allocation is a CIDR assignment from an IPAM pool to another IPAM pool or to a resource.

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 1 (10.0.0.0/16)
        - Development pool for non-production VPCs (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.

> **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.

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, creating a resource using a CIDR larger than /24 from this pool violates an allocation rule on the pool and creation fails. Existing resources with a CIDR larger than /24 are flagged as noncompliant.

Important

This topic covers how to create a top-level IPv4 pool with an IP address range provided by AWS. If you want to bring your own IPv4 address range to AWS (BYOIP), there are prerequisites. For more information, see Tutorial: Bring your IP addresses to IPAM.

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. Choose **Create pool**.
4. Under **IPAM scope**, choose the private scope you want to use. For more information about scopes, see How IPAM works.

   By default, when you create a pool, the default private scope is selected. Pools in the private scope must be IPv4 pools. Pools in the public scope can be IPv4 or IPv6 pools. The public scope is intended for all public space.
5. (Optional) Add a **Name tag** for the pool and a description for the pool.
6. Under **Source**, choose **IPAM scope**.
7. Under **Address family**, choose **IPv4**.
8. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.
9. 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. If the home Region of the IPAM is unavailable due to an outage and the pool has a locale different than the home Region of the IPAM, the pool can still be used to allocate IP addresses.
10. (Optional) 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. To provision a CIDR, choose **Add new**

CIDR. Enter an IPv4 CIDR to provision for the pool. If you want to bring your own IPv4 or IPv6 IP address range to AWS there are prerequisites. For more information, see Tutorial: Bring your IP addresses to IPAM.

11. 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.

  Warning
  - After you create an IPAM, when you create a VPC, choose the IPAM-allocated CIDR block option. If you do not, the CIDR you choose for your VPC may overlap with an IPAM CIDR allocation.
  - If you have a VPC already allocated in an IPAM pool, a VPC with an overlapping CIDR cannot be automatically imported. For example, if you have a VPC with a 10.0.0.0/26 CIDR allocated in an IPAM pool, a VPC with a 10.0.0.0/23 CIDR (that would cover the 10.0.0.0/26 CIDR) cannot be imported.
  - It takes some time for existing VPC CIDR allocations to be auto-imported into IPAM.

- **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. Choose **Create pool**.
14. See Create a Regional IPv4 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 IPv4 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 from an IPAM pool.

> **Note**
>
> The pool creation process is different for pools in public and private scopes. This section includes steps for creating a regional pool in the private scope. For BYOIP and BYOASN tutorials, see Tutorials.

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 1 (10.0.0.0/16)**
        - Development pool for non-production VPCs (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. Choose **Create pool**.
4. Under **IPAM scope**, choose the same scope that you used when you created the top-level pool. For more information about scopes, see How IPAM works.
5. (Optional) Add a **Name tag** for the pool and a description for the pool.
6. Under **Source**, choose **IPAM pool**. Then choose the top-level pool that you created in the previous section.
7. If you are creating this pool in the public scope, you'll see an option for **Address family**. Choose **IPv4**.

8. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.
9. 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. If the home Region of the IPAM is unavailable due to an outage and the pool has a locale different than the home Region of the IPAM, the pool can still be used to allocate IP addresses.

   > **Note**
   >
   > If you are creating a pool in the Free Tier, you can only choose the locale that matches the home Region of your IPAM. To use all IPAM features across locales, upgrade to the Advanced Tier.

10. If you are creating this pool in the public scope, you'll see an option for **Service**. Choose **EC2 (EIP/VPC)**. The service you select determines the AWS service where the CIDR will be advertisable. Currently, the only option is **EC2 (EIP/VPC)**, which means that the CIDRs allocated from this pool will be advertisable for the Amazon EC2 service (for Elastic IP addresses) and the Amazon VPC service (for CIDRs associated with VPCs).
11. (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.
12. You have the same allocation rule options here as you did when you created the top-level pool. See Create a top-level IPv4 pool 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.
13. (Optional) Choose **Tags** for the pool.
14. When you've finished configuring your pool, choose **Create pool**.
15. See Create a development IPv4 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 IPv4 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 from an IPAM pool.

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)

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. Choose **Create pool**.
4. Under **IPAM scope**, choose the same scope that you used when you created the top-level and Regional pools. For more information about scopes, see How IPAM works.
5. (Optional) Add a **Name tag** for the pool and a description for the pool.
6. Under **Source**, choose **IPAM pool**. Then choose the Regional pool.
7. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.
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 IPv4 pool 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**.
12. See Allocate CIDRs from an IPAM 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.

Create IPv6 address pools in your IPAM

AWS offers IPv6 connectivity across many of its services, including EC2, VPC, and S3, enabling you to use the increased address space and enhanced security features of IPv6. IPv6 was designed to resolve this fundamental limitation of IPv4. By moving to a 128-bit address space, IPv6 offers a large number of unique IP addresses. This massive address expansion enables the continued proliferation of connected technologies, from smartphones and IoT devices to cloud infrastructure.

In addition, you can use IPAM to ensure that you are using contiguous IPv6 CIDRs for VPC creation. Contiguously-allocated CIDRs are CIDRs that are allocated sequentially. They enable you to simplify your security and networking rules; the IPv6 CIDRs can be aggregated in a single entry across networking and security constructs like access control lists, route tables, security groups, and firewalls.

Follow the steps in this section to create an IPAM IPv6 pool hierarchy. When you create the pool, you can 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.

> **Note**
>
> Both public and private IPv6 addressing is available in AWS. AWS considers public IP addresses those advertised on the internet from AWS, while private IP addresses are not and cannot be advertised on the internet from AWS. If you want your private networks to support IPv6 and have no intention of routing traffic from these addresses to the internet, create your IPv6 pool in a private scope. For more information about public and private IPv6 addresses, see IPv6 addresses in the *Amazon VPC User Guide*.

The following example shows the hierarchy of the pool structure that you can create with instructions in this guide. In this section, you are creating an IPv6 IPAM pool hierarchy:

- IPAM operating in AWS Region 1 and AWS Region 2
  - Scope
    - Regional pool in AWS Region 1 (2001:db8::/52)
      - Development pool (2001:db8::/54)
        - Allocation for a VPC (2001:db8::/56)

In the preceding example, the CIDRs that are used are examples only. They illustrate that the Development pool within the Regional pool is provisioned with a portion of the Regional pool CIDR.

Contents

- Create a Regional IPv6 address pool in your IPAM
- Create a development IPv6 address pool in your IPAM

Create a Regional IPv6 address pool in your IPAM

Follow the steps in this section to create an IPv6 regional IPAM pool. When you provision an Amazon-provided IPv6 CIDR block to a pool, it must be provisioned to a pool with a locale (AWS Region) selected. When you create the pool, you can provision a CIDR for the pool to use or add it later. You then assign that space to an allocation. An allocation is a CIDR assignment from an IPAM pool to another IPAM pool or to a resource.

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 IPv6 regional IPAM pool:

- IPAM operating in AWS Region 1 and AWS Region 2
  - Scope
    - **Regional pool in AWS Region 1 (2001:db8::/52)**
      - Development pool (2001:db8::/54)
        - Allocation for a VPC (2001:db8::/56)

In the preceding example, the CIDRs that are used are examples only. They illustrate that each pool within the IPv6 regional pool is provisioned with a portion of the IPv6 regional 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:

- 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.

Note

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.

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.

Note

- This topic covers how to create an IPv6 regional pool with an IPv6 address range provided by AWS or with a private IPv6 range. If you want to bring your own public IPv4 or IPv6 IP address ranges to AWS (BYOIP), there are prerequisites. For more information, see Tutorial: Bring your IP addresses to IPAM.

- If you are creating an IPv6 pool in a private scope, you can use a private IPv6 GUA or ULA range. To use a private GUA range, you have to have first enabled the option on your IPAM (see Enable provisioning private IPv6 GUA CIDRs).

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. Choose **Create pool**.
4. Under **IPAM scope**, choose a private or public scope. If you want your private networks to support IPv6 and have no intention of routing traffic from these addresses to the internet, choose a private scope. For more information about scopes, see How IPAM works.

   By default, when you create a pool, the default private scope is selected.
5. (Optional) Add a **Name tag** for the pool and a description for the pool.
6. Under **Source**, choose **IPAM scope**.
7. For **Address family**, select **IPv6**. If you're creating this pool in the public scope, all CIDRs in this pool will be publicly advertisable.
8. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.
9. Choose the **Locale** for the pool. If you want to provision an Amazon-provided IPv6 CIDR block to a pool, it must be provisioned to a pool with a locale (AWS Region) selected. 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 for the IPAM when you created it. You can add additional operating Regions at any time.

   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. If the home Region of the IPAM is unavailable due to an outage and the pool has

a locale different than the home Region of the IPAM, the pool can still be used to allocate IP addresses.

> **Note**
>
> If you are creating a pool in the Free Tier, you can only choose the locale that matches the home Region of your IPAM. To use all IPAM features across locales, upgrade to the Advanced Tier.

10. (Optional) If you are creating an IPv6 pool in the public scope, under **Service**, choose **EC2 (EIP/VPC)**. The service you select determines the AWS service where the CIDR will be advertisable. Currently, the only option is **EC2 (EIP/VPC)**, which means that the CIDRs allocated from this pool will be advertisable for the Amazon EC2 service (for Elastic IP addresses) and the Amazon VPC service (for CIDRs associated with VPCs).
11. (Optional) If you are creating an IPv6 pool in the public scope, under **Public IP source** option, choose **Amazon owned** to have AWS provide an IPv6 address range for this pool. As noted at the top of this page, this topic covers how to create an IPv6 regional pool with an IP address range provided by AWS. If you want to bring your own IPv4 or IPv6 address range to AWS (BYOIP), there are prerequisites. For more information, see Tutorial: Bring your IP addresses to IPAM.
12. (Optional) 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. To provision a CIDR, do one of the following:

    - If you are creating an IPv6 pool in the public scope with **Public IP source Amazon-owned**, to provision a CIDR, under **CIDRs to provision**, choose **Add Amazon-owned CIDR** and choose the netmask size between /40 and /52 for the CIDR. When you choose a netmask length in the dropdown menu, you see the netmask length as well as the number of /56 CIDRs that the netmask represents. By default, you can add one Amazon-provided IPv6 CIDR block to the Regional pool. For information on increasing the default limit, see Quotas for your IPAM.
    - If you are creating an IPv6 pool in a private scope, you can use a private IPv6 GUA or ULA range:
        - For important details about private IPv6 addressing, see Private IPv6 addresses in the *Amazon VPC User Guide*.

- To use a private IPv6 ULA range, under **CIDRs to provision**, choose **Add ULA CIDR by netmask** and choose a netmask size or choose **Input private IPv6 CIDR** and enter a ULA range. Valid IPv6 ULA space is anything under fd00::/8 that does not overlap with the Amazon reserved range fd00::/16.
- To use a private IPv6 GUA range, you have to have first enabled the option on your IPAM (see Enable provisioning private IPv6 GUA CIDRs). Once you've enabled private IPv6 GUA CIDRs, enter an IPv6 GUA in **Input private IPv6 CIDR**.

13. Choose optional allocation rules for this pool:

- **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 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 `2001:db8::/52` and you enter 56 here, any new allocations in this pool will default to a netmask length of /56.
- **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. For example, if you enter /56 here, the smallest netmask length that can be allocated for CIDRs from this pool is /56.
- **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.

14. (Optional) Choose **Tags** for the pool.
15. Choose **Create pool**.
16. See Create a development IPv6 address pool in your IPAM.

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 an IPv6 regional pool in your IPAM:

1. If you want to enable provisioning private IPv6 GUA CIDRs, modify the IPAM with modify-ipam and include the option to `enable-private-gua`. For more information, see Enable provisioning private IPv6 GUA CIDRs.
2. Create a pool with create-ipam-pool.
3. Provision a CIDR to the pool: provision-ipam-pool-cidr.
4. Edit the pool after you create it to modify the allocation rules: modify-ipam-pool.

Create a development IPv6 address pool in your IPAM

Follow the steps in this section to create a development pool within your IPv6 Regional pool. If you only need a Regional pool and don't need development pools, skip to Allocate CIDRs from an IPAM pool.

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
  - Scope
    - Regional pool in AWS Region 1 (2001:db8::/52)
      - **Development pool (2001:db8::/54)**
        - Allocation for a VPC (2001:db8::/56)

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 an IPv6 Regional pool

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

2. In the navigation pane, choose **Pools**.
3. Choose **Create pool**.
4. Under **IPAM scope**, choose a scope. For more information about scopes, see How IPAM works.
5. (Optional) Add a **Name tag** for the pool and a description for the pool.
6. Under **Source**, choose **IPAM pool**. Then, under **Source pool**, choose the IPv6 Regional pool.
7. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.
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 IPv6 Regional pool. See Create a Regional IPv6 address pool in your IPAM 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**.
12. See Allocate CIDRs from an IPAM 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 an IPv6 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 IPv6 Regional pool, as needed.

Allocate CIDRs from an IPAM pool

One important feature of IPAM is the ability to allocate and manage IP address space. When creating a VPC, you must specify an IP address CIDR block, which defines the range of IP addresses available for that VPC. IPAM simplifies this process by providing a global view of your entire IP address inventory, helping you strategically assign and reuse IP prefixes across multiple VPCs.

This address space allocation is crucial for ensuring there are no overlapping IP ranges, which could cause routing conflicts and connectivity issues. IPAM also enables you to reserve IP address space for future VPC expansion, avoiding the need for complex renumbering later.

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.

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

Create a VPC that uses an IPAM pool CIDR

With Amazon Virtual Private Cloud (Amazon VPC), you can launch AWS resources in a logically isolated virtual network that you've defined. This virtual network closely resembles a traditional network that you'd operate in your own data center, with the benefits of using the scalable infrastructure of AWS.

A *virtual private cloud* (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. You can specify an IP address range for the VPC, add subnets, add gateways, and associate security groups.

Follow the steps in Create 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 for Amazon VPC IP Address Manager 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.

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.
4. In the content pane, choose a pool.
5. Choose **Actions** > **Create custom allocation**.
6. Choose whether to add a specific CIDR to allocate (for example, `10.0.0.0/24` for IPv4 or `2001:db8::/52` for IPv6) or add a CIDR by size by choosing the netmask length only (for example, `/24` for IPv4 or `/52` for IPv6).
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.

Managing IP address space in IPAM

The tasks in this section are optional. Note that this section is a grouping of procedures all related to working with IPAM. The procedures are ordered alphabetically.

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

- Change the monitoring state of VPC CIDRs
- Create additional scopes
- Delete an IPAM
- Delete a pool
- Delete a scope
- Deprovision CIDRs from a pool
- Edit an IPAM pool
- Enable cost distribution
- Enable provisioning private IPv6 GUA CIDRs
- Enforce IPAM use for VPC creation with SCPs
- Exclude organizational units from IPAM
- Modify IPAM tier
- Modify IPAM operating Regions
- Provision CIDRs to a pool
- Move VPC CIDRs between scopes
- Release an allocation
- Share an IPAM pool using AWS RAM
- Work with resource discoveries

Change the monitoring state of VPC CIDRs

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

Note

- You cannot ignore VPC CIDRs in the public scope.
- If a CIDR is ignored, you are still charged for the active IP addresses in the CIDR. For more information, see Pricing for IPAM.
- If a CIDR is ignored, you can still view the history of IP addresses in the CIDR. For more information, see View IP address history.

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

- **Monitored**: The VPC CIDR has been detected by IPAM and is being monitored for overlap with other CIDRs and allocation rule compliance.
- **Ignored**: The VPC CIDR has been chosen to be exempt from monitoring. Ignored VPC CIDRs are not evaluated for overlap with other CIDRs or Allocation rule compliance. Once a VPC CIDR is chosen to be ignored, any space allocated to it from an IPAM pool is returned to the pool and the VPC CIDR 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 CIDR allocated to a VPC

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 VPC and view the details of the VPC.

5. Under **VPC CIDRs**, select one of the CIDRs allocated to the VPC and choose **Actions** > **Mark as ignored** or **Unmark 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 VPC CIDR:

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

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 or private IPv6 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 VPC CIDRs between scopes.

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

Delete an IPAM

You may want to delete an IPAM if it is no longer needed, if you need to restructure your IP address management, or if you want to start fresh with a new IPAM configuration. Deleting an IPAM can help simplify your IP address management and align with changing business or operational requirements.

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.

> **Note**
>
> 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.
- Deprovision CIDRs provisioned to pools within the IPAM. For more information, see Deprovision CIDRs from a pool.
- Delete any additional non-default scopes. For more information, see Delete a scope.
- Delete your IPAM pools. For more information, see Delete a 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 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.

Delete a pool

An IPAM pool in AWS represents a defined range of IP addresses that can be allocated and managed within a specific AWS environment or organization. Pools are used to organize IP address space, enable automated IP address management, and enforce IP address governance policies across your cloud infrastructure.

You may want to delete an IPAM pool to remove unused or unnecessary IP address space and reclaim it for other purposes. 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 before you can delete the pool.

Follow the steps in this section to delete an IPAM 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.
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 IPv4 pool.

Delete a scope

You may want to delete an IPAM scope if it no longer serves its intended purpose, such as when you restructure your network, consolidate regions, or adjust your IP address allocation. Deleting unused scopes can help streamline your IPAM configuration and optimize your IP address management within AWS.

Note

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 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. To delete the IPAM, see Delete an IPAM.

Deprovision CIDRs from a pool

You may want to deprovision a pool CIDR to free up IP address space, simplify IP address management, prepare for network changes, or meet compliance requirements. Deprovisioning

a pool CIDR allows for better control and optimization of your IP address allocations within IPAM, while ensuring unused IP space is reclaimed and made available for future use. You can't deprovision the CIDR if there are allocations in the pool. To remove allocations, see the section called “Release an allocation”.

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.

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.
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. If you want to delete the pool, see Delete a pool.

Edit an IPAM pool

You may want to edit a pool to do one of the following:

- Change the allocation rules for the pool. For more information about allocation rules, see Create a top-level IPv4 pool.
- Modify the pool's name, description, or other metadata to improve organization and visibility within IPAM.
- Change pool options like auto-import discovered resources to optimize IPAM's automated IP address management.

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

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
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 IPv4 pool.
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

Enable cost distribution

When you enable cost distribution, you distribute the charges for active IP addresses to the accounts using the IP addresses rather than to the IPAM owner. This is useful for large organizations where the delegated IPAM admin manages the IP addresses centrally using IPAM and each account is responsible for their own usage, eliminating the need for manual billing calculations.

The cost distribution option is available when you create an IPAM or modify an IPAM under **Metering mode**, where:

- **IPAM owner** (default): The AWS account which owns the IPAM is charged for all active IP addresses managed in IPAM.
- **Resource owner**: The AWS account that owns the IP address is charged for the active IP address.

Requirements

- Your IPAM must be integrated with AWS Organizations.
- The IPAM must have been created by the delegated IPAM admin in your AWS Organization.
- The IPAM's home region must be a Region that's enabled by default. It cannot be an opt-in Region.

How charging works

- Even though you can distribute IP address charges within an organization, all IPAM charges are consolidated to the organization's payer account through AWS Organizations consolidated billing.
- When cost distribution is enabled, organization member accounts can still view their individual IPAM usage and charges in their account bills.
- The IPAM ARN will appear on individual account bills when cost distribution is enabled, which allows resource owners to track their IPAM active IP usage. If you use AWS Data Exports, IPAM charges appear with the associated IPAM ARN in both consolidated and individual account bills.
- Only accounts within the delegated administrator's organization can receive charges for the resources that they own. IP address costs outside of the organization are charged to the IPAM owner.

Time restrictions

- You have 24 hours to opt out after enabling cost distribution. After 24 hours, you cannot change the setting for 7 days. After 7 days, you can disable cost distribution.

Enable provisioning private IPv6 GUA CIDRs

If you want your private networks to support IPv6 and have no intention of routing traffic from these addresses to the internet, you can provision a private IPv6 ULA or GUA range to an IPAM pool in a private scope.

For important details about private IPv6 addressing, see Private IPv6 addresses in the *Amazon VPC User Guide*.

There are two types of private IPv6 addresses:

- **IPv6 ULA ranges**: IPv6 addresses as defined in RFC4193. These address ranges will always start with “fc” or “fd”, which makes them easily identifiable. Valid IPv6 ULA space is anything under fd00::/8 that does not overlap with the Amazon reserved range fd00::/16.
- **IPv6 GUA ranges**: IPv6 addresses as defined in RFC3587. The option to use IPv6 GUA ranges as private IPv6 addresses is disabled by default and must be enabled before you can use it.

To use an IPv6 ULA address ranges, you choose the IPv6 option when you provision a CIDR to an IPAM pool and enter the IPv6 ULA range. To use your own IPv6 GUA ranges as private IPv6 addresses, however, you must first complete the steps in this section. The option is disabled by default.

Note

- When you use private IPv6 GUA ranges, we require that you use IPv6 GUA ranges owned by you.
- IPAM discovers resources with IPv6 ULA and GUA addresses and monitors pools for overlapping IPv6 ULA and GUA address space.
- If you want to connect to the internet from a resource that has a private IPv6 address, you can do it, but you must route traffic through a resource in another subnet with a public IPv6 address to accomplish it.

- If you have a private IPv6 GUA range allocated to a VPC, you cannot use public IPv6 GUA space that overlaps the private IPv6 GUA space in the same VPC.
- Communication between resources with private IPv6 ULA and GUA address ranges is supported (such as across Direct Connect. VPC peering, transit gateway, or VPN connections).
- A private GUA IPv6 range cannot be converted to a publicly-advertised IPv6 GUA range.

AWS Management Console

To enable provisioning private IPv6 GUA CIDRs

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **IPAMs**.
3. Choose your IPAM and choose **Actions** > **Edit**.
4. Under **Private IPv6 GUA CIDRs**, choose **Enable provisioning GUA CIDR space into private IPv6 IPAM pools**.
5. Choose **Save changes**.

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 enable provisioning private IPv6 GUA CIDRs:

1. View current IPAMs with describe-ipams
2. Modify the IPAM with modify-ipam and include the option to `enable-private-gua`.

Once you enable the option to provision private IPv6 GUA CIDRs, you can provision a private IPv6 GUA CIDR to a pool. For more information, see Provision CIDRs to a pool.

Enforce IPAM use for VPC creation with SCPs

> **Note**
>
> This section is only applicable to you if you've enabled IPAM to integrate with AWS Organizations. For more information, see Integrate IPAM with accounts in an AWS Organization.

This section describes how to create a service control policy in AWS Organizations that requires members in your organization to use IPAM when they create a VPC. Service control policies (SCPs) are a type of organization policy that enable you to manage permissions in your organization. For more information, see Service control policies in the *AWS Organizations User Guide*.

Enforce IPAM when creating VPCs

Follow the steps in this section to require members in your organization to use IPAM when creating VPCs.

To create an SCP and restrict VPC creation to IPAM

1. Follow the steps in Create a service control policy 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": {
            "Null": {
                "ec2:Ipv4IpamPoolId": "true"
            }
        }
     }]
}
```

2. Attach the policy to one or more organizational units in your organization. For more information, see Attach policies and Detach policies in the *AWS Organizations User Guide*.

Enforce an IPAM pool when creating VPCs

Follow the steps in this section to require members in your organization to use a specific IPAM pool when creating VPCs.

To create an SCP and restrict VPC creation to an IPAM pool

1. Follow the steps in Create a service control policy 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. Attach the policy to one or more organizational units in your organization. For more information, see Attach policies and Detach policies in the *AWS Organizations User Guide*.

Enforce IPAM for all but a given list of OUs

Follow the steps in this section to enforce IPAM for all but a given list of Organizational Units (OUs). The policy described in this section requires OUs in the organization except for the OUs that you specify in `aws:PrincipalOrgPaths` to use IPAM to create and expand VPCs. The listed OUs can either use IPAM when creating VPCs or specify an IP address range manually.

To create an SCP and enforce IPAM for all but a given list of OUs

1. Follow the steps in Create a service control policy 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": {
         "Null": {
        "ec2:Ipv4IpamPoolId": "true"
                },
         "ForAnyValue:StringNotLike": {
             "aws:PrincipalOrgPaths": [
                 "o-a1b2c3d4e5/r-ab12/ou-ab12-11111111/ou-ab12-22222222/",
                 "o-a1b2c3d4e5/r-ab12/ou-ab13-22222222/ou-ab13-33333333/"
             ]
                }
            }
     }]
}
```

2. Remove the example values (like `o-a1b2c3d4e5/r-ab12/ou-ab12-11111111/ou-ab12-22222222/`) and add the AWS Organizations entity paths of the OUs that you want to have the option (but not require) to use IPAM. For more information about entity path, see Understand the AWS Organizations entity path and aws:PrincipalOrgPaths in the *IAM User Guide*.
3. Attach the policy to your organization root. For more information, see Attach policies and Detach policies in the *AWS Organizations User Guide*.

Exclude organizational units from IPAM

If your IPAM is integrated with AWS Organizations, you can exclude an organizational unit (OU) from being managed by IPAM. When you exclude an OU, IPAM will not manage the IP addresses in accounts in that OU. This feature gives you more flexibility in how you use IPAM.

You can use OU exclusions in the following ways:

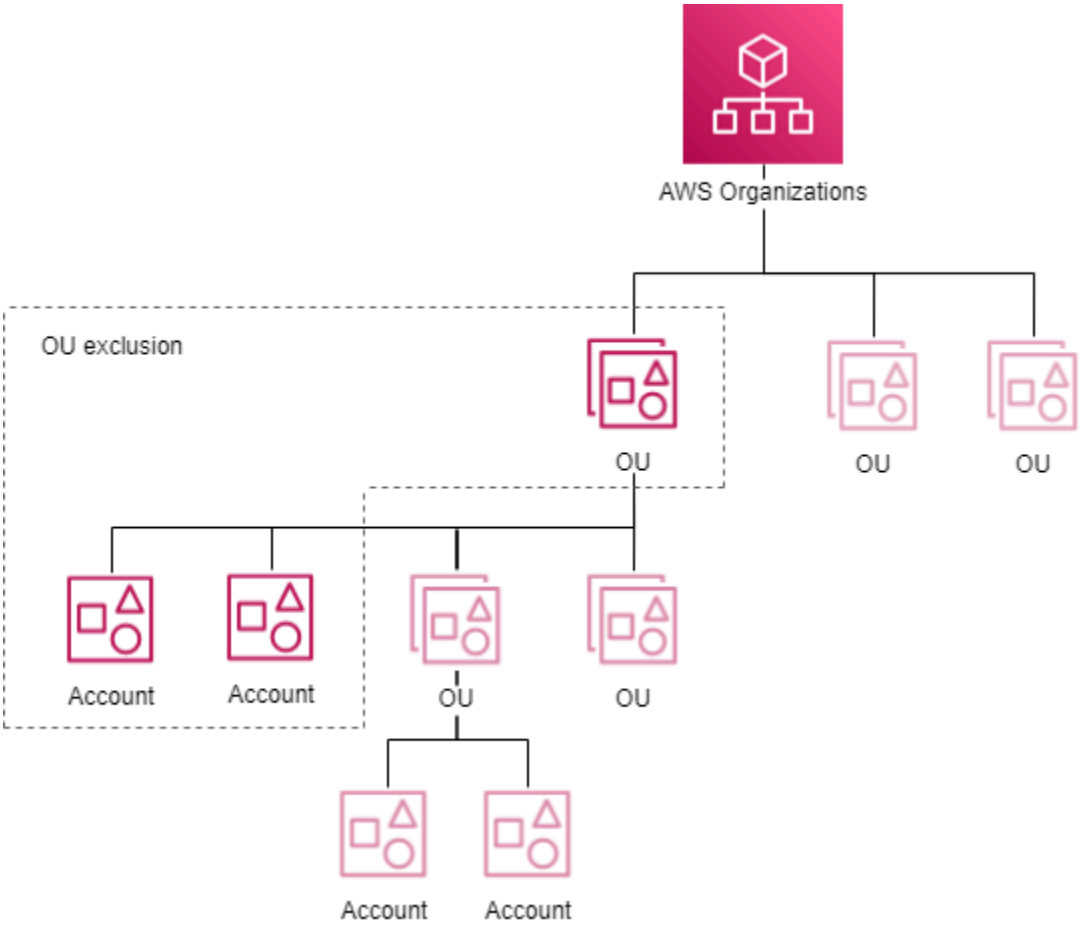
- **Enable IPAM for specific parts of your business**: If you have multiple business units or subsidiaries in AWS Organizations, you can now use IPAM just for the ones that need it.

- **Keep your sandbox accounts separate**: You can exclude your sandbox accounts from IPAM, focusing only on the accounts that really matter for your IP management.

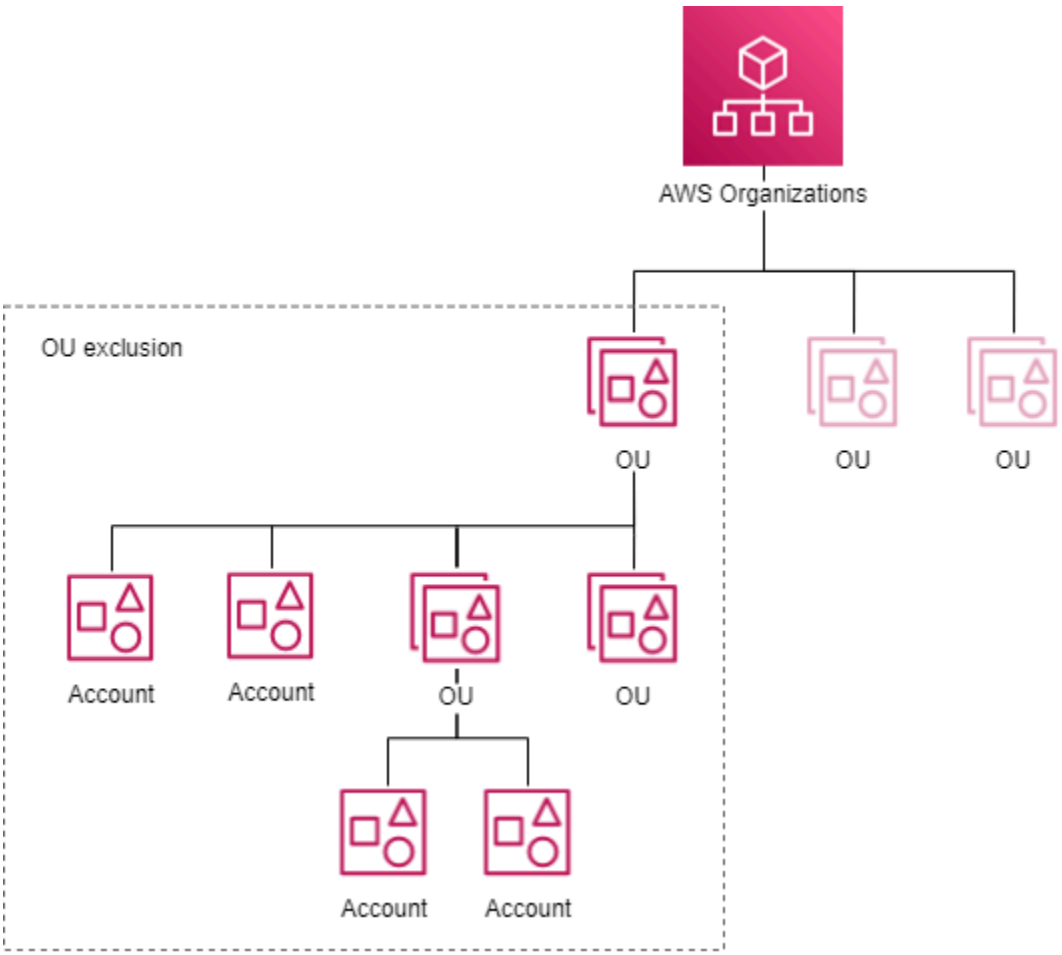
How OU exclusions work

The diagrams in this section demonstrate two use cases for adding OU exclusions in IPAM.

The first diagram shows the impact of adding an organization unit (OU) exclusion on a parent OU only. As a result, IPAM will not manage the IP addresses in accounts in the parent OU. IPAM will manage the IP addresses in accounts in the other OUs outside the exclusion.



The second diagram shows the impact of adding an organization unit (OU) exclusion on a parent OU *and* all child OUs. As a result, IPAM will not manage the IP addresses in accounts in the parent OU or in accounts in any child OUs. IPAM will manage the IP addresses in accounts in the OUs outside of the exclusion.



Add or remove OU exclusions

Complete the steps in this section to add or remove OU exclusions.

Note

- The delegated IPAM admin account is not excluded even if it's within an OU that's excluded.
- Your IPAM must be integrated with AWS Organizations to add an OU exclusion. The Organization must have OUs in it.
- You must be the delegated IPAM admin to view, add, or remove OU exclusions.
- It takes time for IPAM to discover recently created organizational units.
- There is a default quota for the number of exclusions you can add per resource discovery. For more information, see *Organizational unit exclusions per resource discovery* in Quotas for your IPAM.

- If you share a resource discovery with another account, that account can see the OU exclusions on it, which contains information such as the Org ID, Root ID, and organizational unit IDs of the resource discovery owner's Organization.

AWS Management Console

To add or remove OU exclusions

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Resource discoveries**.
3. Choose your default resource discovery.
4. Choose **Edit**.
5. Under **Organizational unit exclusions**, do the following:

   - **To add an OU exclusion**:
     - If you want to exclude the OU and all its child OUs:
       - Find the OU in the table and select the checkbox. All child OUs are automatically selected.
     - If you want to exclude only parent OU accounts:
       - Find the OU in the table and select the checkbox. All child OUs are automatically selected. Deselect all child OUs.
     - Alternatively, you can use the **Actions** column to select only a parent OU or parent and child OUs:
       - **Select all child OUs**: Include any child OUs in the exclusion. As a result of choosing an OU, the OU is added on screen. Each OU contains the ID and the entity path of the OU exclusion.
       - **Select only this OU**: Include only this OU in the exclusion. As a result of choosing an OU, the OU is added on screen. Each OU contains the ID and the entity path of the OU exclusion.
       - **Copy OU entity path**: Copy the AWS Organizations entity path to use as needed.
     - If you know the AWS Organizations entity path already or you want to build it:
       - Choose **Input organizational unit exclusion** and enter the entity path of the OU exclusion. Build the path for the OU(s) using AWS Organizations IDs separated by a `/`. Include all child OUs by ending the path with `/*`.

- Example 1
  - Path to a child OU: `o-a1b2c3d4e5/r-f6g7h8i9j0example/ou-ghi0-awsccccc/ou-jkl0-awsddddd/`
  - In this example, `o-a1b2c3d4e5` is the organization ID, `r-f6g7h8i9j0example` is the root ID, `ou-ghi0-awsccccc` is an OU ID, and `ou-jkl0-awsddddd` is a child OU ID.
  - IPAM will not manage the IP addresses in accounts in the child OU.
- Example 2
  - Path where all child OUs will be part of the exclusion: `o-a1b2c3d4e5/r-f6g7h8i9j0example/ou-ghi0-awsccccc/*`
  - In this example, IPAM will not manage the IP addresses in accounts in the OU (`ou-ghi0-awsccccc`) or in accounts in any OUs that are children of the OU.

- **To remove an OU exclusion**:
  - Choose the **X** next to an OU that's already been added. The `/*` after the OU ID indicates that it's a parent OU and that child OUs are part of the OU exclusion.

6. Choose **Save changes**.

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.

1. View resource discovery details to get the ID of the default resource discovery for the next step with describe-ipam-resource-discoveries.

   Input:

   ```
   aws ec2 describe-ipam-resource-discoveries
   ```

   Output:

   ```
   {

       "IpamResourceDiscoveries": [
   ```

```
        {

            "OwnerId": "111122223333",

            "IpamResourceDiscoveryId": "ipam-res-disco-1234567890abcdef0",

            "IpamResourceDiscoveryArn": "arn:aws:ec2::111122223333:ipam-
resource-discovery/ipam-res-disco-1234567890abcdef0",
            "IpamResourceDiscoveryRegion": "us-east-1",

            "OperatingRegions": [

                {

                    "RegionName": "us-east-1"

                },

                {

                    "RegionName": "us-west-1"

                },

                {

                    "RegionName": "us-west-2"

                }

            ],

            "IsDefault": true,

            "State": "modify-complete",

            "Tags": []

        }

    ]
```

```
}
```

2. Add or remove an organizational unit exclusion from a resource discovery with modify-ipam-resource-discovery and the `--add-organizational-unit-exclusions` or `--remove-organizational-unit-exclusions` options. You'll need enter an AWS Organizations entity path. Build the path for the OU(s) using AWS Organizations IDs separated by a `/`. Include all child OUs by ending the path with `/*`. You can't include the same entity path more than once in the add or remove parameters.

   - Example 1
     - Path to a child OU: `o-a1b2c3d4e5/r-f6g7h8i9j0example/ou-ghi0-awsccccc/ou-jkl0-awsddddd/`
     - In this example, `o-a1b2c3d4e5` is the organization ID, `r-f6g7h8i9j0example` is the root ID, `ou-ghi0-awsccccc` is an OU ID, and `ou-jkl0-awsddddd` is a child OU ID.
     - IPAM will not manage the IP addresses in accounts in the child OU.
   - Example 2
     - Path where all child OUs will be part of the exclusion: `o-a1b2c3d4e5/r-f6g7h8i9j0example/ou-ghi0-awsccccc/*`
     - In this example, IPAM will not manage the IP addresses in accounts in the OU (`ou-ghi0-awsccccc`) or in accounts in any OUs that are children of the OU.

   Note

   The resulting set of exclusions must not "overlap", meaning two or more OU exclusions must not exclude the same OU.
   Example of non-overlapping entity paths:

   - Path 1 ="o-1/r-1/ou-1/"
   - Path 2 ="o-1/r-1/ou-1/ou-2/"

   These paths are do not overlap because Path 1 only excludes the accounts under ou-1 and Path 2 only excludes accounts under ou-2.
   Example of overlapping entity paths:

   - Path 1 ="o-1/r-1/ou-1/*"

- Path 2 ="o-1/r-1/ou-1/ou-2/"

These paths overlap because Path 1 represents both "o-1/r-1/ou-1/" and "o-1/r-1/ou-1/ou-2/", and "o-1/r-1/ou-1/ou-2/" overlaps with Path 2.

Input:

```
aws ec2 modify-ipam-resource-discovery \
    --ipam-resource-discovery-id ipam-res-disco-1234567890abcdef0 \
    --add-organizational-unit-exclusions OrganizationsEntityPath='o-a1b2c3d4e5/
r-f6g7h8i9j0example/ou-ghi0-awsccccc/*' \
    --remove-organizational-unit-exclusions OrganizationsEntityPath='o-
a1b2c3d4e5/r-f6g7h8i9j0example/ou-ghi0-awsccccc/ou-jkl0-awsddddd/' \
    --region us-east-1
```

Output:

```
{
    "IpamResourceDiscovery": {
        "OwnerId": "111122223333",
        "IpamResourceDiscoveryId": "ipam-res-disco-1234567890abcdef0",
        "IpamResourceDiscoveryArn": "arn:aws:ec2::111122223333:ipam-resource-
discovery/ipam-res-disco-1234567890abcdef0",
        "IpamResourceDiscoveryRegion": "us-east-1",
        "OperatingRegions": [
            {
                "RegionName": "us-east-1"
            }
        ],
        "IsDefault": false,
        "State": "modify-in-progress",
        "OrganizationalUnitExclusions": [
            {
                "OrganizationsEntityPath": "o-a1b2c3d4e5/r-f6g7h8i9j0example/ou-
ghi0-awsccccc/*"
            }
        ]
    }
}
```

Modify IPAM tier

IPAM offers two tiers: Free Tier and Advanced Tier. Switching to the Advanced Tier of Amazon VPC IP Address Manager provides more granular control over your IP address management. This can be beneficial as your network complexity grows, allowing you to better optimize and manage your IP address space. For more information about the features available in the Free Tier and the costs associated with the Advanced Tier, see the IPAM tab in the Amazon VPC pricing page.

Note

Before you can switch from the Advanced Tier to the Free Tier, you must:

- Delete private scope pools.
- Delete non-default private scopes.
- Delete pools with locales different than the IPAM home Region.
- Delete non-default resource discovery associations.
- Delete pool allocations to accounts that are not the IPAM owner.

AWS Management Console

To modify the IPAM tier

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** > **Edit**.

Note

If you are in the Free Tier, you will see **Your estimated IPAM total active IP count is...**.
The total active IP count is the number of active IP addresses in your IPAM that you would be charged if you switched from the Free Tier to the Advanced Tier. An active IP address is defined as an IP address or a prefix associated with an Elastic Network Interface (ENI) that is attached to a resource such as an EC2 Instance.

- This metric is only available to customers in the Free Tier.
- If your IPAM is integrated with AWS Organizations, the active IP count covers all the Organization accounts.
- You cannot view a breakdown of the active IP count by IP type (public/private) or class (IPv4/IPv6).
- IPAM only counts IPs from ENIs owned by monitored accounts. The count may be inaccurate for shared subnets. IP addresses are excluded if the subnet owner or ENI owner is not covered by IPAM.

5. Choose the **IPAM tier** you want to use for the IPAM.
6. Choose **Save changes**.

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 view and modify an IPAM tier:

1. View current IPAMs: describe-ipams
2. Modify the IPAM tier: modify-ipam
3. View your updated IPAMs: describe-ipams

Modify IPAM operating Regions

Operating Regions are AWS Regions where the IPAM is allowed to manage IP address CIDRs. IPAM only discovers and monitors resources in the AWS Regions you select as operating Regions.

Adding an operating region to an IPAM allows you to manage IP address space across multiple AWS Regions. This can improve IP address utilization, enable regional segmentation, and support geographically distributed infrastructure. Expanding the IPAM's Regional scope provides greater flexibility and control over your overall IP address management.

AWS Management Console

To modify the IPAM operating Regions

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** > **Edit**.
5. Under **IPAM settings**, choose the **Operating Regions** you want to use for the IPAM.
6. Choose **Save changes**.

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 view and modify IPAM operating Regions:

1. View current IPAMs: describe-ipams
2. Add or remove IPAM operating Regions: modify-ipam
3. View your updated IPAMs: describe-ipams

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.

> **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 VPC or Elastic IP address.

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.
4. In the content pane, choose the pool that you want to add a CIDR to.
5. Choose **Actions** > **Provision CIDRs**.
6. Do one of the following:
   - If you are provisioning a CIDR to a pool in the public scope, enter the **Netmask**.
   - If you are provisioning a CIDR to an IPv4 pool in the private scope, enter the **CIDR**.
   - If you are provisioning a CIDR to an IPv6 pool in the private scope, note the following:
     - For important details about private IPv6 addressing, see Private IPv6 addresses in the *Amazon VPC User Guide*.
     - To use a private IPv6 ULA range, under **CIDRs to provision**, choose **Add ULA CIDR by netmask** and choose a netmask size or choose **Input private IPv6 CIDR** and enter a ULA range. Valid ranges for private IPv6 ULA are /9 to /60 starting with fd80::/9.
     - To use a private IPv6 GUA range, you have to have first enabled the option on your IPAM (see Enable provisioning private IPv6 GUA CIDRs). Once you have enabled private IPv6 GUA CIDRs, enter an IPv6 GUA in **Input private IPv6 CIDR**.

   Note

   - By default, you can add one Amazon-provided IPv6 CIDR block to a Regional pool. For information on increasing the default limit, see Quotas for your IPAM.
   - 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 **Provision**.

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

Move VPC CIDRs between scopes

Moving CIDRs between scopes allows you to optimize IP address allocation, organize by Region, separate concerns, enforce compliance, and adapt to infrastructure changes. This flexibility helps manage your IP address space efficiently as your workloads evolve.

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

Important

- You can only move VPC CIDRs. When you move a VPC CIDR, the VPC's subnet CIDRs are moved automatically as well.
- You can only move VPC CIDRs from one private scope to another. You cannot move VPC CIDRs out of a public scope to a private scope or from a private scope to a public scope.
- The same AWS account must own both scopes.
- If a VPC CIDR is currently allocated from a pool in a private scope, the move request succeeds, but the VPC CIDR will not be moved until you release the VPC CIDR allocation from the current pool. For information on releasing an allocation, see Release an allocation.

AWS Management Console

To move a CIDR allocated to a VPC

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 VPC and view the details of the VPC.
5. Under **VPC 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 VPC CIDR to.
7. Choose **Move CIDR to different scope**.

Command line

Use the following AWS CLI commands to move a VPC CIDR:

1. Get a VPC CIDR in current scope: get-ipam-resource-cidrs
2. Move a VPC CIDR: modify-ipam-resource-cidr
3. Get a VPC CIDR in the other scope: get-ipam-resource-cidrs

Release an allocation

If you are planning to delete a pool, you might need to release a pool allocation. An allocation is a CIDR assignment from an IPAM pool to another resource or IPAM pool.

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 VPC CIDRs. 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 or *Cleanup* in Bring your own IPv6 CIDR to IPAM using only the AWS CLI. 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 or multi-account use.

When you release a CIDR that's managed by your IPAM, Amazon VPC IPAM recycles the CIDR back into an IPAM pool. If you are using IPAM in the Advanced Tier, it takes a few minutes for the CIDR to become available for future allocations. If you are using IPAM in the Free Tier, it will take up to 48 hours for the CIDR to become available for future allocations. For more information about pools and allocations, see How IPAM works.

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.
4. In the content pane, choose the pool that the allocation is in.
5. Choose the **Allocations** tab.

6. Select one or more allocations. You can identify allocations by their **Resource type**:
   - **custom**: A custom allocation.
   - **vpc**: A VPC allocation.
   - **ipam-pool**: An IPAM pool allocation.
   - **ec2-public-ipv4-pool**: A public IPv4 pool allocation.
   - **subnet**: A subnet allocation.
7. Choose **Actions** > **Release custom allocation**.
8. 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 from an IPAM pool. To delete the pool after releasing allocations, you must first Deprovision CIDRs from a 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 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 accounts in an AWS Organization. You cannot share an IPAM pool with AWS RAM if 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 the home AWS Region of your IPAM. 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 the IAM policy attached to their IAM role:
  - `ec2:PutResourcePolicy`
  - `ec2:DeleteResourcePolicy`
- You can add multiple IPAM pools to a RAM share.
- While you can share IPAM pools with any AWS account outside an AWS Organization, IPAM will only monitor the IP addresses in accounts outside the Organization if the account owner has gone through the process of sharing their resource discovery with the delegated IPAM admin as described in Integrate IPAM with accounts outside of your organization.

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.
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 a BYOIP IPv4 CIDR to IPAM.
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. For more information on how to create a VPC and allocate a CIDR from a shared IPAM pool, see Create a VPC in the *Amazon VPC User Guide*.

Work with resource discoveries

A resource discovery is an IPAM component that enables IPAM to manage and monitor resources that belong to the account that owns the resource discovery. This enables IPAM to maintain an up-to-date inventory of IP address usage across your workloads, facilitating IP address management and planning.

A resource discovery is created by default when you create an IPAM. You can also create a resource discovery independently of an IPAM and integrate it with an IPAM owned by another account or organization. If the resource discovery owner is the delegated administrator of an organization, IPAM will monitor resources for all members of the organization.

> **Note**
>
> Creating, sharing, and associating resource discoveries is part of the process of integrating IPAM with accounts outside of your organizations (see Integrate IPAM with accounts outside of your organization). If you are not creating an IPAM and integrating it with accounts outside your organization, you do not need to create, share, or associate resource discoveries.

Note that this section is a grouping of procedures all related to working with resource discoveries.

Contents

- Create a resource discovery to integrate with another IPAM
- View resource discovery details
- Share a resource discovery with another AWS account
- Associate a resource discovery with an IPAM
- Disassociate a resource discovery
- Delete a resource discovery

Create a resource discovery to integrate with another IPAM

This section describes how to create a resource discovery. A resource discovery is created by default when you create an IPAM. The default quota for resource discoveries per Region is 1. For more information about IPAM quotas, see Quotas for your IPAM.

> **Note**
>
> Creating, sharing, and associating resource discoveries is part of the process of integrating IPAM with accounts outside of your organizations (see Integrate IPAM with accounts outside of your organization). If you are not creating an IPAM and integrating it with accounts outside your organization, you do not need to create, share, or associate resource discoveries.

If you are integrating an IPAM with accounts outside of your organizations, this is a required step that must be completed by the **Secondary Org Admin Account**. For more information about the roles involved in this process, see Process overview.

AWS Management Console

To create a resource discovery

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Resource discoveries**.
3. Choose **Create resource discovery**.
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 a resource discovery.
5. (Optional) Add a **Name** tag to the resource discovery. A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.
6. (Optional) Add a description.
7. Under **Operating regions**, select the AWS Regions in which resources will be discovered. The current Region will automatically be set as one of the operating Regions. If you're creating the resource discovery so that you can share it with an IPAM in operating Region `us-east-1`, make sure you select `us-east-1` here. If you forget an operating Region, you can return at a later time and edit your resource discovery settings.

> **Note**
>
> In most cases, the resource discovery should have the same operating Regions as the IPAM or you will only get resource discovery in that one Region.

8. (Optional) Choose any additional **Tags** for the pool.
9. 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.

- Create a resource discovery: create-ipam-resource-discovery

View resource discovery details

Viewing the details of a resource discovery in AWS IPAM can provide valuable insights, such as:

- Identifying the specific AWS resources that have been imported and their associated IP address allocations.
- Monitoring the status and progress of the resource discovery process.
- Troubleshooting any issues or discrepancies between IPAM and the discovered resources.
- Analyzing the IP address utilization and trends.

This information can help you optimize your IP address management and ensure alignment between IPAM and your actual resource deployments.

AWS Management Console

To view resource discovery details

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Resource discoveries**.
3. Choose a resource discovery.

4. Under **Resource discovery details**, view details related to the resource discovery, such as Default, which indicates whether the resource discovery is the default. The default resource discovery is the resource discovery automatically created when you create an IPAM.
5. In the tabs, view the details of a resource discovery:

   - **Discovered resources** - Resources monitored under a resource discovery. IPAM monitors CIDRs from the following resource types VPCs, Public IPv4 pools, VPC subnets, and Elastic IP addresses.
     - **Name (Resource ID)** – Resource discovery ID.
     - **IPs allocated** – The percentage of IP address space in use. To convert the decimal to a percentage, multiply the decimal by 100. Note the following:
       - For resources that are VPCs, this is the percentage of IP address space in the VPC that's taken up by subnet CIDRs.
       - For resources that are subnets, if the subnet has an IPv4 CIDR provisioned to it, this is the percentage of IPv4 address space in the subnet that's in use. If the subnet has an IPv6 CIDR provisioned to it, the percentage of IPv6 address space in use is not represented. The percentage of IPv6 address space in use cannot currently be calculated.
       - For resources that are public IPv4 pools, this is the percentage of IP address space in the pool that's been allocated to Elastic IP addresses (EIPs).
     - **CIDR** – Resource CIDR.
     - **Region** – Resource Region.
     - **Owner ID** – Resource owner ID.
     - **Sample time** – The last successful resource discovery time.
   - **Discovered accounts**: AWS accounts being monitored under a resource discovery. If you have integrated IPAM with AWS Organizations, all accounts in the organization are discovered accounts.
     - **Account ID** – The account ID.
     - **Region** – The AWS Region that the account information is returned from.
     - **Last attempted discovery time** – The last attempted resource discovery time.
     - **Last successful discovery time** – The last successful resource discovery time.
     - **Status** – Resource discovery failure reason.
   - **Operating regions** – The operating Regions for the resource discovery.

- **Resource sharing** – If the resource discovery has been shared, the resource share ARN is listed.
  - **Resource share ARN** – Resource share ARN.
  - **Status** – The current status of the resource share. Possible values are:
    - **Active** – Resource share is active and available for use.
    - **Deleted** – Resource share is deleted and is no longer available for use.
    - **Pending** – An invitation to accept the resource share is waiting for a response.
  - **Created at** – When the resource share was created.
- **Tags** – A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

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 resource discovery details: describe-ipam-resource-discoveries

Share a resource discovery with another AWS account

Follow the steps in this section to share a resource discovery using AWS Resource Access Manager. For more information about AWS RAM, see Sharing your AWS resources in the *AWS RAM User Guide*.

> **Note**
>
> Creating, sharing, and associating resource discoveries is part of the process of integrating IPAM with accounts outside of your organizations (see Integrate IPAM with accounts outside of your organization). If you are not creating an IPAM and integrating it with accounts outside your organization, you do not need to create, share, or associate resource discoveries.

When you create an IPAM that monitors accounts outside your organization, the Secondary Org Admin Account shares their resource discovery with the Primary Org IPAM Account using AWS RAM. You must first share a resource discovery with the Primary Org IPAM Account before the Primary Org IPAM Account can associate the resource discovery with their IPAM. For more information about the roles involved in this process, see Process overview.

Note

- When you create a resource share using AWS RAM to share a resource discovery, you must create the resource share in the home Region of the Primary Org IPAM.
- The account that creates and deletes a resource share for a resource discovery must have the following permissions in their IAM policy:
  - ec2:PutResourcePolicy
  - ec2:DeleteResourcePolicy
- If you share a resource discovery with another account, that account can see any OU exclusions on it, which contains information such as the Org ID, Root ID, and organizational unit IDs of the resource discovery owner's Organization.

If you are integrating an IPAM with accounts outside of your organizations, this is a required step that must be completed by the **Secondary Org Admin Account**.

AWS Management Console

To share a resource discovery

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Resource discoveries**.
3. Choose the **Resource sharing** tab.
4. Choose **Create resource share**. The AWS RAM console opens, which is where you will create the resource share.
5. In the AWS RAM console, choose **Settings**.
6. Choose **Enable sharing with AWS Organizations**, and then choose **Save settings**.
7. Choose **Create a resource share**.
8. Add a **Name** for the shared resource.

9. Under **Select resource type**, select **IPAM Resource Discovery**, and choose the resource discovery.
10. Choose **Next**.
11. Under **Associate permissions**, you can view the default permission that will be enabled for principals that are granted access to this resource share:
    - AWSRAMPermissionIpamResourceDiscovery
    - Actions allowed by this permission:
      - ec2:AssociateIpamResourceDiscovery
      - ec2:GetIpamDiscoveredAccounts
      - ec2:GetIpamDiscoveredPublicAddresses
      - ec2:GetIpamDiscoveredResourceCidrs
12. Specify the principals that are allowed access to the shared resource. For **Principals**, choose the Primary Org IPAM Account, and then choose **Add**.
13. Choose **Next**.
14. Review the resource share options and the principals that you'll be sharing with. Then choose **Create resource share**.
15. After a resource discovery is shared, it must be accepted by the Primary Org IPAM Account and then associated with an IPAM by the Primary Org IPAM Account. For more information, see Associate a resource discovery with an IPAM.

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.

1. Create the resource share: create-resource-share
2. View the resource share: get-resource-shares

Associate a resource discovery with an IPAM

This section describes how to associate a resource discovery with an IPAM. When you associate a resource discovery with an IPAM, the IPAM monitors all resources CIDRs and accounts discovered

under the resource discovery. When you create an IPAM, a default resource discovery is created for your IPAM and automatically associated with your IPAM.

The default quota for resource discovery associations is 5. For more information (including how to adjust this quota), see Quotas for your IPAM.

> **Note**
>
> Creating, sharing, and associating resource discoveries is part of the process of integrating IPAM with accounts outside of your organizations (see Integrate IPAM with accounts outside of your organization). If you are not creating an IPAM and integrating it with accounts outside your organization, you do not need to create, share, or associate resource discoveries.

If you are integrating an IPAM with accounts outside of your organizations, this is a required step that must be completed by the **Primary Org IPAM Account**. For more information about the roles involved in this process, see Process overview.

AWS Management Console

To associate a resource discovery

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **IPAMs**.
3. Select **Associated discoveries**, and then choose **Associate resource discoveries**.
4. Under **IPAM resource discoveries**, choose a resource discovery that's been shared with you by the **Secondary Org Admin Account**.
5. Choose **Associate**.

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.

- Associate a resource discovery: associate-ipam-resource-discovery

Disassociate a resource discovery

This section describes how to disassociate a resource discovery from an IPAM. When you disassociate a resource discovery from an IPAM, the IPAM no longer monitors all resources CIDRs and accounts discovered under the resource discovery.

> **Note**
>
> You cannot disassociate a default resource discovery association. A default resource discovery association is one that is created automatically when you create an IPAM. The default resource discovery association is deleted, however, if you delete the IPAM.

This step must be completed by the **Primary Org IPAM Account**. For more information about the roles involved in this process, see Process overview.

AWS Management Console

To disassociate a resource discovery

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **IPAMs**.
3. Select **Associated discoveries,** and then choose **Disassociate resource discoveries**.
4. Under **IPAM resource discoveries**, choose a resource discovery that's been shared with you by the **Secondary Org Admin Account**.
5. Choose **Disassociate**.

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 disassociate a resource discovery: disassociate-ipam-resource-discovery

Delete a resource discovery

This section describes how to delete a resource discovery.

> **Note**
>
> You cannot delete a default resource discovery. A default resource discovery is one that is created automatically when you create an IPAM. The default resource discovery is deleted, however, if you delete the IPAM.

This step must be completed by the **Secondary Org Admin Account**. For more information about the roles involved in this process, see Process overview.

AWS Management Console

To delete a resource discovery

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Resource discoveries**.
3. Select a resource discovery and choose **Actions** > **Delete resource discovery**.

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 delete a resource discovery: delete-ipam-resource-discovery

Tracking IP address usage in IPAM

Amazon VPC IP Address Manager offers IP address usage tracking features that can benefit anyone who manages complex network environments. IPAM provides visibility into IP address allocation, utilization, and consumption trends across AWS. This helps you identify unused or inefficiently used IP addresses, optimization of the address space and preventing potential IP address exhaustion.

IPAM tracks IP address usage at the CIDR, scope, and IPAM levels, providing detailed reporting and analytics. This is valuable for large-scale deployments, multi-account setups, and evolving network requirements.

By leveraging IPAM's usage tracking, you can make informed decisions, improve IP address management, and ensure efficient utilization of IP resources.

> **Note**
>
> 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.

Contents

- Monitor CIDR usage with the IPAM dashboard
- Monitor CIDR usage by resource
- Monitor IPAM with Amazon CloudWatch
- View IP address history
- View public IP insights

Monitor CIDR usage with the IPAM dashboard

The IPAM dashboard in Amazon VPC IP Address Manager allows you to monitor CIDR usage for several key scenarios:

- **Identify unused or underutilized IP address space**: The dashboard provides visibility into CIDR utilization, enabling you to identify CIDRs with available capacity that can be reclaimed or reallocated.
- **Optimize IP address management**: By closely tracking CIDR usage, you can make informed decisions about expanding, contracting, or reassigning IP address blocks to meet changing business and infrastructure requirements.
- **Prevent IP address exhaustion**: Monitoring CIDR usage helps you anticipate when you may need to acquire additional IP address space, allowing you to proactively plan and avoid service disruptions due to IP address depletion.
- **Ensure compliance and governance**: The IPAM dashboard can help you demonstrate IP address usage patterns to meet regulatory requirements or internal policies around IP address management.
- **Troubleshoot network issues**: Detailed CIDR usage data can assist in identifying the root causes of network connectivity problems or resource conflicts.

By closely monitoring CIDR usage through the IPAM dashboard, you can enhance the efficiency, resilience, and compliance of your IP address management within AWS.

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.
4. The dashboard presents an overview of your IPAM pools and CIDRs within a scope. You can add, remove, resize, and move widgets to customize the dashboard.

    - **Scope**: The details for this scope. A scope is the highest-level container within IPAM. An IPAM contains two default scopes, one private and one public. Each scope represents the IP space for a single network. You may have multiple private scopes, but you can only have one public scope.
        - **Scope ID**: The ID for this scope.

    - **Scope type**: The type of scope.
    - **IPAM ID**: The ID of the IPAM that the scope is in.
    - **IPAM pools in this scope**: The ID of the IPAM that the scope is in.
    - **View networking resources in this scope**: Takes you to the **Resources** section of the IPAM console.
    - **Search the history of an IP address in this scope**: Takes you to the **Search IP history** section of the IPAM console.
- **Resource CIDR types**: The types of resource CIDRs in the scope.
    - **Subnet**: The number of CIDRs for subnets.
    - **VPC**: The number of CIDRs for VPCs.
    - **EIPs**: The number of CIDRs for Elastic IP addresses.
    - **Public IPv4 pools**: The number of CIDRs for public IPv4 pools.
- **Management state**: The management state of the CIDRs.
    - **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).
    - **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.
- **Overlapping resource CIDRs**: The number of overlapping and nonoverlapping CIDRs. Overlapping CIDRs can lead to incorrect routing in your VPCs.
    - **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.
    - **Nonoverlapping CIDRs**: The number of resource CIDRs that do not overlap within the IPAM pools in this scope.
- **Compliant resource CIDRs**: The number of compliant resource CIDRs.
    - **Compliant CIDRs**: The number of resource CIDRs that comply with the allocation rules for IPAM pools 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.
- **Overlap status**: The number of CIDRs that overlap over time.
    - **OverlappingResourceCidrs**: The number of CIDRs that overlap within the IPAM pools in this scope. Overlapping CIDRs can lead to incorrect routing in your VPCs.
- **Compliance status**: The number of CIDRs that comply versus do not comply with the allocation rules for IPAM pools in the scope over time.
    - **CompliantResourceCidrs**: The number of resource CIDRs that comply with the allocation rules.
    - **NoncompliantResourceCidrs**: The number of resource CIDRs that do not comply with the allocation rules.
- **VPC utilization**: VPCs (IPv4 and IPv6) with the highest or lowest IP utilization. You can use this information to configure Amazon CloudWatch alarms to be alerted if an IP utilization threshold is breached. For more information, see IPAM resource utilization metrics.
- **Subnet utilization**: Subnets (IPv4 only) with the highest or lowest IP utilization. You can use this information to decide if you want to keep or delete resources that are underutilized. For more information, see IPAM resource utilization metrics.
- **VPCs with highest IPs allocated**: The VPCs that have the highest percentage of IP address space allocated to subnets. This is useful to show you if you need to provision additional IP address space to the VPCs.
- **Subnets with highest IPs allocated**: The subnets that have the highest percentage of IP address space allocated to resources. This is useful to show you if you need to provision additional IP address space to the subnets.
- **Pool assignment**: The percentage of IP space that has been assigned to resources and manual allocations in the scope over time.
- **Pool allocation**: The percentage of a pool's IP space that has been allocated to other pools in the scope over time.

Command line

The information displayed in the dashboard comes from metrics stored in Amazon CloudWatch. For more information about the metrics stored in Amazon CloudWatch, see Monitor IPAM with

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.

Monitor CIDR usage by resource

The **Resources** view in Amazon VPC IP Address Manager provides a centralized overview of IP address utilization across your AWS resources. This enables you to quickly identify which resources are consuming IP addresses, track address allocation trends, and optimize your IP address management to align with your evolving infrastructure and business needs.

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, so it's important to understand the difference between the two:

- **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:
  - Elastic IP addresses
  - Public IPv4 pools

    > **Note**
    >
    > 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.

  - VPCs
- **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:
  - Elastic IP addresses

- Public IPv4 pools
- VPCs
- VPC subnets

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 IP dropdown menu at the top of the content pane, choose the IP address protocol that you want to use: IPv4 or IPv6.
4. From the scope 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.
5. Use the resource CIDR map to view available, allocated, and overlapping IP address space in a scope:

   - **Available**: An IP address range is available for allocation.
   - **Compliant and nonoverlapping**: An IP address range is allocated to a resource managed by IPAM.
   - **Occupied**: An IP address range is allocated to a resource.
   - **Overlapping**: An IP address range has been allocated to multiple resources and is overlapping.
   - **Noncompliant**: An IP address range is not compliant. There is a resource using the IP address range that is not compliant with the allocation rules set up for the pool.

   In the CIDR map, choose an IP address block at the bottom of the map to view the resources in smaller CIDR blocks. Choose an IP address block at the top of the map to view the resources in larger CIDR blocks.
6. In the table, you can view the following details about resources in the scope:

   - **Name (Resource ID)**: The name and resource ID of the resource.
   - **CIDR**: The CIDR associated with the resource.
   - **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 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 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 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 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 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 manage.

- **IPs allocated**: For resources that are VPCs, this is the percentage of IP address space in the VPC that's taken up by subnet CIDRs. For resources that are subnets, if the subnet has an IPv4 CIDR provisioned to it, this is the percentage of IPv4 address space in the subnet that's in use. If the subnet has an IPv6 CIDR provisioned to it, the percentage of IPv6 address space in use is not represented. The percentage of IPv6 address space in use cannot currently be calculated. For resources that are public IPv4 pools, this is the percentage of IP address space in the pool that's been allocated to Elastic IP addresses (EIPs).
- **Region**: The AWS Region of the resource.
- **Owner ID**: The AWS account ID of the person that created this resource.
- **Resource type**: Whether the resource is a VPC, subnet, Elastic IP address, or public IPv4 pool.
- **Pool ID**: The ID of the IPAM pool that the resource is in.

7. Use **Filter resources** to filter the resources table by column property, like VPC ID or compliance status.

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 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) and resource utilization in the `AWS/IPAM` Amazon CloudWatch namespace in the home Region of your IPAM.

Integrating IPAM with CloudWatch enhances your ability to monitor, analyze, and optimize your IP address management within AWS.

Use cases include:

- **Tracking IP address utilization trends**: CloudWatch can monitor CIDR pool usage, scope allocation, and other IPAM metrics, helping you proactively identify potential IP address exhaustion risks.
- **Setting utilization-based alerts**: You can configure CloudWatch alarms to notify you when CIDR utilization reaches predetermined thresholds, enabling timely intervention and optimization.
- **Monitoring IPAM events**: CloudWatch can capture and analyze IPAM-related events, such as CIDR allocations, deallocations, and scope modifications, providing visibility into IP address management activities.
- **Generating custom dashboards**: By combining IPAM data with other AWS metrics, you can create comprehensive dashboards to visualize and analyze your IP address landscape alongside related infrastructure and performance indicators.

Contents

- IPAM metrics
- IPAM resource utilization metrics

IPAM metrics

IPAM publishes data about your IPAM, pools, and scopes to Amazon CloudWatch. 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 with Amazon CloudWatch is outside the scope of this section. 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 metrics

The AWS/IPAM namespace includes the following IPAM metrics.

Metric name	Description
TotalActiveIpCount	The total active IP count is the number of active IP addresses in your IPAM that you would be charged if you switched from the Free Tier to the Advanced Tier. An active IP address is defined as an IP address or a prefix associated with an Elastic Network Interface (ENI) that is attached to a resource such as an EC2 Instance. • This metric is only available to customers in the Free Tier. • If your IPAM is integrated with AWS Organizations, the active IP count covers all the Organization accounts. • You cannot view a breakdown of the active IP count by IP type (public/private) or class (IPv4/IPv6). • IPAM only counts IPs from ENIs owned by monitored accounts. The count may be inaccurate for shared subnets. IP addresses are excluded if the subnet owner or ENI owner is not covered by IPAM.

IPAM pool metrics

The AWS/IPAM namespace includes the following pool metrics for IPAM.

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 IPv4 pool.
NoncompliantResourceCidrs	The number of managed resource CIDRs that do not comply with the allocation rules of the IPAM pool. For more informati on about allocation rules, see Create a top-level IPv4 pool.

Metric name	Description
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

The `AWS/IPAM` namespace includes the following scope metrics for IPAM.

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

IPAM public IP metrics

The `AWS/IPAM` namespace includes the following public IP metrics for IPAM.

Metric name	Description
AmazonOwnedContigIPs	The number of IP addresses within CIDRs that are provisioned to Amazon-provided contiguous public IPv4 pools owned by the IPAM.
AllocatedAmazonOwnedContigIPs	The number of IP addresses that have been allocated from an Amazon-provided contiguous public IPv4 pool CIDR block.
UnallocatedAmazonOwnedContigIPs	The number of IP addresses within the Amazon-provided contiguous public IPv4 pool CIDR block owned by the IPAM.
AssociatedAmazonOwnedContigIPs	The number of Elastic IP addresses that have been allocated from an Amazon-provided contiguous public IPv4 pool CIDR block that are associated with an elastic network interface.
UnassociatedAmazonOwnedContigIPs	The number of Elastic IP addresses that have been allocated from an Amazon-provided contiguous public IPv4 pool CIDR block that are not associated with an elastic network interface.

Metric dimensions

To filter the IPAM metrics, use the following dimensions.

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.

For information about monitoring VPCs with Amazon CloudWatch, see CloudWatch metrics for your VPCs in the *Amazon Virtual Private Cloud User Guide*.

IPAM resource utilization metrics

IPAM publishes IP utilization metrics for resources that the IPAM monitors to Amazon CloudWatch. These resources include:

- VPCs (IPv4 and IPv6)
- Subnets (IPv4)
- Public IPv4 pools

IPAM calculates and publishes IP utilization metrics separately by IP address family (IPv4 or IPv6). The IP utilization of a resource is calculated across all of its CIDRs of the same address family.

For each resource type and address family combination, IPAM uses three rules to determine which metrics to publish:

- Up to 50 resources with the highest IP utilization. You can use this information to configure alarms to be alerted if an IP utilization threshold is breached.
- Up to 50 resources with the lowest IP utilization. You can use this information to decide if you want to keep or delete resources that are underutilized.
- Up to 50 other resources. You can use this information to consistently track the IP utilization of resources that may not be captured within the high or low utilization group.
  - Up to 50 VPCs containing a CIDR allocated from an IPAM pool (prioritized by total size of CIDR blocks).
  - Up to 50 subnets whose VPC contains a CIDR allocated from an IPAM pool (prioritized by total size of CIDR blocks).
  - Up to 50 public IPv4 pools containing a CIDR allocated from an IPAM pool (prioritized by total size of CIDR blocks).

After applying each rule, the metrics are aggregated and published under the same metric name for each resource type. See below for detailed information on the metric names and their dimensions.

Important

There is a unique limit for each resource type, address family, and rule combination. The default value of each limit is 50. You can adjust these limits by contacting the AWS Support Center as described in AWS service quotas in the *AWS General Reference*.

Example Example

Let's say that your IPAM monitors 2,500 VPCs and 10,000 subnets, all with IPv4 and IPv6 CIDRs. IPAM publishes the following IP utilization metrics:

- Up to 150 metrics for VPC IPv4 IP utilization, including:
  - The 50 VPCs with the highest IPv4 IP utilization
  - The 50 VPCs with the lowest IPv4 utilization
  - Up to 50 VPCs containing an IPv4 CIDR allocated from an IPAM pool
- Up to 150 metrics for VPC IPv6 utilization, including:
  - The 50 VPCs with the highest IPv6 IP utilization
  - The 50 VPCs with the lowest IPv6 utilization
  - Up to 50 VPCs containing an IPv6 CIDR allocated from an IPAM pool
- Up to 150 metrics for subnet IPv4 utilization, including:
  - The 50 subnets with the highest IPv4 IP utilization
  - The 50 subnets with the lowest IPv4 IP utilization
  - Up to 50 subnets whose VPC contains an IPv4 CIDR allocated from an IPAM pool

VPC metrics

The VPC metric name and description is listed below.

Metric name	Description
VpcIPUsage	The total IPs covered by CIDRs in the VPC's subnets divided by the total IPs covered by CIDRs in the VPC. This is calculated across all VPC CIDRs in the same IPAM Scope and separately for IPv4 and IPv6 CIDRs.

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

Dimension	Description
AddressFamily	The IP address family for resource CIDRs (IPv4 or IPv6).
OwnerID	The ID of the VPC owner.
Region	The AWS Region where the VPC is located.
ScopeID	The ID of the IPAM scope that the VPC belongs to.
VpcID	The ID of the VPC.

Subnet metrics

The subnet metric name and description is listed below.

Metric name	Description
SubnetIPUsage	The number of active IPs divided by total IPs in the subnet's IPv4 CIDR.

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

Dimension	Description
AddressFamily	The IP address family for resource CIDRs (IPv4 only).
OwnerID	The ID of the subnet owner.
Region	The AWS Region where the subnet is located.
ScopeID	The ID of the IPAM scope that the subnet belongs to.
SubnetID	The ID of the subnet.
VpcID	The ID of the VPC that the subnet belongs to.

Public IPv4 pool metrics

The public IPv4 pool metric name and description is listed below.

Metric name	Description
PublicIPv4PoolIPUsage	The number of EIPs from the public IPv4 Pool divided by total IPs in the pool.

The dimensions you can use to filter the public IPv4 pool metrics are listed below.

Dimension	Description
OwnerID	The ID of the public IPv4 pool owner.
PublicIPv4PoolID	The ID of the public IPv4 pool.
Region	The AWS Region where the public IPv4 pool is located.
ScopeID	The ID of the IPAM scope that the public IPv4 pool belongs to.

Public IP insight metrics

The public IP insight metric names and descriptions are listed below.

Metric name	Description
AmazonOwnedElasticIPs	The number of Amazon-owned Elastic IP addresses that you have provisioned or assigned to resources in your AWS account.
AssociatedAmazonOwnedElasticIPs	The number of Amazon-owned Elastic IP addresses that you have associated with resources in your AWS account.
AssociatedBringYourOwnIPs	The number of public IPv4 addresses that you have brought to AWS using Bring your own IP addresses (BYOIP) and have associated with resources in your AWS account.

Metric name	Description
BringYourOwnIPs	The number of public IPv4 addresses that you have brought to AWS using Bring your own IP addresses (BYOIP).
EC2PublicIPs	The number of public IPv4 addresses assigned to EC2 instances when the instances were launched into a default subnet or into a subnet configured to automatically assign a public IPv4 address.
ServiceManagedBringYourOwnIPs	The number of public IPv4 addresses that you have brought to AWS using Bring your own IP addresses (BYOIP) that are provisioned and managed by an AWS service.
ServiceManagedIPs	The number of public IPv4 addresses provisioned and managed by an AWS service.
UnassociatedAmazonOwnedElasticIPs	The number of Amazon-owned Elastic IP addresses that you have not associated with resources in your AWS account.
UnassociatedBringYourOwnIPs	The number of public IPv4 addresses that you have brought to AWS using Bring your own IP addresses (BYOIP) and have not associated with any resources in your AWS account.

The dimensions you can use to filter the public IP insight metrics are listed below.

Dimension	Description
IpamId	The ID of the IPAM that the IP address belongs to.
Region	The AWS Region where the public IP address is located.

Quick tip for creating alarms

To quickly create an Amazon CloudWatch alarm for resources with high IP address utilization, open the CloudWatch console, choose **Metrics**, **All metrics**, choose the **Query** tab, choose the **Namespace** `AWS/IPAM > VPC IP Usage Metrics`, `AWS/IPAM > Subnet IP Usage`

`Metrics`, or `AWS/IPAM > Public IPv4 Pool IP Usage Metrics`, choose the **Metric name** `MAX(VpcIPUsage)`, `MAX(SubnetIPUsage)`, or `MAX(PublicIPv4PoolIPUsage)`, and choose **Create alarm**. For more information, see Create alarms on Metrics Insights queries in the *Amazon CloudWatch User Guide*.

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 that are attached to instances, you can use the Search IP history feature to search for historical data on EC2 instance and network interface CIDRs.

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. For more information, see Move VPC CIDRs between scopes.
- If you delete or transfer a resource to an AWS account that's not monitored by your IPAM, any new history related to the resource will not be visible and your IPAM won't monitor the resource. The IP address of the resource, however, will still be searchable.

- If you Integrate IPAM with accounts outside of your organization, the IPAM owner can view the IP address history of all resource CIDRs owned by those accounts.

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 **Search IP history**.
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.
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 Search IP history 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 Search IP history, 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 that this resource belongs to (if applicable).
- **Region**: The AWS Region of this resource.
- **Owner ID**: The AWS account ID of the user that created this resource (if applicable).

View public IP insights

You can use **Public IP insights** to see the following:

- If your IPAM is integrated with accounts in an AWS Organization, you can view all public IPv4 addresses used by services across all AWS Regions for your entire AWS Organization.
- If your IPAM is integrated with a single account, you can view all public IPv4 addresses used by services across all AWS Regions in your account.

A public IPv4 address is an IPv4 address that is routable from the internet. A public IPv4 address is necessary for a resource to be directly reachable from the internet over IPv4.

> **Note**
>
> AWS charges for all public IPv4 addresses, including public IPv4 addresses associated with running instances and Elastic IP addresses. For more information, see the **Public IPv4 Address** tab on the Amazon VPC pricing page.

You can view insights into the following public IPv4 address types:

- **Elastic IP addresses (EIPs)**: Static, public IPv4 addresses provided by Amazon that you can associate with an EC2 instance, elastic network interface, or AWS resource.
- **EC2 public IPv4 addresses**: Public IPv4 addresses assigned to an EC2 instance by Amazon (if the EC2 instance is launched into a default subnet or if the instance is launched into a subnet that's been configured to automatically assign a public IPv4 address).
- **BYOIPv4 addresses**: Public IPv4 addresses in the IPv4 address range that you've brought to AWS using Bring your own IP addresses (BYOIP).
- **Service-managed IPv4 addresses**: Public IPv4 addresses automatically provisioned on AWS resources and managed by an AWS service. For example, public IPv4 addresses on Amazon ECS, Amazon RDS, or Amazon WorkSpaces.

Public IP insights shows you all public IPv4 addresses used by services across Regions. You can use these insights to identify public IPv4 address usage and view recommendations to release unused Elastic IP addresses.

- **Public IP types**: The number of public IPv4 addresses organized by type.
  - **Amazon-owned EIPs**: Elastic IP addresses that you have provisioned or assigned to resources in your AWS account.
  - **EC2 public IPs**: Public IPv4 addresses assigned to EC2 instances when the instances were launched into a default subnet or into a subnet that's been configured to automatically assign a public IPv4 address.
  - **BYOIP**: Public IPv4 addresses that you have brought to AWS using Bring your own IP addresses (BYOIP).
  - **Service managed IPs**: Public IPv4 addresses provisioned and managed by an AWS service.
  - **Service managed BYOIP**: Public IPv4 addresses brought to AWS and managed by an AWS service.

  - **Amazon-owned contiguous EIPs**: Elastic IP addresses allocated from an Amazon-provided contiguous public IPv4 IPAM pool.
- **EIP usage**: The number of Elastic IP addresses organized by how they are used.
  - **Associated Amazon-owned EIPs**: Elastic IP addresses that you have provisioned in your AWS account and that you have associated with an EC2 instance, network interface, or AWS resource.
  - **Associated BYOIP**: Public IPv4 addresses you have brought to AWS using BYOIP that you have associated with a network interface.
  - **Unassociated Amazon-owned EIPs**: Elastic IP addresses that you have provisioned in your AWS account but you have not associated with a network interface.
  - **Unassociated BYOIP**: Public IPv4 addresses you have brought to AWS using BYOIP but you have not associated with a network interface.
  - **Associated Amazon-owned contiguous EIPs**: Elastic IP addresses allocated from an Amazon-provided contiguous public IPv4 IPAM pool and associated with a resource.
  - **Unassociated Amazon-owned contiguous EIPs**: Elastic IP addresses allocated from an Amazon-provided contiguous public IPv4 IPAM pool and unassociated with a resource.
- **Amazon-owned IPv4 contiguous IPs usage**: A table that shows contiguous public IPv4 address usage over time and related Amazon-owned IPv4 IPAM pools.
- **Public IP addresses**: A table of public IPv4 addresses and their attributes.
  - **IP address**: The public IPv4 address.
  - **Associated**: Whether or not the address is associated with an EC2 instance, network interface, or AWS resource.
    - **Associated**: The public IPv4 address is associated with an EC2 instance, network interface, or AWS resource.
    - **Unassociated**: The public IPv4 address is not associated to any resource and is idle in your AWS account.
  - **Address type**: The IP address type.
    - **Amazon-owned EIP**: The public IPv4 address is an Elastic IP address.
    - **BYOIP**: The public IPv4 address was brought to AWS using BYOIP.
    - **EC2 public IP**: The public IPv4 address was assigned automatically to an EC2 instance.
    - **Service managed BYOIP**: The public IPv4 address was brought to AWS using Bring your own IP (BYOIP).

  - **Service managed IP**: The public IPv4 address was provisioned and is managed by an AWS service.
- **Service**: The service that the IP address is associated with.
  - **AGA**: An AWS Global Accelerator. If a custom routing accelerator is used, its public IPs are not listed. To view these public IPs, see Viewing your custom routing accelerators.
  - **Database Migration Service**: An AWS Database Migration Service (DMS) replication instance.
  - **Redshift**: An Amazon Redshift cluster.
  - **RDS**: An Amazon Relational Database Service (RDS) instance.
  - **Load balancer (EC2)**: An Application Load Balancer or a Network Load Balancer.
  - **NAT gateway (VPC)**: An Amazon VPC public NAT gateway.
  - **Site-to-Site VPN**: An AWS Site-to-Site VPN virtual private gateway.
  - **Other**: Other service that is not currently identifiable.
- **Name (EIP ID)**: If this public IPv4 address is an Elastic IP address allocation, this is the name and ID of the EIP allocation.
- **Network interface ID**: If this public IPv4 address is associated with a network interface, this is the ID of the network interface.
- **Instance ID**: If this public IPv4 address is associated with an EC2 instance, this is the instance ID.
- **Security groups**: If this public IPv4 address is associated with an EC2 instance, this is the name and ID of the security group assigned to the instance.
- **Public IPv4 pool**: If this is an Elastic IP address from an IP address pool owned and managed by Amazon, the value is "-". If this is an Elastic IP address from an IP address range which you own and have brought to Amazon (using BYOIP), the value is the public IPv4 pool ID.
- **Network border group**: If the IP address is advertised, this is the AWS Region from which the IP address is advertised.
- **Owner ID**:The AWS account number of resource owner.
- **Sample time**: The last successful resource discovery time.
- **Resource discovery ID**: ID of the resource discovery that has discovered this public IPv4 address.
- **Service resource**: Resource ARN or ID.

If an Elastic IP address is allocated to your account but is not associated with a network interface, a banner appears informing you that you have unassociated EIPs in your account and you should release them.

> **Important**
>
> Public IP insights was recently updated. If you see an error related to not having permissions to call GetIpamDiscoveredPublicAddresses, the managed permission attached to a resource discovery that was shared with you needs to be updated. Contact the person who created the resource discovery and ask them to update the managed permission `AWSRAMPermissionIpamResourceDiscovery` to the default version. For more information, see Update a resource share in the *AWS RAM User Guide* .

AWS Management Console

To view public IP address insights

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Public IP insights**.
3. To view details for a public IP address, select an IP address by clicking on it.
4. View the following information about the IP address:

   - **Details**: The same information visible in the columns of the main Public IP insights pane, such as the **Address type** and **Service**.
   - **Inbound security group rules**: If this IP address is associated with an EC2 instance, these are the security group rules that control the inbound traffic to the instance.
   - **Outbound security group rules**: If this IP address is associated with an EC2 instance, these are the security group rules that control the outbound traffic from the instance.
   - **Tags**: Key and value pairs that act as metadata for organizing your AWS resources.

Command line

Use the following command to get the public IP addresses that have been discovered by IPAM: get-ipam-discovered-public-addresses

Tutorials for Amazon VPC IP Address Manager

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

Contents

- Tutorial: Create an IPAM and pools using the console
- Tutorial: Create an IPAM and pools using the AWS CLI
- Tutorial: View IP address history using the AWS CLI
- Tutorial: Bring your ASN to IPAM
- Tutorial: Bring your IP addresses to IPAM
- Tutorial: Transfer a BYOIP IPv4 CIDR to IPAM
- Tutorial: Plan VPC IP address space for subnet IP allocations
- Allocate sequential Elastic IP addresses from an IPAM pool

Tutorial: Create an IPAM and pools using the console

In this tutorial, you create an IPAM, integrate with AWS Organizations, create IP address pools, and create a VPC with a CIDR from an IPAM pool.

This tutorial shows you how you can use IPAM to organize IP address space based on different development needs. Once you've completed this tutorial, you'll have one IP address pool for pre-production resources. You can then create other pools based on your routing and security needs, such as a pool for production resources.

While you can use IPAM as a single user, integrating with AWS Organizations enables you to manage IP addresses across accounts in your organization. This tutorial covers integrating IPAM with accounts in an organization. It does not cover how to Integrate IPAM with accounts outside of your organization.

> **Note**
>
> For the purposes of this tutorial, the instructions will tell you to name IPAM resources in a particular way, create IPAM resources in specific Regions, and use specific IP address CIDR

ranges for your pools. This is intended to streamline the choices available in IPAM and get you started with IPAM quickly. Once you've completed this tutorial, you may decide to create a new IPAM and configure it differently.

Contents

- Prerequisites
- How AWS Organizations integrates with IPAM
- Step 1: Delegate an IPAM administrator
- Step 2: Create an IPAM
- Step 3: Create a top-level IPAM pool
- Step 4: Create Regional IPAM pools
- Step 5: Create a pre-production development pool
- Step 6: Share the IPAM pool
- Step 7: Create a VPC with a CIDR allocated from an IPAM pool
- Step 8: Cleanup

Prerequisites

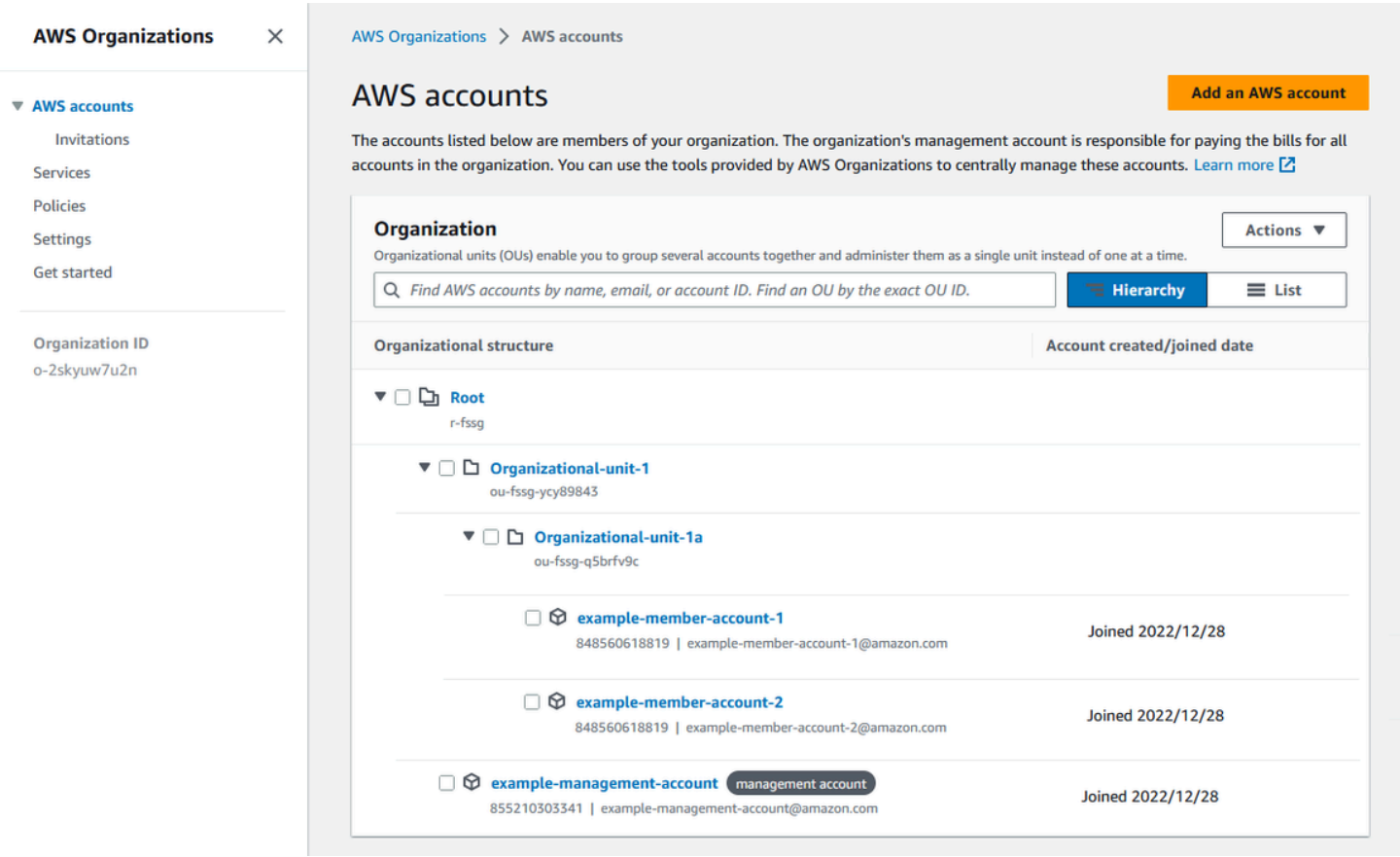
Before you begin, you must have set up an AWS Organizations account with at least one member account. For how-to instructions, see Creating and managing an organization in the *AWS Organizations User Guide*.

How AWS Organizations integrates with IPAM

This section shows an example of the AWS Organizations accounts you use in this tutorial. There are three accounts in your organization that you use when you integrate with IPAM in this tutorial:

- The management account (called **example-management-account** in the following image) to log into the IPAM console and delegate an IPAM admin. You cannot use the organization's management account as your IPAM admin.
- A member account (called *example-member-account-1* in the following image) as the IPAM admin account. The IPAM admin account is responsible for creating an IPAM and using it to manage and monitor IP address usage across the organization. Any member account in your organization can be delegated as the IPAM admin.

- A member account (called *example-member-account-2* in the following above) as the developer account. This account creates a VPC with a CIDR allocated from an IPAM pool.



In addition to the accounts, you'll need the ID of the organizational unit (**ou-fssg-q5brfv9c** in the preceding image) that contains the member account you'll use as the developer account. You need this ID so that, in a later step, when you share your IPAM pool, you can share it with this OU.

> **Note**
>
> For more information about AWS Organizations account types like *management* and *member* accounts, see AWS Organizations terminology and concepts.

Step 1: Delegate an IPAM administrator

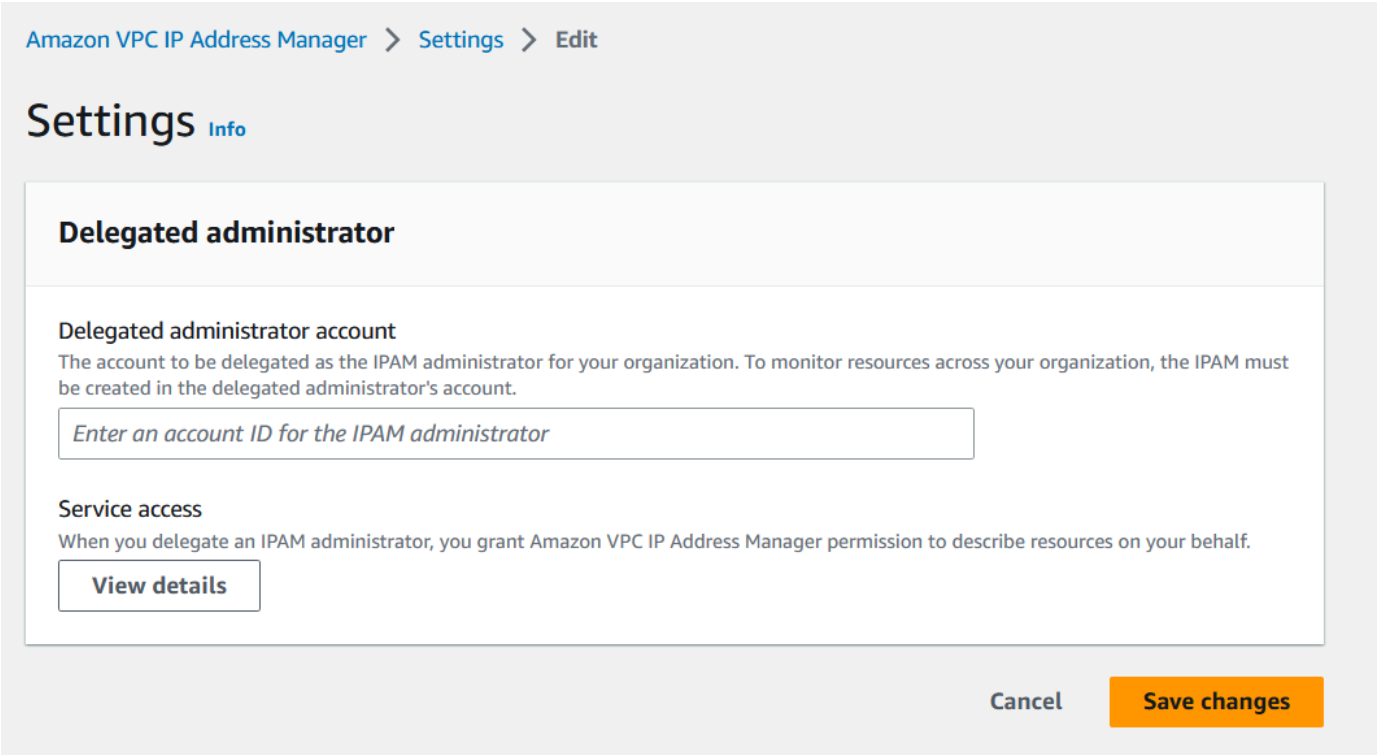
In this step, you'll delegate an AWS Organizations member account as the IPAM admin. When you delegate an IPAM admin, a service-linked role is automatically created in each of your AWS Organizations member accounts. IPAM monitors the IP address usage in these accounts by

assuming the service-linked role in each member account. It can then discover the resources and their CIDRs regardless of their Organizational Unit.

You cannot complete this step unless you have the required AWS Identity and Access Management (IAM) permissions. For more information, see Integrate IPAM with accounts in an AWS Organization.

To delegate an IPAM admin account

1. Using the AWS Organizations management account, 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 **Organization settings**.
4. Choose **Delegate**. The **Delegate** option is available only if you logged in to the console as the AWS Organizations management account.
5. Enter the AWS account ID for an organization member account. The IPAM administrator must be an AWS Organizations member account, not the management account.



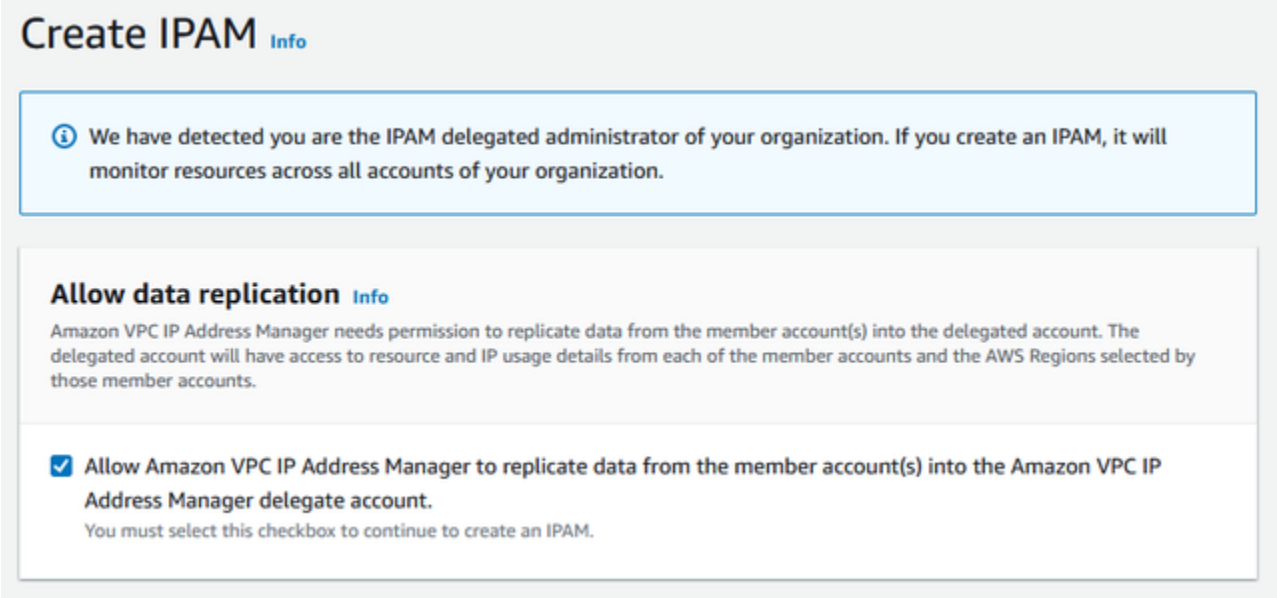
6. Choose **Save changes**. The **Delegated administrator** information is populated with details related to the member account.

Step 2: Create an IPAM

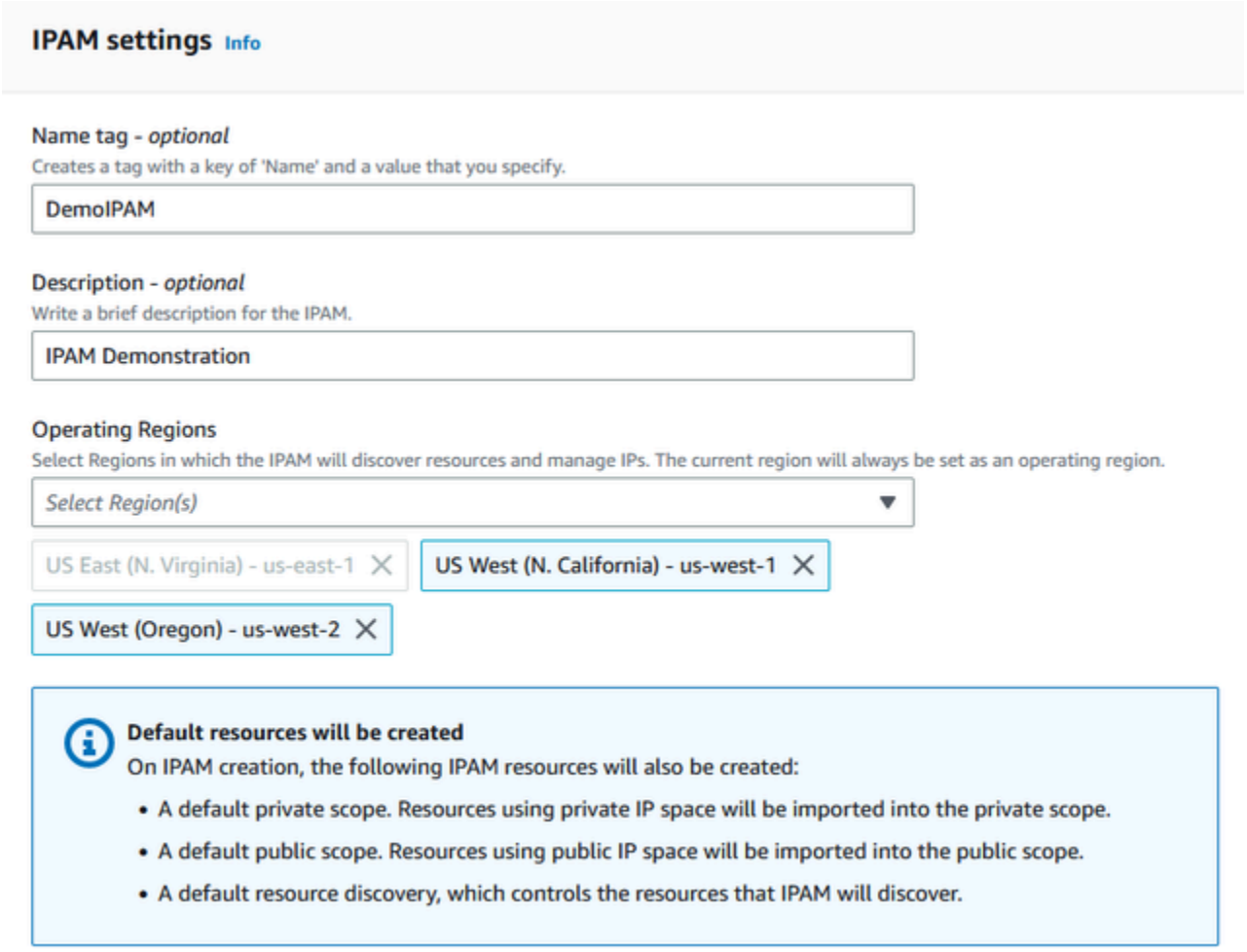
In this step you'll create an IPAM. When you create an IPAM, IPAM automatically creates two scopes for the IPAM: the private scope that's intended for all private space, and the public scope that's intended for all public space. The scopes, together with pools and allocations, are key components of your IPAM. For more information, see How IPAM works.

To create an IPAM

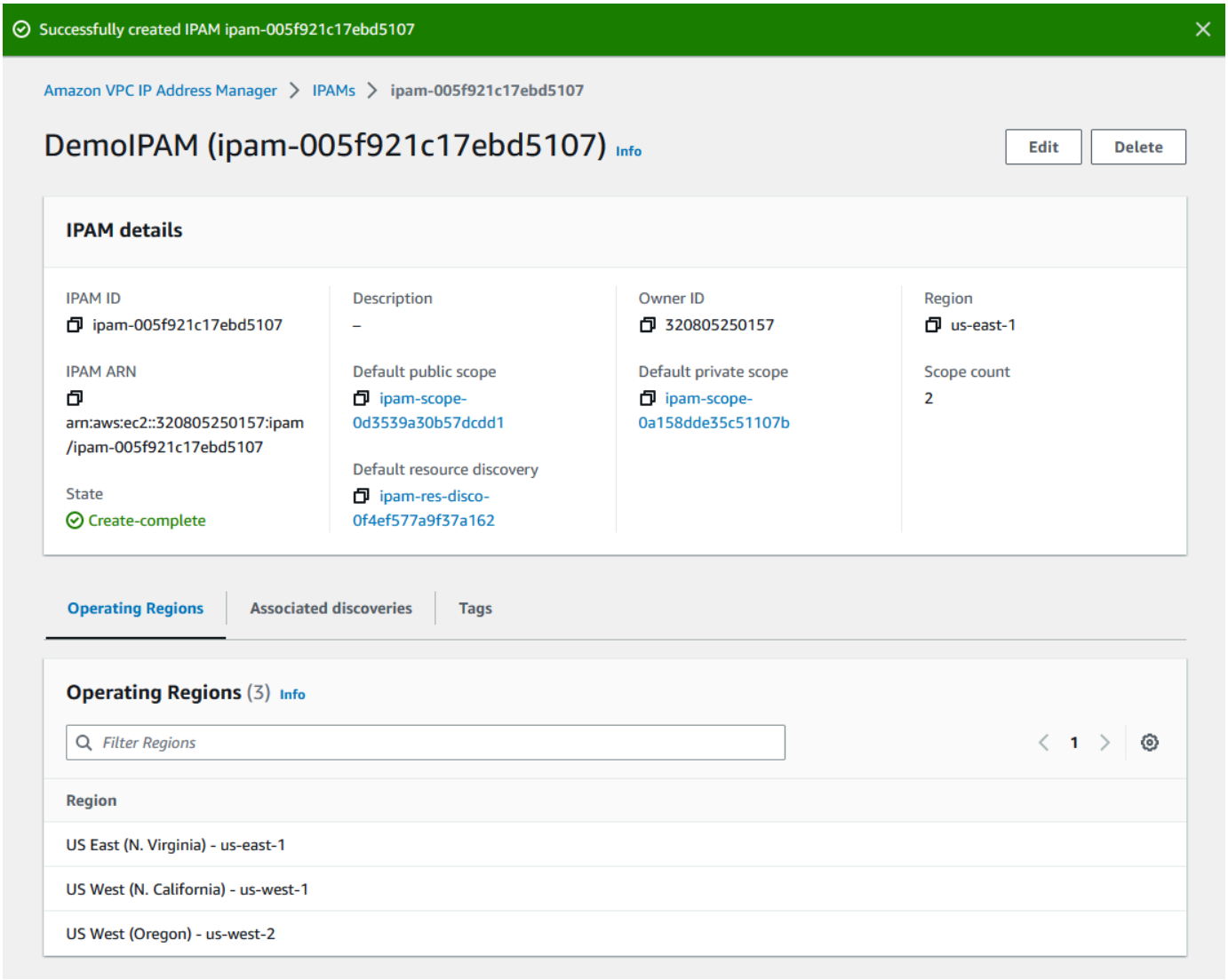
1. Using the AWS Organizations member account delegated as the IPAM admin in the previous step, open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the AWS Management Console, choose the AWS Region in which you want to create the IPAM. Create the IPAM in your main Region of operations.
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**, choose the AWS Regions in which this IPAM can manage and discover resources. The AWS Region in which you are creating your IPAM is automatically selected as one of the operating Regions. In this tutorial, the home Region of our IPAM is us-east-1, so we'll choose us-west-1 and us-west-2 as additional operating Regions. If you forget an operating Region, you can edit your IPAM settings later and add or remove Regions.



6. Choose **Create IPAM**.



Step 3: Create a top-level IPAM pool

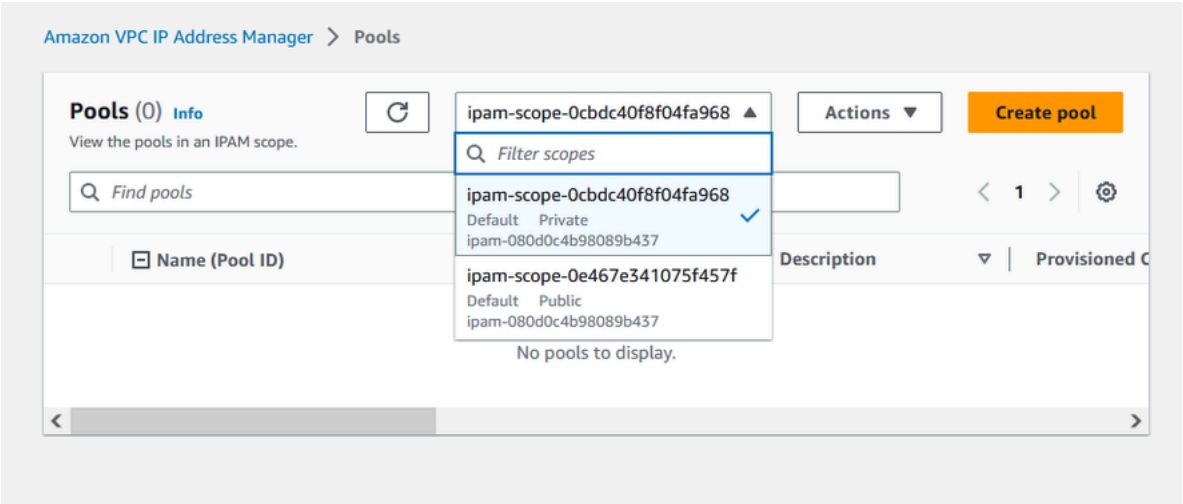
In this tutorial, you create a hierarchy of pools starting with the top-level IPAM pool. In the subsequent steps, you'll create a pair of Regional pools and a pre-production development pool in one of the regional pools.

For more information about pool hierarchies that you can build with IPAM, see Example IPAM pool plans.

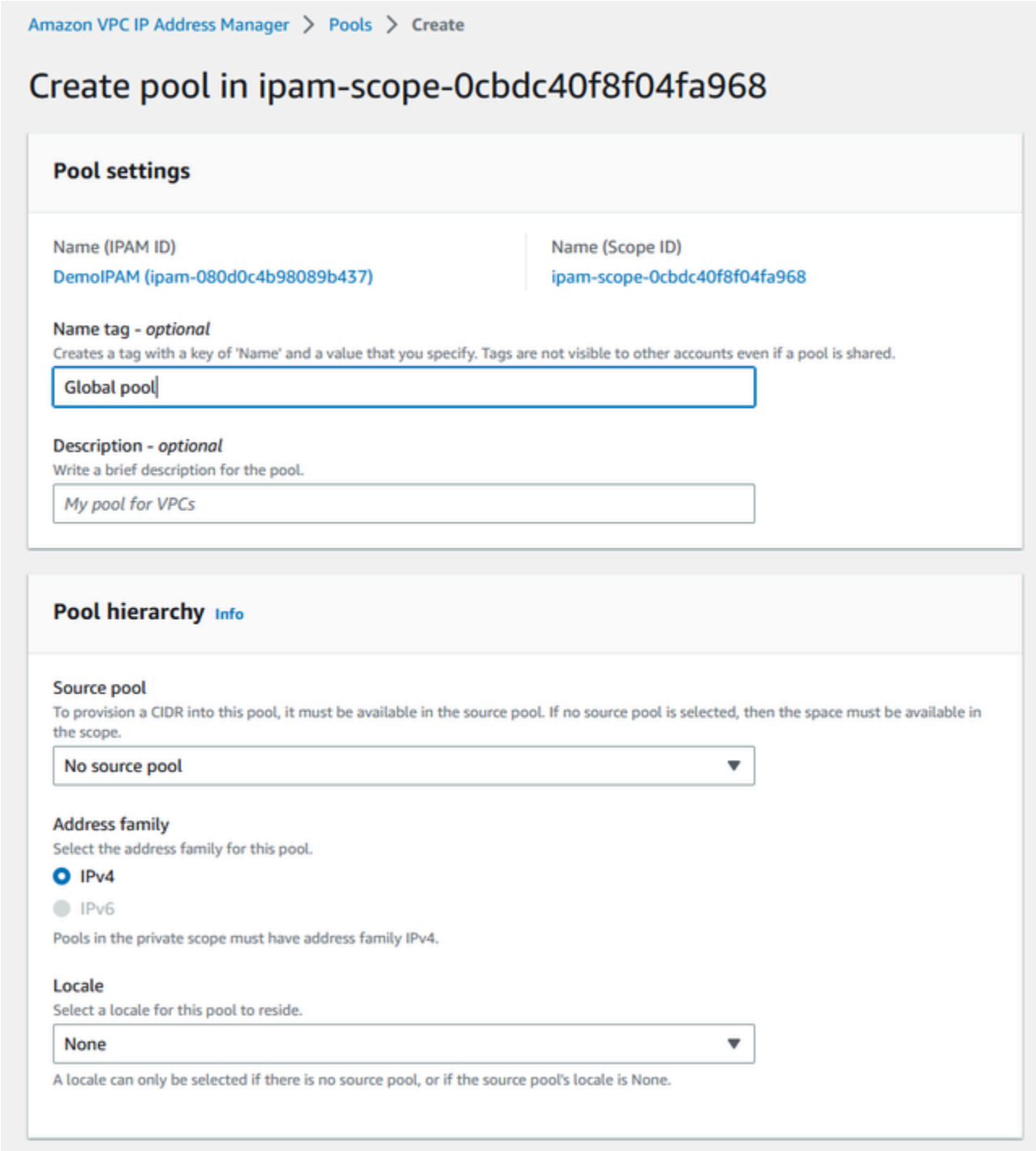
To create a top-level pool

1. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.

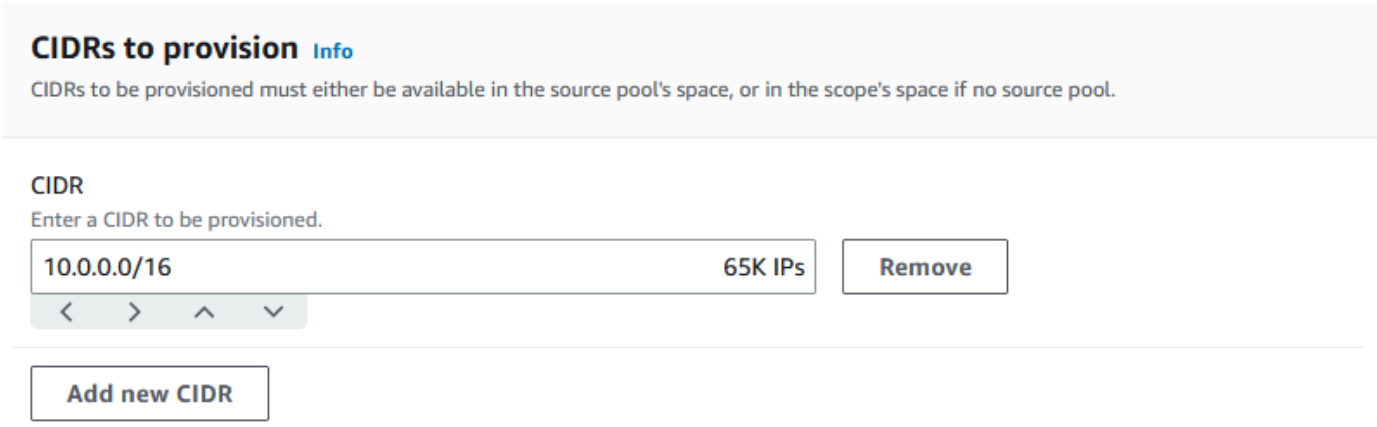
2. In the navigation pane, choose **Pools**.
3. Choose the private scope.



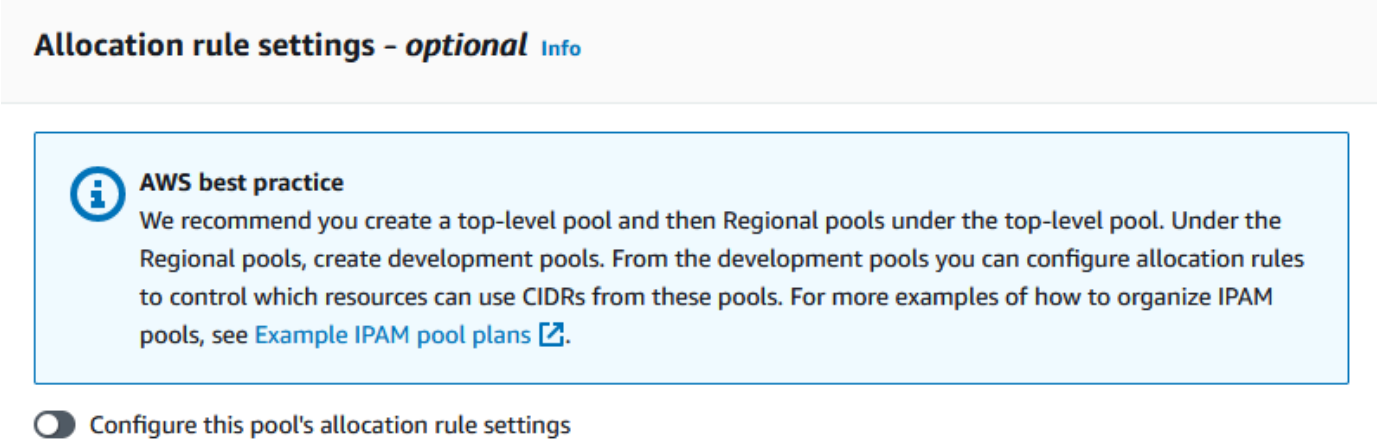
4. Choose **Create pool**.
5. Under **IPAM scope**, leave the private scope selected.
6. (Optional) Add a **Name tag** for the pool and a description for the pool, such as “Global pool”.
7. Under **Source**, choose **IPAM scope**. Because this is our top level pool, it will not have a source pool.
8. Under **Address family**, choose **IPv4**.
9. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.
10. For the **Locale**, choose **None**. Locales are the AWS Regions where you want this IPAM pool to be available for allocations. You’ll set the locale for the Regional pools that you create in the next section of this tutorial.



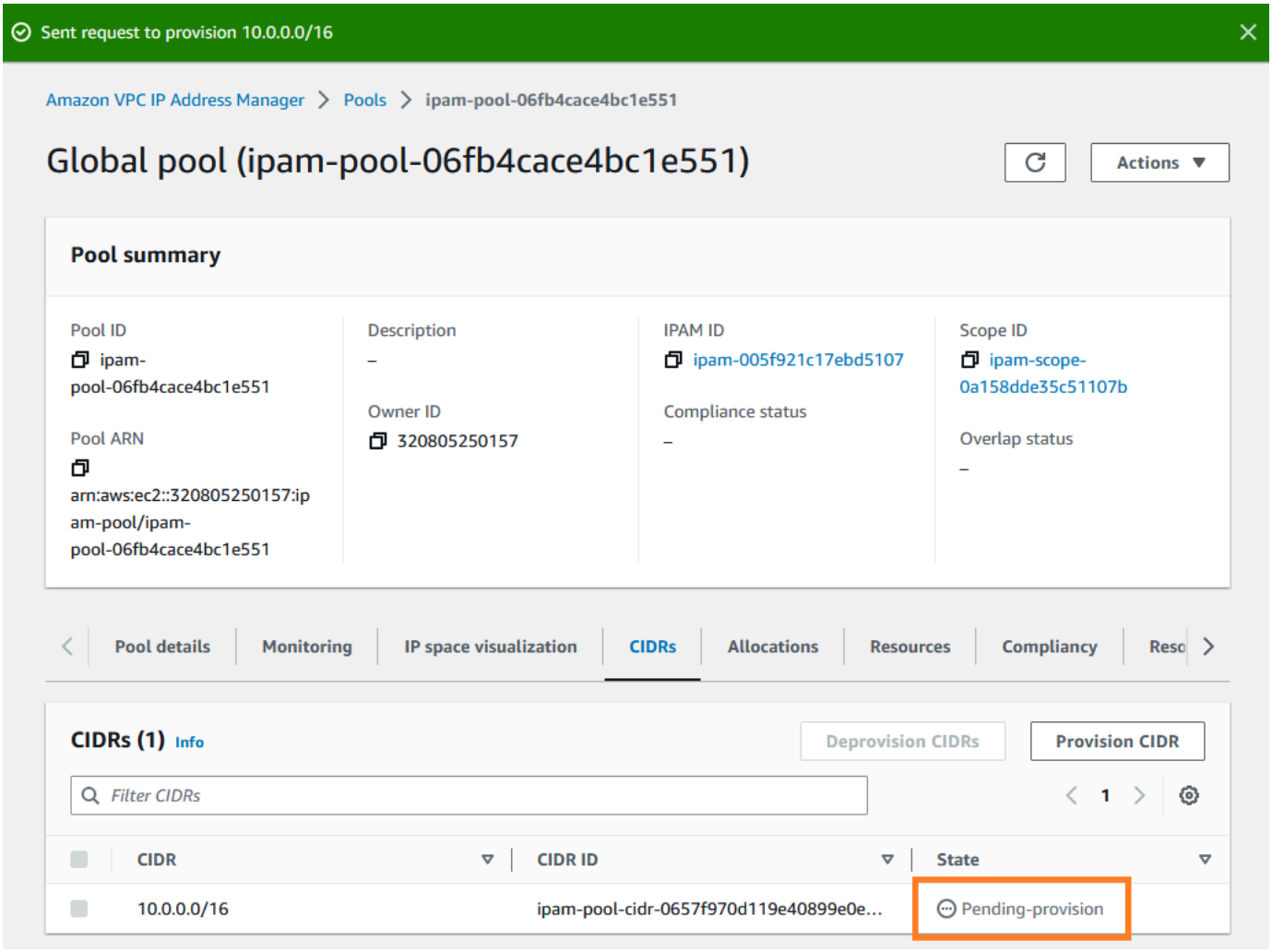
11. Choose a CIDR to provision for the pool. In this example, we provision 10.0.0.0/16.



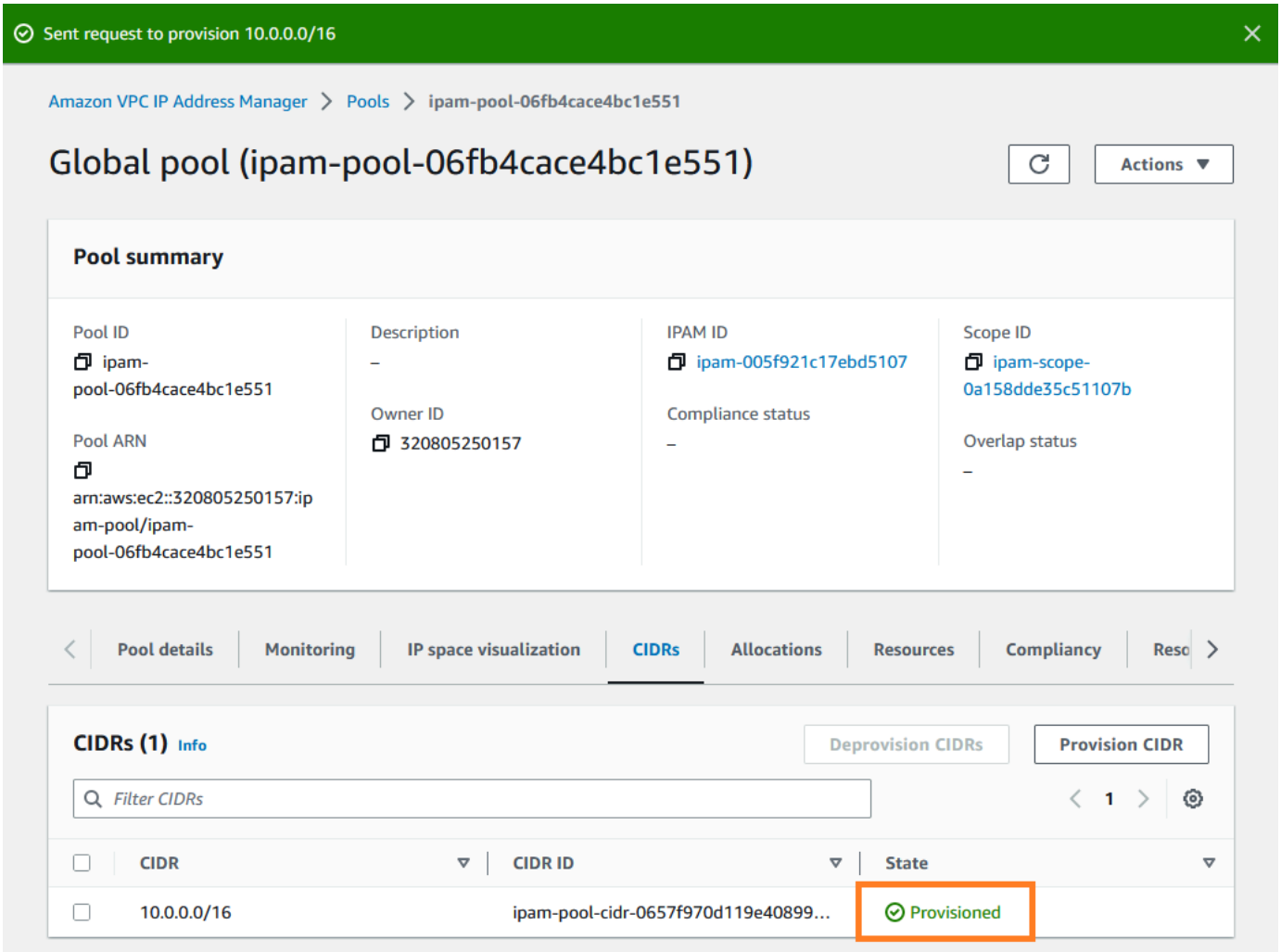
12. Leave **Configure this pool's allocation rule settings** disabled. This is our top-level pool, and you will not be allocating CIDRs to VPCs directly from this pool. Instead, you will allocate them from a sub-pool that you create from this pool.



13. Choose **Create pool**. The pool is created and the CIDR is in a **Pending-provision** state:



14. Wait for the state to be **Provisioned** before you go to the next step.



Now that you have created your top-level pool, you'll create Regional pools in us-west-1 and us-west-2.

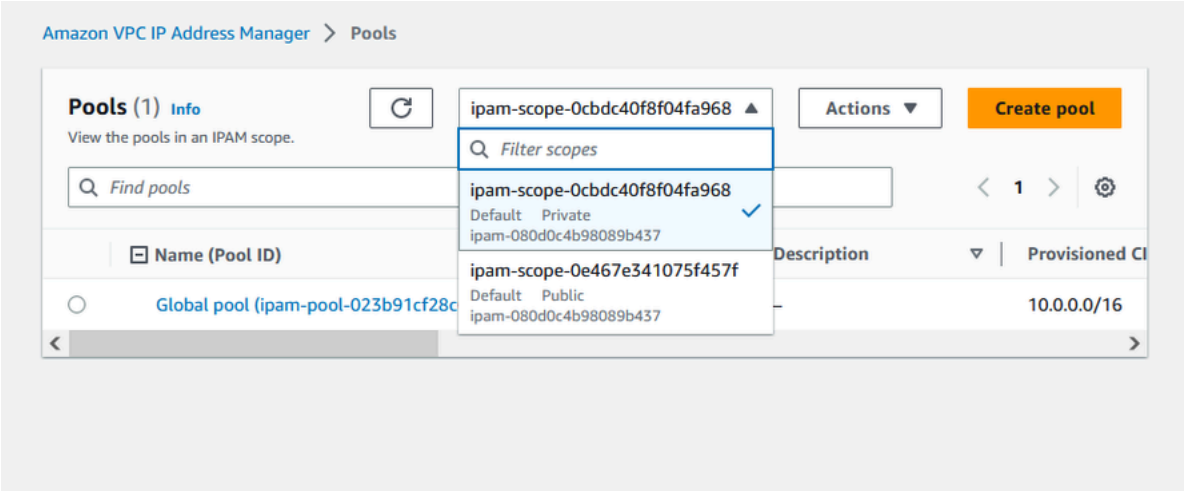
Step 4: Create Regional IPAM pools

This section shows you how to organize your IP addresses using two Regional pools. In this tutorial, we're following one of the example IPAM pool plans and creating two Regional pools which can be used by the member accounts in your organization for allocating CIDRs to their VPCs.

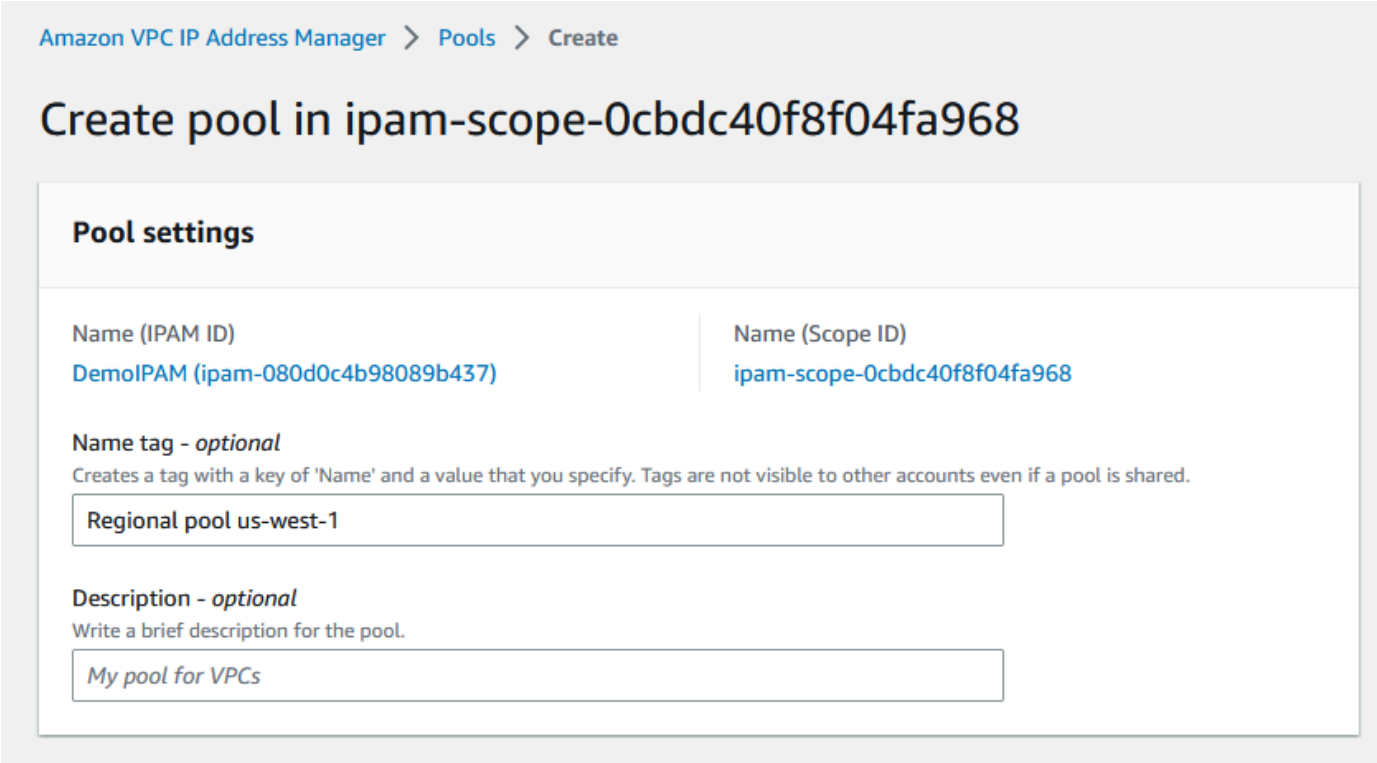
To create a Regional pool

1. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.

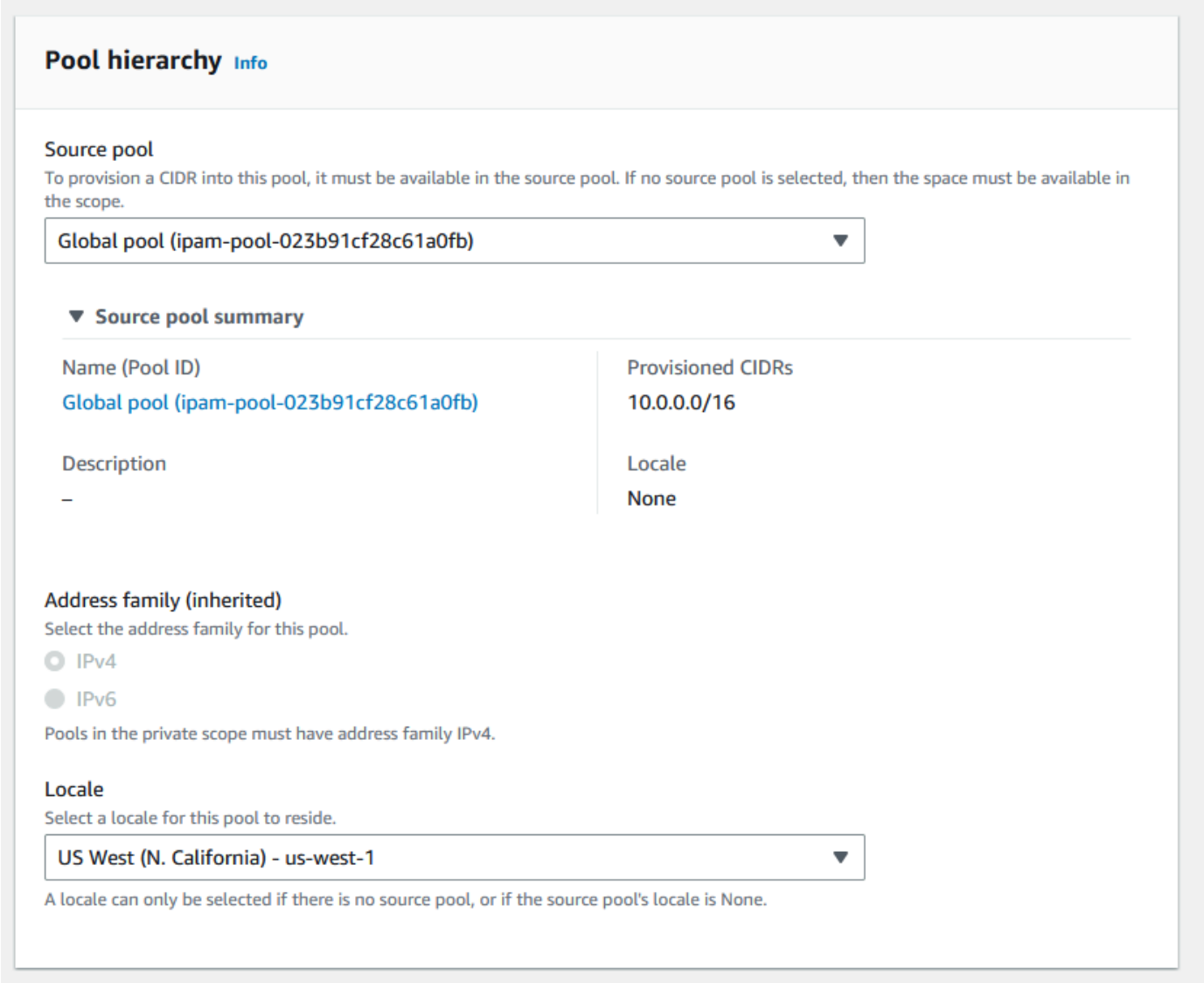
3. Choose the private scope.



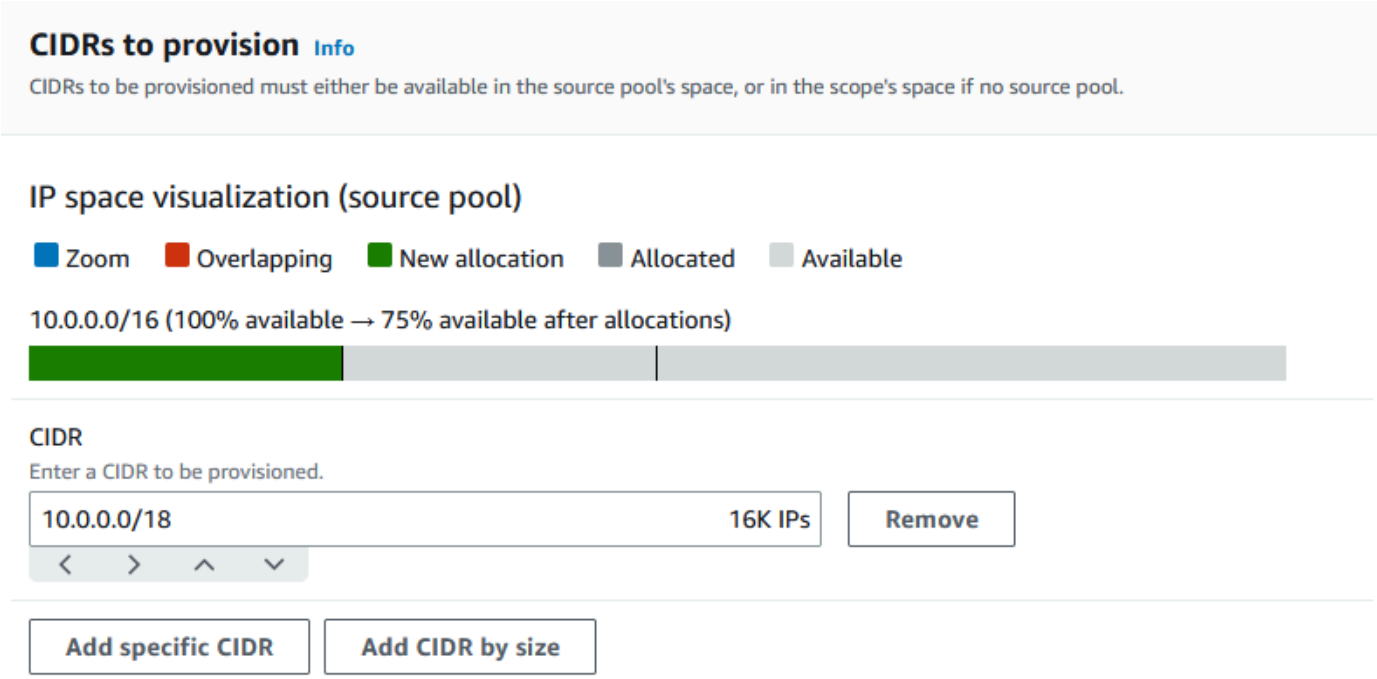
4. Choose **Create pool**.
5. Under **IPAM scope**, leave the private scope selected.
6. (Optional) Add a **Name tag** for the pool and a description for the pool, such as **Regional pool us-west-1**.



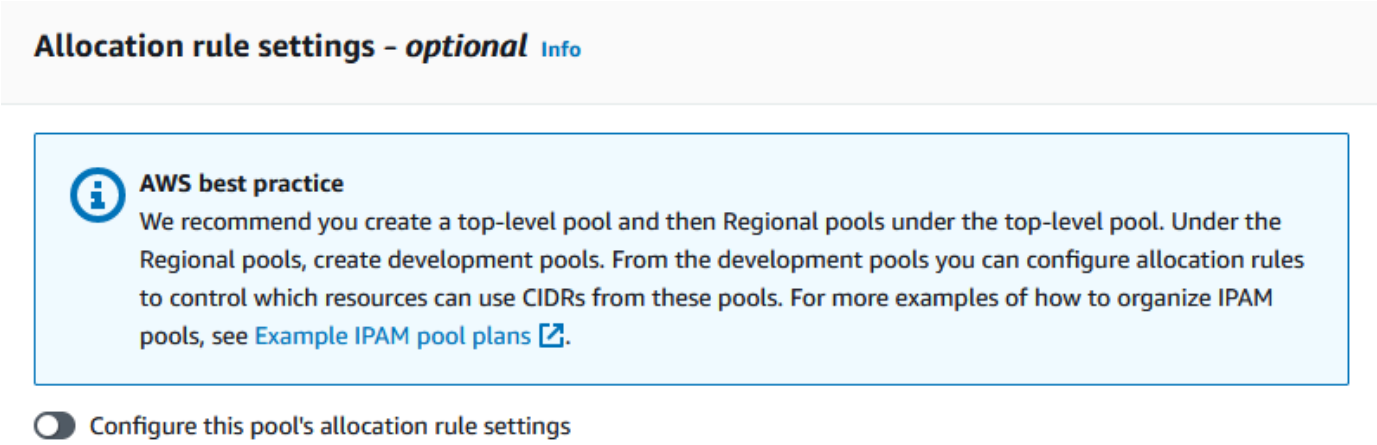
7. Under **Source**, select **IPAM pool** and select the top-level pool ("Global pool") that you created in Step 3: Create a top-level IPAM pool. Then, under **Locale**, choose **us-west-1**.



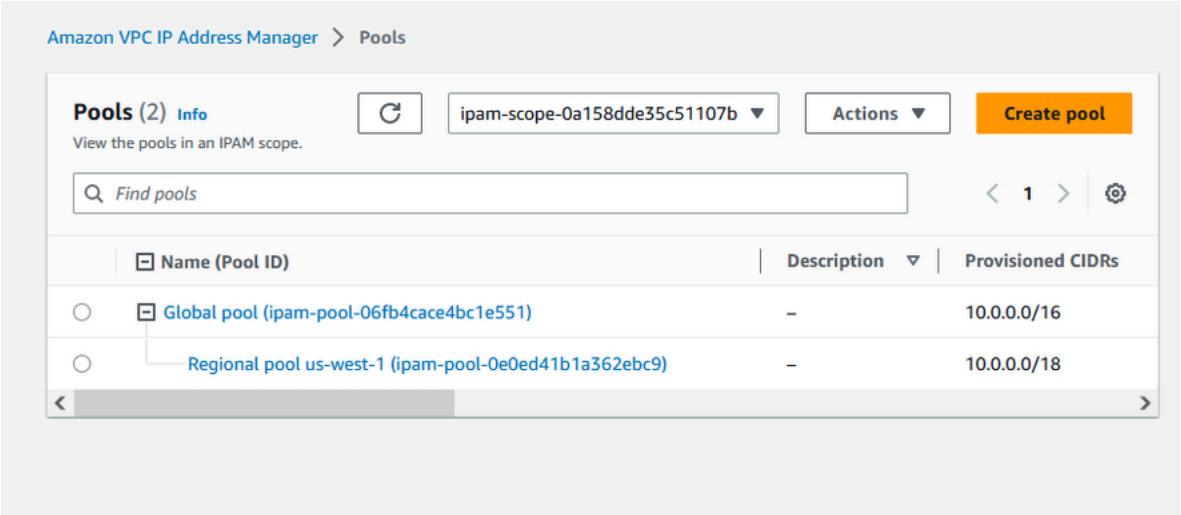
8. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.
9. Under **CIDRs to provision**, enter 10.0.0.0/18, which will give this pool around 16,000 available IP addresses.



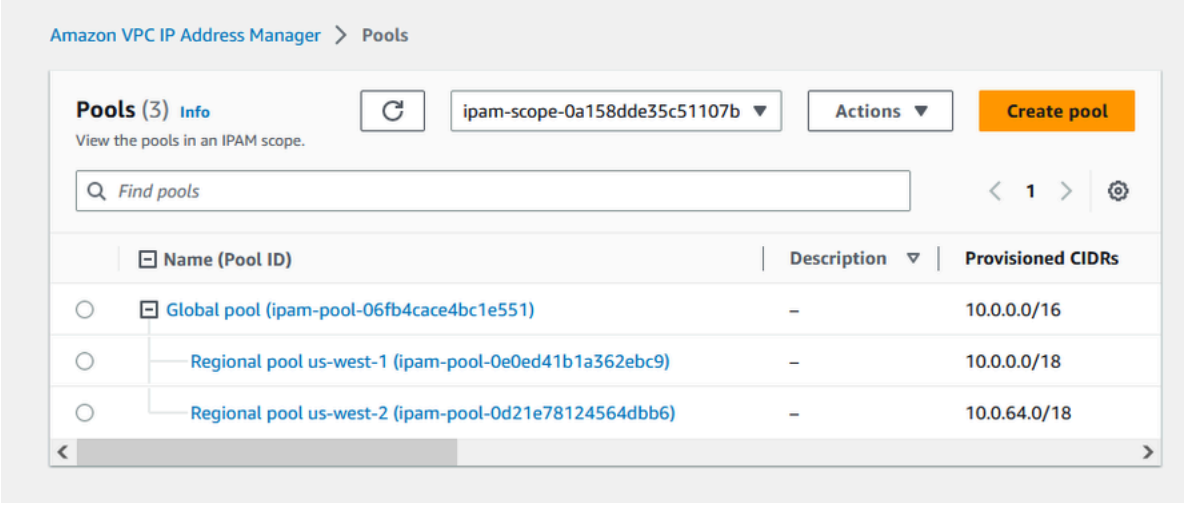
10. Leave **Configure this pool's allocation rule settings** disabled. You will not be allocating CIDRs to VPCs directly from this pool. Instead, you will allocate them from a sub-pool that you create from this pool.



11. Choose **Create pool**.
12. Return to the **Pools** view to see the hierarchy of IPAM pools that you've created.



13. Repeat the steps in this section and create a second Regional pool in **us-west-2** locale with the CIDR **10.0.64.0/18** provisioned to it. When you complete that process, you'll have three pools in a hierarchy similar to this one:

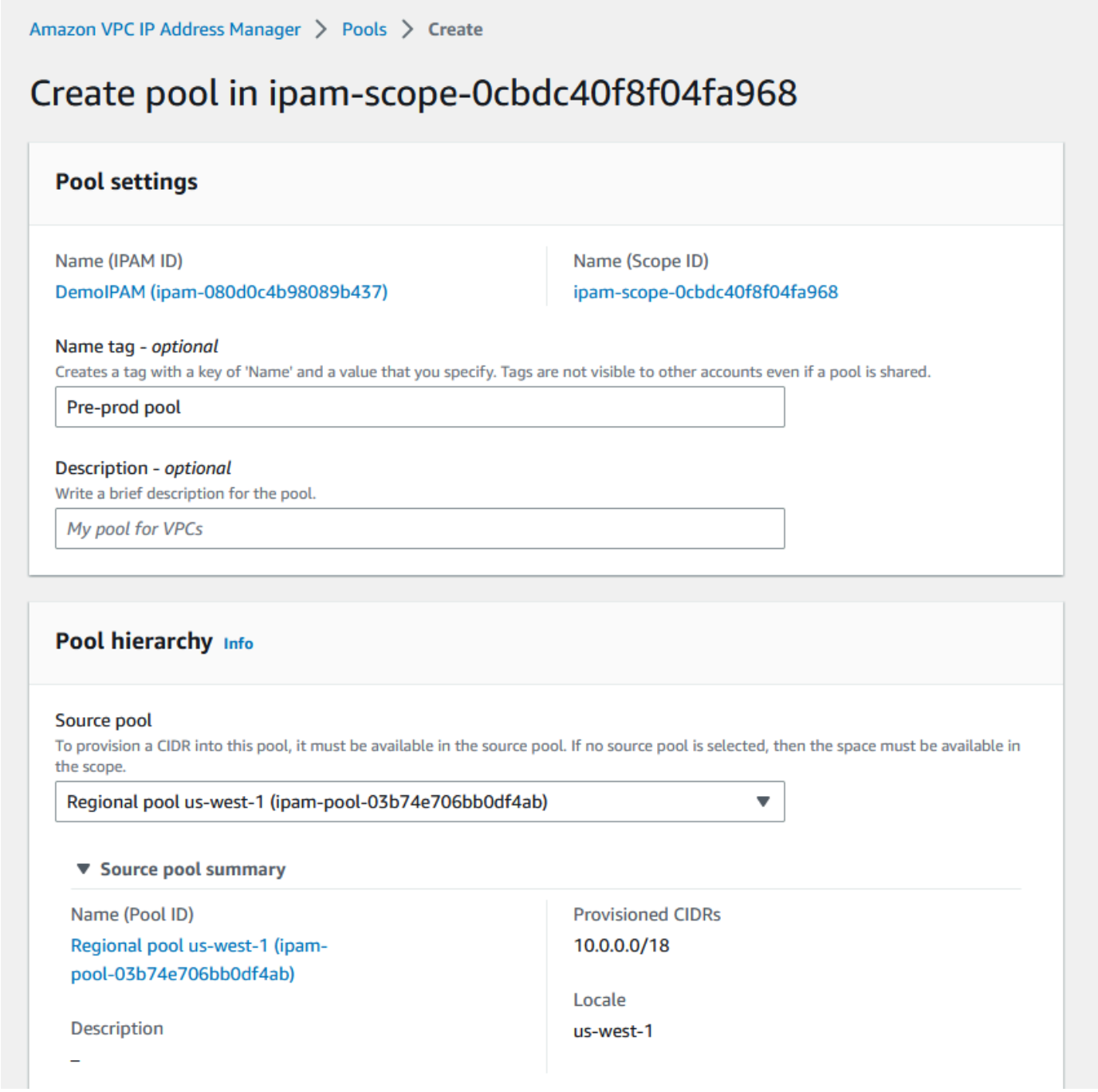


Step 5: Create a pre-production development pool

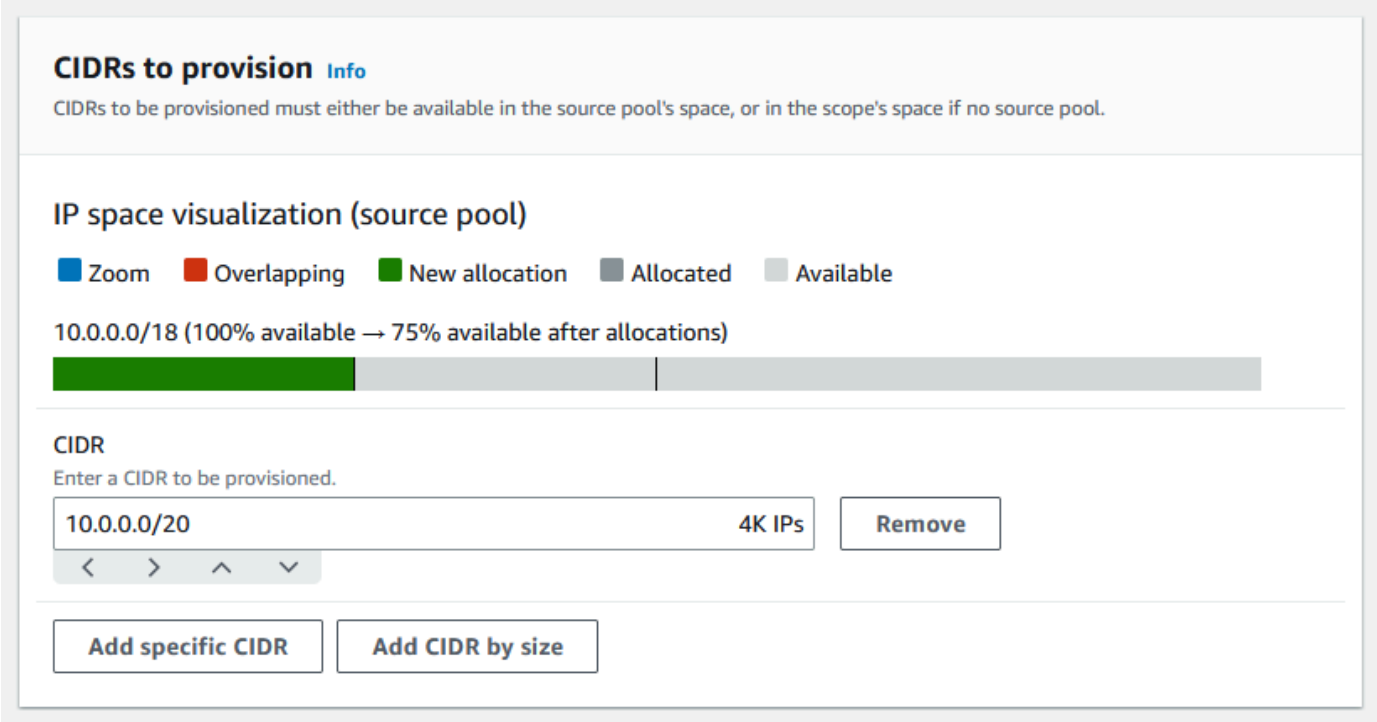
Follow the steps in this section to create a development pool for pre-production resources within one of your Regional pools.

To create a pre-production development pool

1. In the same way that you did in the previous section, using the IPAM admin account, create a pool called **Pre-prod pool**, but this time use **Regional pool us-west-1** as the source pool.

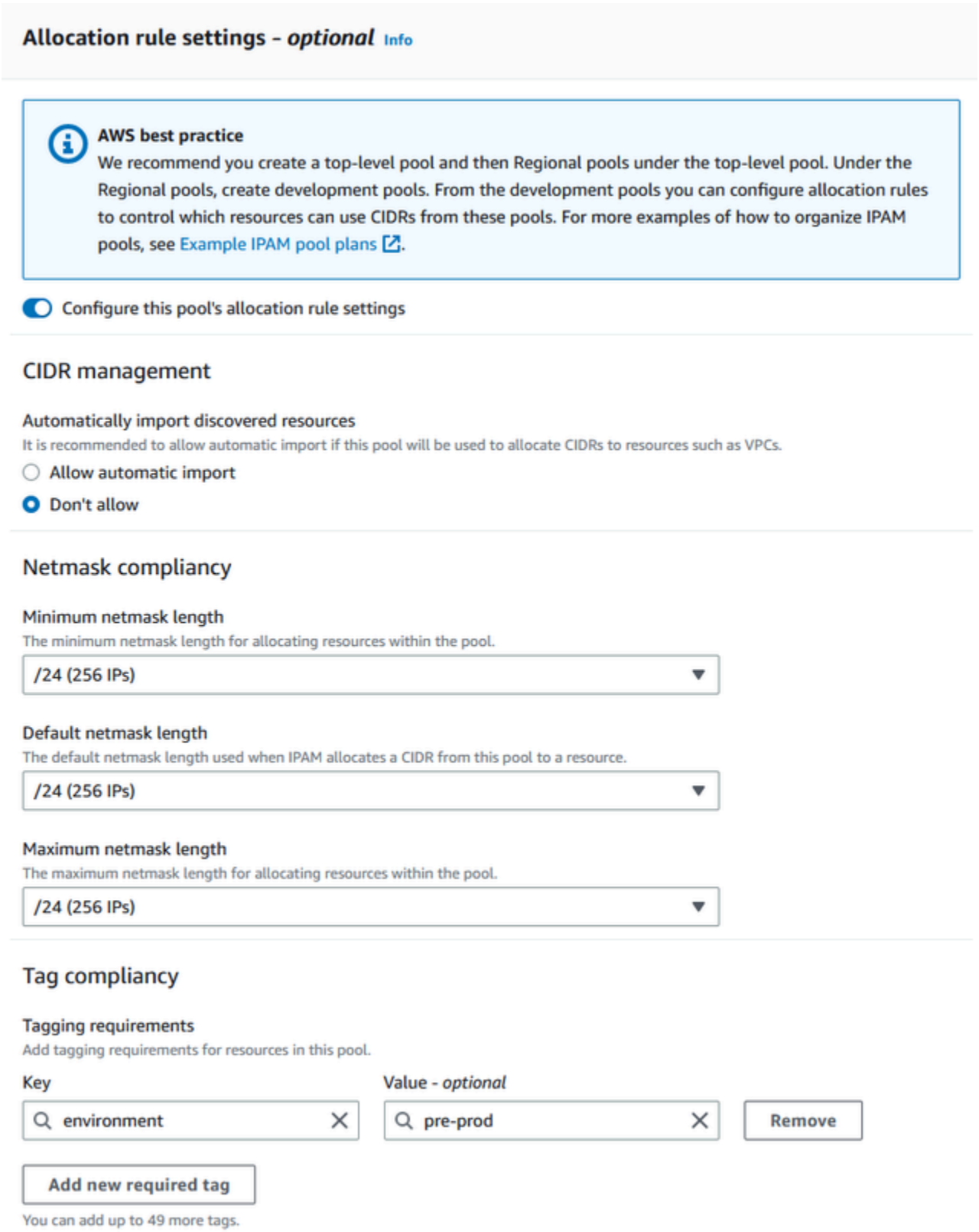


2. Specify a CIDR of 10.0.0.0/20 to provision, which will give this pool around 4,000 IP addresses.



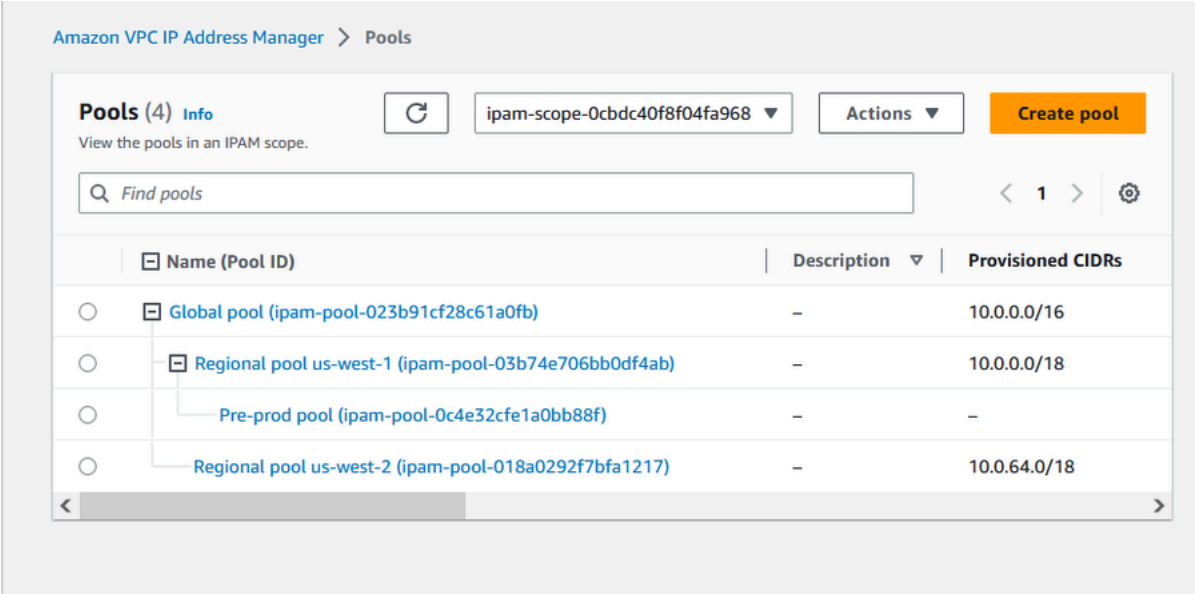
3. Toggle the option for **Configure this pool's allocation rule settings**. Do the following:

   1. Under **CIDR management**, for **Automatically import discovered resources**, leave the default **Don't allow** option selected. This option would enable IPAM to automatically import resource CIDRs it discovers in the pool's locale. A detailed description of this option is outside the scope of this tutorial, but you can read more about the option in Create a top-level IPv4 pool.
   2. Under **Netmask compliancy**, choose **/24** for the minimum, default, and maximum netmask length. A detailed description of this option is outside the scope of this tutorial, but you can read more about the option in Create a top-level IPv4 pool. What's important to note is that the VPC that you create later with a CIDR from this pool will be limited to /24 based on what we set here.
   3. Under **Tag compliance**, enter **environment/pre-prod**. This tag will be required for VPCs to allocate space from the pool. We will demonstrate later how this works.



4. Choose **Create pool**.

5. The pool hierarchy now includes an additional subpool under **Regional pool us-west-1**:



Now you're ready to share the IPAM pool with another member account in your organization and enable that account to allocate a CIDR from the pool to create a VPC.

Step 6: Share the IPAM pool

Follow the steps in this section to share the pre-production IPAM pool using AWS Resource Access Manager (RAM).

This section consists of two subsections:

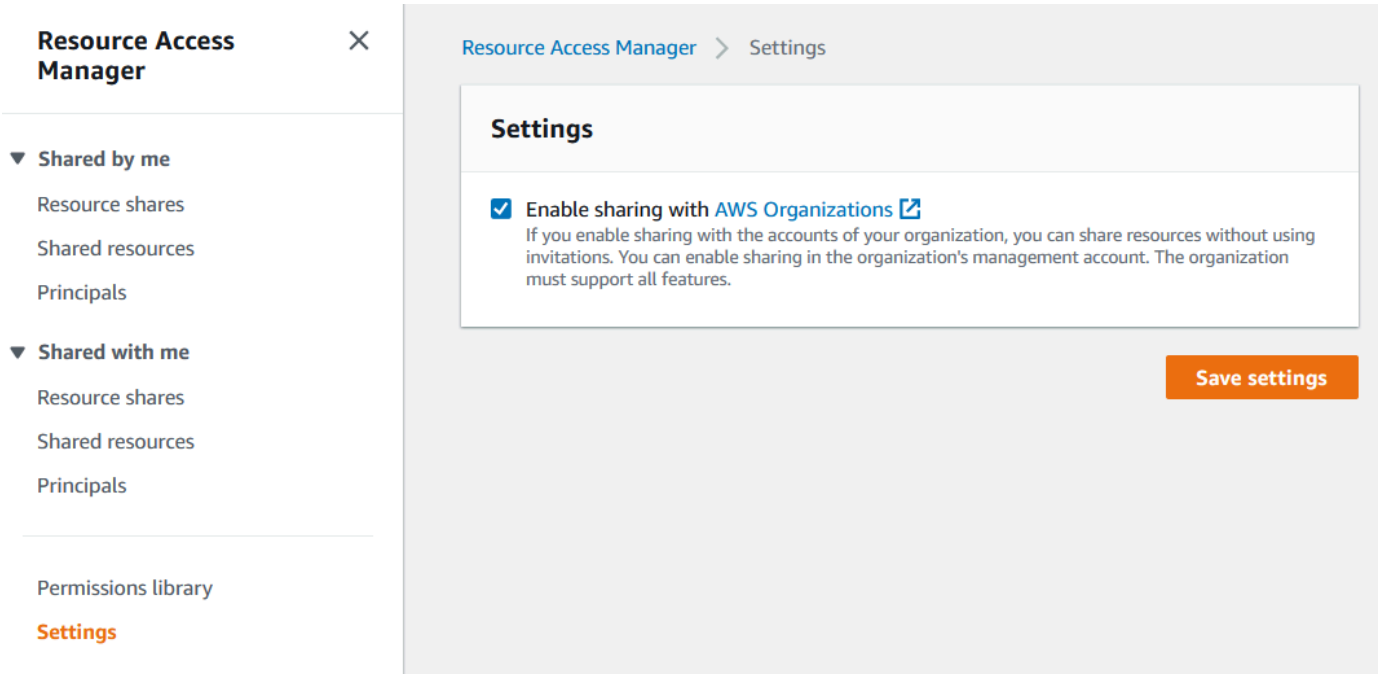
- Step 6.1. Enable resource sharing in AWS RAM: This step must be done by the AWS Organizations management account.
- Step 6.2. Share an IPAM pool using AWS RAM: This step must be done by the IPAM admin.

Step 6.1. Enable resource sharing in AWS RAM

After you create your IPAM, you'll want to share IP address pools with other accounts in your organization. Before you share an IPAM pool, complete the steps in this section to enable resource sharing with AWS RAM.

To enable resource sharing

1. Using the AWS Organizations management account, open the AWS RAM console at https://console.aws.amazon.com/ram/.
2. In the left navigation pane, choose **Settings**, choose **Enable sharing with AWS Organizations**, and then choose **Save settings**.



You can now share an IPAM pool with other members of the organization.

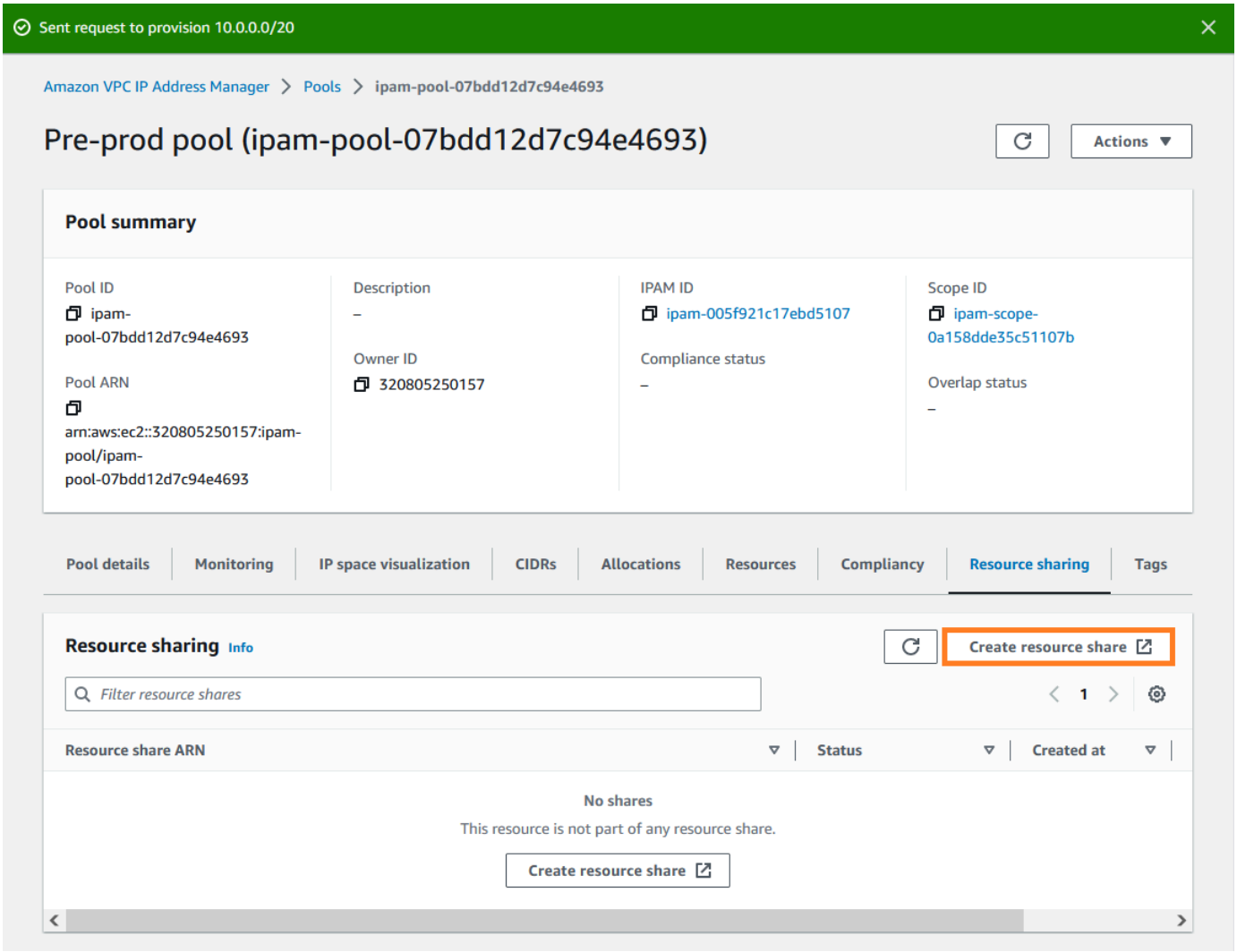
Step 6.2. Share an IPAM pool using AWS RAM

In this section you'll share the pre-production development pool with another AWS Organizations member account. For complete instructions on sharing IPAM pools, including information on the required IAM permissions, see Share an IPAM pool using AWS RAM.

To share an IPAM pool using AWS RAM

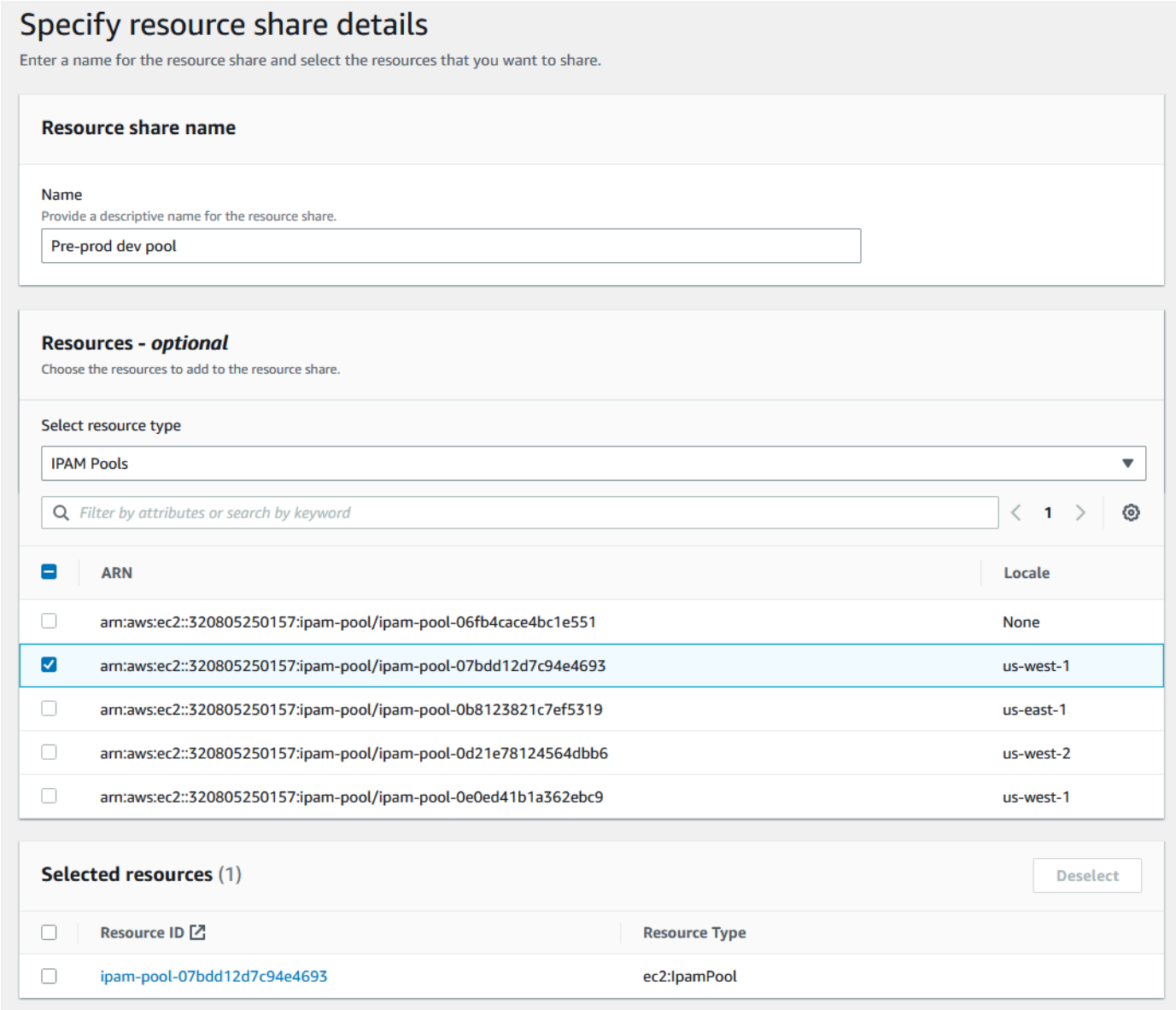
1. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. Choose the private scope, choose the pre-production IPAM pool, and choose **Actions** > **View details**.

4. Under **Resource sharing**, choose **Create resource share**. The AWS RAM console opens. You'll share the pool using AWS RAM.
5. Choose **Create a resource share**.

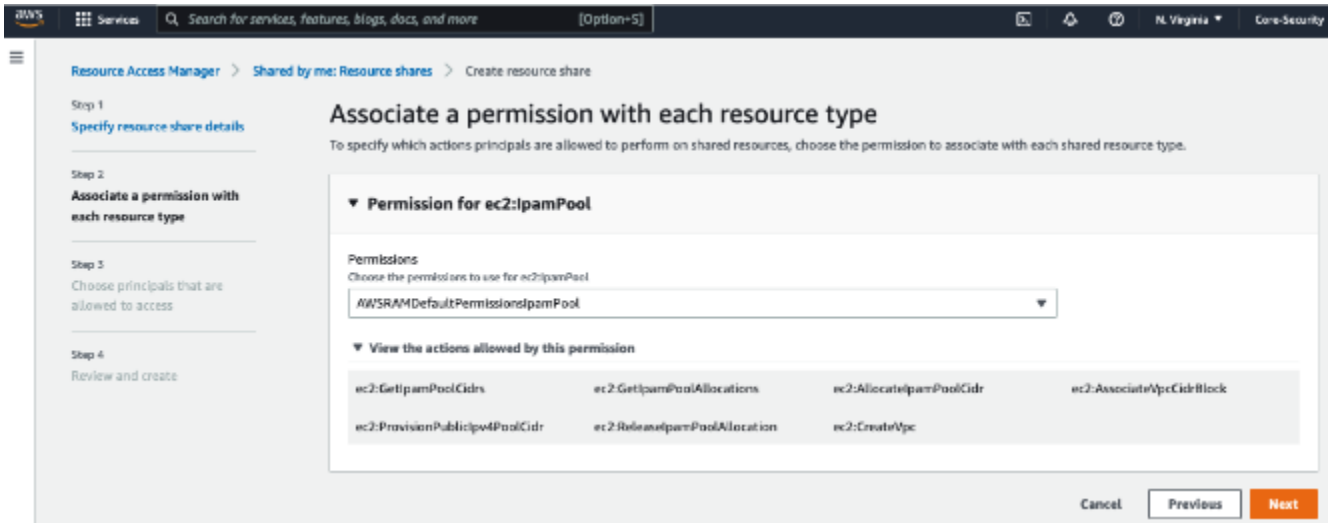


The AWS RAM console opens.

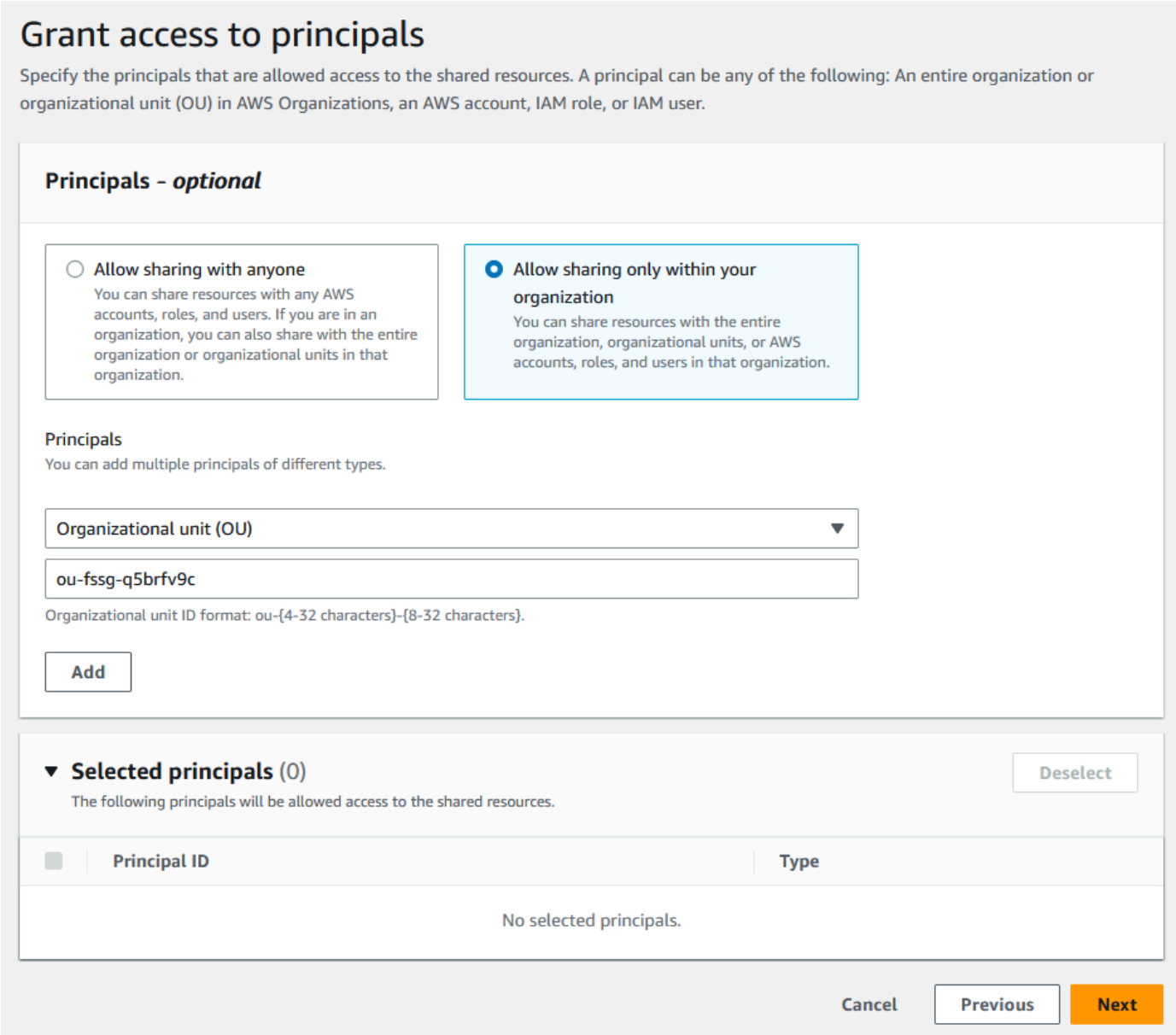
6. In the AWS RAM console, choose **Create a resource share** again.
7. Add a **Name** for the shared pool.
8. Under **Select resource type**, choose **IPAM pools,** and then choose the ARN of the pre-production development pool.



9. Choose **Next**.
10. Leave the default **AWSRAMDefaultPermissionsIpamPool** permission selected. The details of the permission options are out of scope for this tutorial, but you can find out more about these options in Share an IPAM pool using AWS RAM.



11. Choose **Next**.
12. Under **Principals**, choose **Allow sharing only within your organization.** Enter your AWS Organizations organization unit ID (as mentioned in How AWS Organizations integrates with IPAM, and then choose **Add** .



13. Choose **Next**.
14. Review the resource share options and the principals that you'll be sharing with, and then choose **Create**.

Now that the pool has been shared, go to the next step to create a VPC with a CIDR allocated from an IPAM pool.

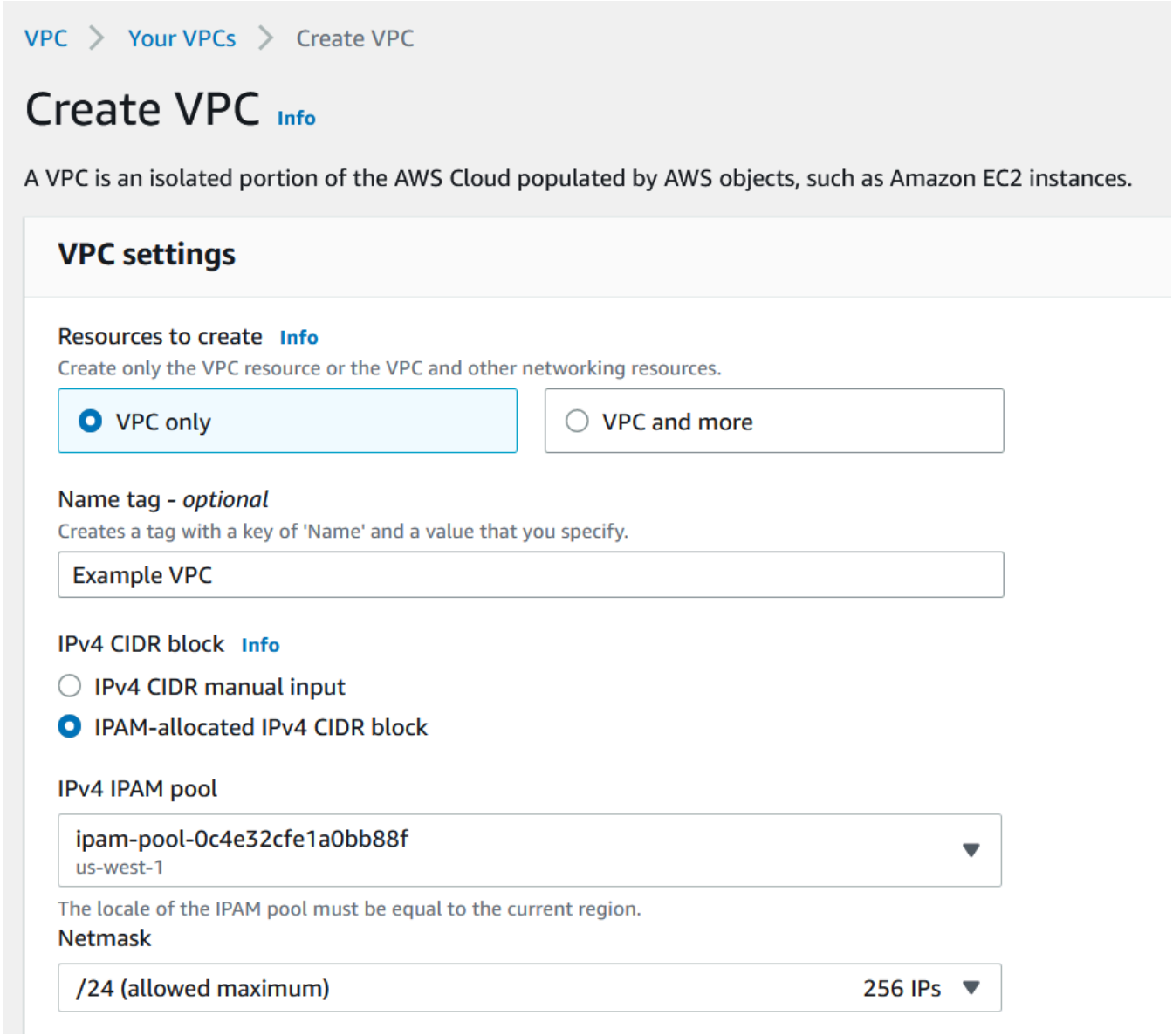
Step 7: Create a VPC with a CIDR allocated from an IPAM pool

Follow the steps in this section to create a VPC with a CIDR allocated from the pre-production pool. This step should be completed by the member account in the OU that the IPAM pool was shared

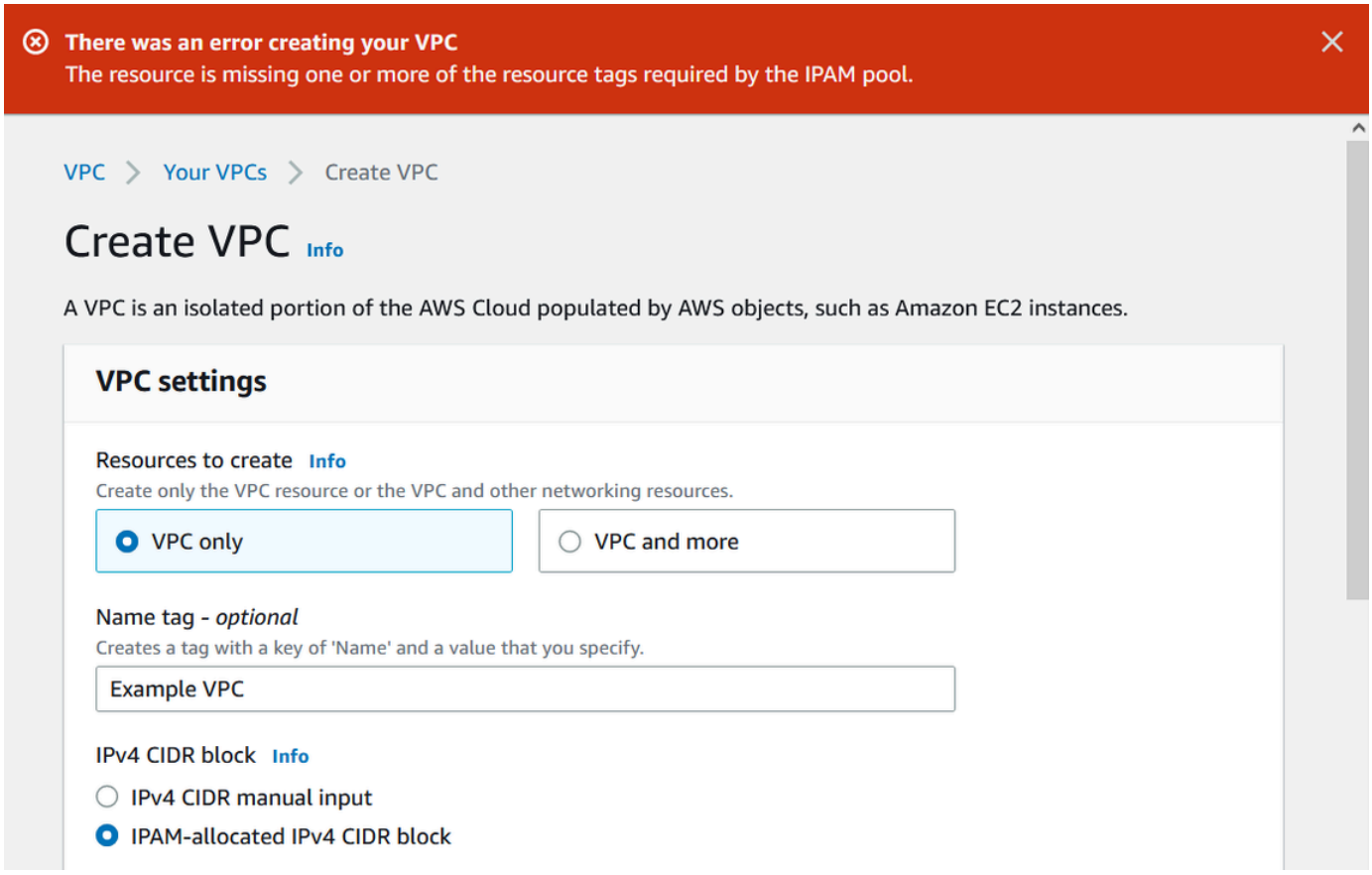
with in the previous section (called **example-member-account-2** in How AWS Organizations integrates with IPAM). For more information about the IAM permissions that are required to create VPCs, see Amazon VPC policy examples in the *Amazon VPC User Guide*.

To create a VPC with a CIDR allocated from an IPAM pool

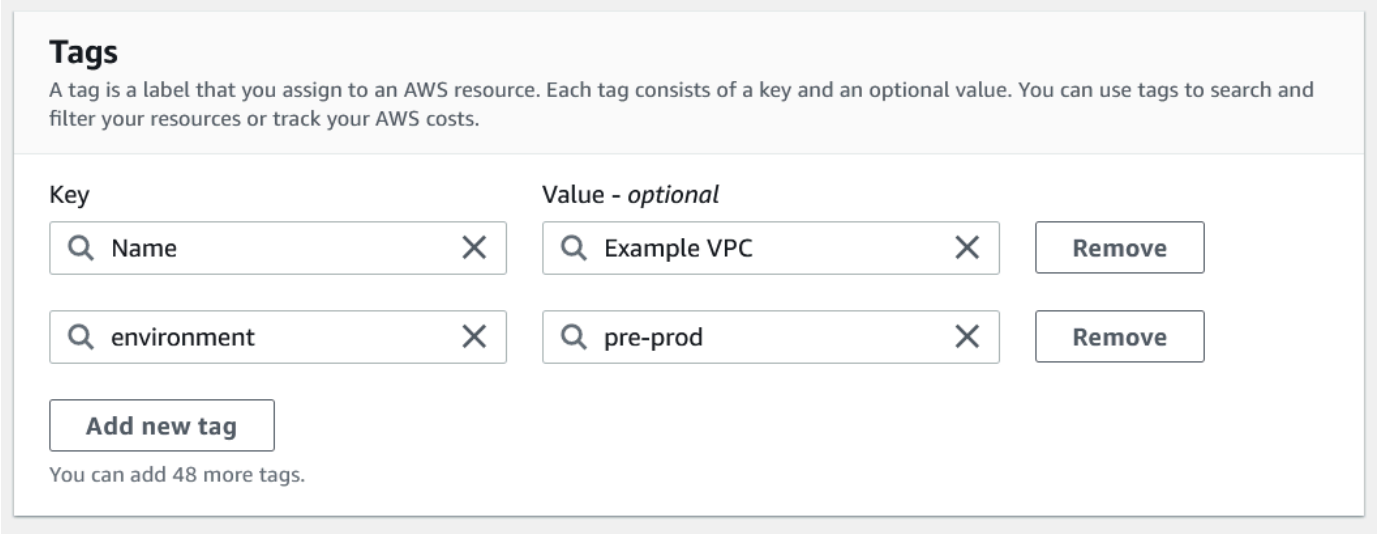
1. Using the member account, open the VPC console at https://console.aws.amazon.com/vpc/ as the member account that you'll use as the developer account.
2. Choose **Create VPC**.
3. Do the following:
    1. Enter a name, such as **Example VPC.**
    2. Choose **IPAM-allocated IPv4 CIDR block.**
    3. Under **IPv4 IPAM pool**, choose the ID of the pre-production pool.
    4. Choose a **Netmask** length. Because you limited the available netmask length for this pool to /24 (in Step 5: Create a pre-production development pool), the only netmask option available is /24.



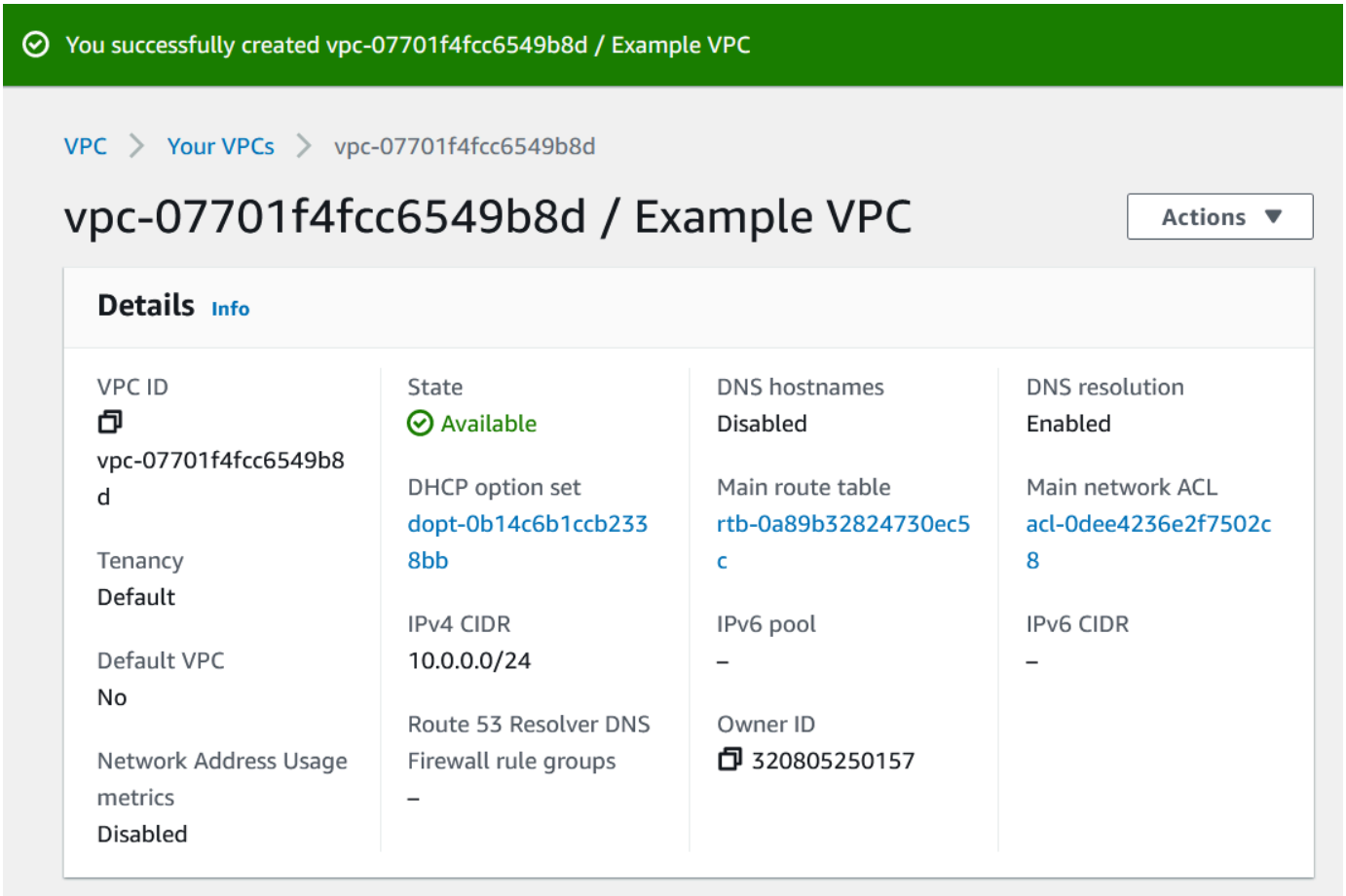
4. For demonstration purposes, under **Tags**, do not add any additional tags at this time. When you created the pre-prod pool (in Step 5: Create a pre-production development pool), you added an allocation rule that required any VPCs that are created with CIDRs from this pool to have an environment/pre-prod tag. Leave the environment/pre-prod tag off for now so that you can see that an error appears telling you that a required tag was not added.
5. Choose **Create VPC**.
6. An error appears telling you that a required tag was not added. The error appears because you set an allocation rule when you created the pre-prod pool (in Step 5: Create a pre-production development pool). The allocation rule required any VPCs that are created with CIDRs from this pool to have an environment/pre-prod tag.



7. Now, under **Tags**, add the tag **environment/pre-prod** and choose **Create VPC** again.



8. The VPC is created successfully, and the VPC complies with the tag rule on the pre-production pool:



In the **Resources** pane of the IPAM console, the IPAM admin will be able to see and manage the VPC and its allocated CIDR. Note that it takes some time for the VPC to appear in the **Resources** pane.

Step 8: Cleanup

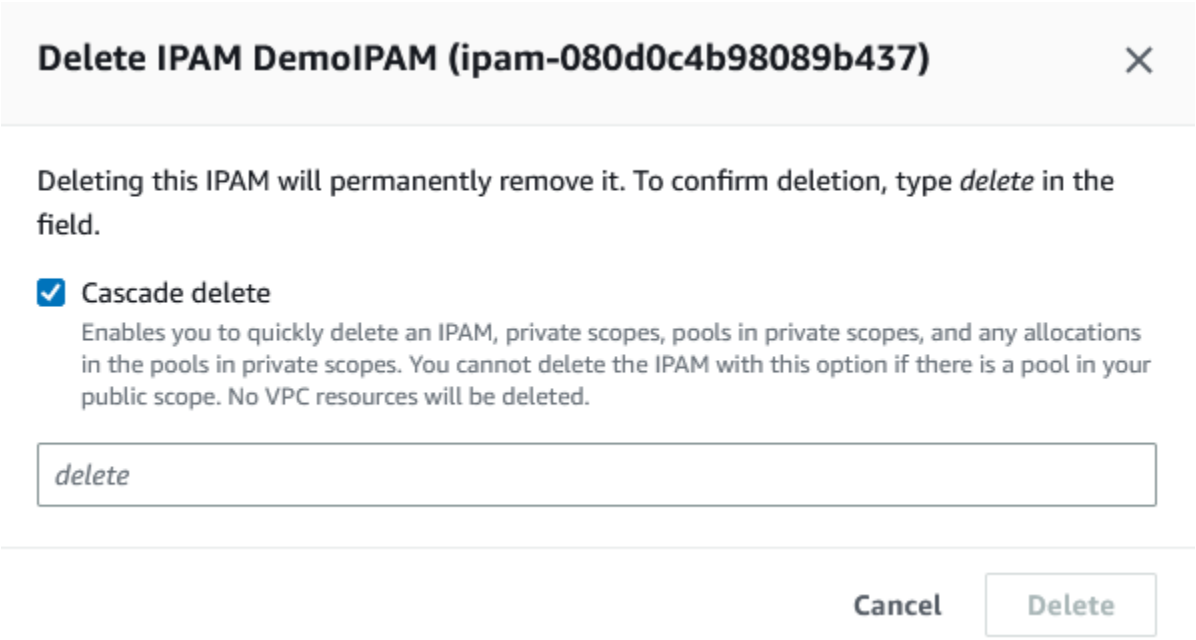
In this tutorial, you created an IPAM with a delegated admin, created multiple pools, and enabled a member account in your organization to allocate a VPC CIDR from a pool.

Follow the steps in this section to clean up the resources that you created in this tutorial.

To cleanup the resources created in this tutorial

1. Using the member account that created the example VPC, delete the VPC. For detailed instructions, see Delete your VPC in the *Amazon Virtual Private Cloud User Guide*.

2. Using the IPAM admin account, delete the example resource share in the AWS RAM console. For detailed instructions, see Deleting a resource share in AWSAWS RAM in the *AWS Resource Access Manager User Guide*.
3. Using the IPAM admin account, log into the RAM console and disable sharing with AWS Organizations that you enable in Step 6.1. Enable resource sharing in AWS RAM.
4. Using the IPAM admin account, delete the example IPAM by selecting the IPAM in the IPAM console and then choosing **Actions** > **Delete**. For detailed instructions, see Delete an IPAM.
5. When you're prompted to delete the IPAM, choose **Cascade delete**. This will delete all scopes and pools within the IPAM before deleting the IPAM.



6. Enter **delete** and then choose **Delete**.
7. Using the AWS Organizations management account, log into the IPAM console, choose **Settings**, and remove the delegated administrator account.
8. (Optional) When you integrate IPAM with AWS Organizations, IPAM automatically creates a service-linked role in each member account. Using each AWS Organizations member account, log into IAM and delete the **AWSServiceRoleForIPAM** service linked role in each member account.
9. Cleanup is complete.

Tutorial: Create an IPAM and pools using the AWS CLI

Follow the steps in this tutorial to use the AWS CLI to create an IPAM, create IP address pools, and allocate a VPC with a CIDR from an IPAM pool.

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. 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
- Step 2: Create an IPAM
- Step 3: Create an IPv4 address pool
- Step 4: Provision a CIDR to the top-level pool
- Step 5. Create a Regional pool with CIDR sourced from the top-level pool
- Step 6: Provision a CIDR to the Regional pool
- Step 7. Create a RAM share for enabling IP assignments across accounts
- Step 8. Create a VPC
- Step 9. Cleanup

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 accounts in an AWS Organization.

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.

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.

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-0008f25d7187a08d9
```

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 accounts in an AWS Organization.

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 home Region of your IPAM. 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.

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
- Scenarios

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 Search IP history 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.

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.

```
{
    "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.

```
{
    "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.


```
{
    "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.

```
{
    "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.

```
{
    "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.

```
{
    "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: Bring your ASN to IPAM

If your applications are using trusted IP addresses and Autonomous System Numbers (ASNs) that your partners or customers have allow listed in their network, you can run these applications in AWS without requiring your partners or customers to change their allow lists.

An Autonomous System Number (ASN) is a globally unique number which enables a group of networks to be identified over the internet and exchange routing data with other networks dynamically using Border Gateway Protocol. Internet service providers (ISPs), for example, use ASNs to identify the network traffic source. Not all organizations purchase their own ASNs, but for organizations which do, they can bring their ASN to AWS.

Bring your own autonomous system number (BYOASN) enables you to advertise the IPv4 or IPv6 addresses that you bring to AWS with your own public ASN instead of the AWS ASN. When you use BYOASN, the traffic originating from your IP address carries your ASN instead of the AWS ASN, and your workloads are reachable by customers or partners that have allow listed traffic based on your IP address and ASN.

Important

- Complete this tutorial using the IPAM admin account in your IPAM's home Region.
- This tutorial assumes you own the public ASN you'd like to bring to IPAM and that you've already brought a BYOIP CIDR to AWS and provisioned it to a pool in your public scope. You can bring an ASN to IPAM at any time, but to use it, you have to associate with a CIDR that you've brought to your AWS account. This tutorial assumes that you have already done that. For more information, see Tutorial: Bring your IP addresses to IPAM.
- You can change between your advertising your own ASN or an AWS ASN without delay, but you are limited to changing from an AWS ASN to your own ASN once per hour.
- If your BYOIP CIDR is currently advertised, you do not have to withdraw it from advertising to associate with your ASN.

Onboarding prerequisites for your ASN

You will need the following to complete this tutorial:

- Your public 2-byte or 4-byte ASN.
- If you've already brought an IP address range to AWS with Tutorial: Bring your IP addresses to IPAM, you need the IP address CIDR range. You'll also need a private key. You can use the private key that you created when you brought the IP address CIDR range to AWS or you can create a new private key as described in Create a private key and generate an X.509 certificate in the *Amazon EC2 User Guide*.

- When you bring an IPv4 or IPv6 address range to AWS with Tutorial: Bring your IP addresses to IPAM, you create an X.509 certificate and upload the X.509 certificate to the RDAP record in your RIR. You must upload the same certificate you created to the RDAP record in your RIR for the ASN. Be sure to include the `-----BEGIN CERTIFICATE-----` and `-----END CERTIFICATE-----` strings before and after the encoded portion. All of this content must be on a single, long line. The procedure for updating RDAP depends on your RIR:
  - For ARIN, use the Account Manager portal to add the certificate in the "Public Comments" section for the "Network Information" object representing your ASN by using the "Modify ASN" option. Do not add it to the comments section for your organization.
  - For RIPE, add the certificate as a new "descr" field to the “aut-num” object representing your ASN. These can usually be found in the "My Resources" section of the

    RIPE Database portal . Do not add it to the comments section for your organization or the "remarks" field of the “aut-num” object.
  - For APNIC, email the certificate to helpdesk@apnic.net to manually add it to the "remarks" field for your ASN. Send the email using the APNIC authorized contact for the ASN.
- When you bring an IP address range to IPAM, you create a ROA to verify that you control the IP address space that you are bringing to IPAM. In addition to that ROA, you must have a second ROA in your RIR with the ASN that you are bringing to IPAM. If you don’t have this second ROA for the ASN in your RIR, complete 3. Create a ROA object in your RIR. Ignore the other steps.

Tutorial steps

Complete the steps below using the AWS console or the AWS CLI.

AWS Management Console

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the left navigation pane, choose **IPAMs**.
3. Choose your IPAM.
4. Choose the **BYOASNs** tab and choose **Provision BYOASNs**.
5. Enter the **ASN**. As a result, the **Message** field is automatically populated with the message you will need to sign in the next step.

- The format of the message is as follows, where ACCOUNT is your AWS account number, ASN is the ASN you are bringing to IPAM, and YYYYMMDD is the expiry date of the message (which defaults to the last day of the next month). Example:

```
text_message="1|aws|ACCOUNT|ASN|YYYYMMDD|SHA256|RSAPSS"
```

6. Copy the message and replace the expiry date with your own value if you want to.
7. Sign the message using the private key. Example:

```
signed_message=$( echo -n $text_message | openssl dgst -sha256 -sigopt
 rsa_padding_mode:pss -sigopt rsa_pss_saltlen:-1 -sign private-key.pem -keyform
 PEM | openssl base64 | tr -- '+=/' '-_~' | tr -d "\n")
```

8. Under **Signature**, enter the signature.
9. (Optional) To provision another ASN, choose **Provision another ASN**. You can provision up to 5 ASNs. To increase this quota, see Quotas for your IPAM.
10. Choose **Provision**.
11. View the provisioning process in the **BYOASNs** tab. Wait for the **State** to change from *Pending-provision* to *Provisioned*. BYOASNs in a *Failed-provision* state are automatically removed after 7 days. Once the ASN is successfully provisioned, you can associate it with a BYOIP CIDR.
12. In the left navigation pane, choose **Pools**.
13. Choose your public scope. For more information about scopes, see How IPAM works.
14. Choose a regional pool that has a BYOIP CIDR provisioned to it. The pool must have **Service** set to **EC2** and must have a locale chosen.
15. Choose the **CIDRs** tab and select a BYOIP CIDR.
16. Choose **Actions** > **Manage BYOASN associations**.
17. Under **Associated BYOASNs**, choose the ASN you brought to AWS. If you have multiple ASNs, you can associate multiple ASNs to the BYOIP CIDR. You can associate as many ASNs as you can bring to IPAM. Note that you can bring up to 5 ASNs to IPAM by default. For more information, see Quotas for your IPAM.
18. Choose **Associate**.
19. Wait for the ASN association to complete. Once the ASN is successfully associated with the BYOIP CIDR, you can advertise the BYOIP CIDR again.
20. Choose the pool **CIDRs** tab.

21. Select the BYOIP CIDR and choose **Actions** > **Advertise**. As a result, your ASN options are displayed: the Amazon ASN and any ASNs you've brought to IPAM.
22. Select the ASN you brought to IPAM and choose **Advertise CIDR**. As a result, the BYOIP CIDR is advertised and the value in the **Advertising** column changes from Withdrawn to Advertised. The **Autonomous System Number** column displays the ASN associated with the CIDR.
23. (optional) If you decide that you want to change the ASN association back to the Amazon ASN, select the BYOIP CIDR and choose **Actions** > **Advertise** again. This time, choose the Amazon ASN. You can swap back to the Amazon ASN at any time, but you can only change to a custom ASN once every hour.

The tutorial is complete.

Cleanup

1. Disassociate the ASN from the BYOIP CIDR
   - To withdraw the BYOIP CIDR from advertising, in your pool in the public scope, choose the BYOIP CIDR and choose **Actions** > **Withdraw from advertising**.
   - To disassociate the ASN from the CIDR, choose **Actions** > **Manage BYOASN associations**.
2. Deprovision the ASN
   - To deprovision the ASN, in the BYOASNs tab, choose the ASN and choose **Deprovision ASN**. As a result, the ASN is deprovisioned. BYOASNs in a *Deprovisioned* state are automatically removed after 7 days.

Cleanup is complete.

Command line

1. Provision your ASN by including your ASN and authorization message. The signature is the message signed with your private key.

```
aws ec2 provision-ipam-byoasn --ipam-id $ipam_id --asn 12345 --asn-
authorization-context Message="$text_message",Signature="$signed_message"
```

2. Describe your ASN to track the provisioning process. If the request succeeds, you should see the *ProvisionStatus* set to *provisioned* after a few minutes.

```
aws ec2 describe-ipam-byoasn
```

3. Associate your ASN with your BYOIP CIDR. Any custom ASN you wish to advertise from must first be associated with your CIDR.

```
aws ec2 associate-ipam-byoasn --asn 12345 --cidr xxx.xxx.xxx.xxx/n
```

4. Describe your CIDR to track the association process.

```
aws ec2 describe-byoip-cidrs --max-results 10
```

5. Advertise your CIDR with your ASN. If the CIDR is already advertised, this will swap the origin ASN from Amazon's to yours.

```
aws ec2 advertise-byoip-cidr --asn 12345 --cidr xxx.xxx.xxx.xxx/n
```

6. Describe your CIDR to see the ASN state change from *associated* to *advertised*.

```
aws ec2 describe-byoip-cidrs --max-results 10
```

The tutorial is complete.

Cleanup

1. Do one of the following:

   - To withdraw just your ASN advertisement and go back to using the Amazon ASNs while keeping the CIDR advertised you must call advertise-byoip-cidr with the special AWS value for the asn parameter. You can swap back to the Amazon ASN at any time, but you can only change to a custom ASN once every hour.

     ```
     aws ec2 advertise-byoip-cidr --asn AWS --cidr xxx.xxx.xxx.xxx/n
     ```

   - To withdraw your CIDR and ASN advertisement simultaneously, you can call withdraw-byoip-cidr.

     ```
     aws ec2 withdraw-byoip-cidr --cidr xxx.xxx.xxx.xxx/n
     ```

2. To clean up your ASN, you must first disassociate it from your BYOIP CIDR.

```
aws ec2 disassociate-ipam-byoasn --asn 12345 --cidr xxx.xxx.xxx.xxx/n
```

3. Once your ASN is disassociated from all the BYOIP CIDRs with which you associated it, you can deprovision it.

```
aws ec2 deprovision-ipam-byoasn --ipam-id $ipam_id --asn 12345
```

4. The BYOIP CIDR can also be deprovisioned once all ASN associations are removed.

```
aws ec2 deprovision-ipam-pool-cidr --ipam-pool-id ipam-pool-1234567890abcdef0 --
cidr xxx.xxx.xxx.xxx/n
```

5. Confirm the deprovisioning.

```
aws ec2 get-ipam-pool-cidrs --ipam-pool-id ipam-pool-1234567890abcdef0
```

Cleanup is complete.

Tutorial: Bring your IP addresses 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 to resources like EC2 instances and application load balancers. Without IPAM, you have to onboard your public IPs for each AWS Region.

Contents

- Verify domain control
- Bring your own IP to IPAM using both the AWS Management Console and the AWS CLI

- Bring your own IP CIDR to IPAM using only the AWS CLI

Verify domain control

Before you bring an IP address range to AWS, you have to use one of the options described in this section to verify that you control the IP address space. Later, when you bring the IP address range to AWS, AWS validates that you control the IP address range. This validation ensures that customers cannot use IP ranges belonging to others, preventing routing and security issues.

There are two methods that you can use to verify that you control the range:

- **X.509 certificate**: If your IP address range is registered with an Internet Registry that supports RDAP (such as ARIN, RIPE and APNIC), you can use an X.509 certificate to verify ownership of your domain.
- **DNS TXT record**: Regardless of whether your Internet Registry supports RDAP, you can use a verification token and a DNS TXT record to verify ownership of your domain.

Contents

- Verify your domain with an X.509 certificate
- Verify your domain with a DNS TXT record

Verify your domain with an X.509 certificate

This section describes how to verify your domain with an X.509 certificate before you bring your IP address range to IPAM.

To verify your domain with an X.509 certificate

1. Complete the three steps in Prerequisites for BYOIP in Amazon EC2 in the *Amazon EC2 User Guide*.

> **Note**
>
> 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.

2. Complete steps 1 and 2 only under Provision a publicly advertisable address range in AWS in the *Amazon EC2 User Guide*, **and don't provision the address range (step 3) yet**. Save the `text_message` and `signed_message`. You'll need them later in this process.

When you've completed these steps, continue with Bring your own IP to IPAM using both the AWS Management Console and the AWS CLI or Bring your own IP CIDR to IPAM using only the AWS CLI.

Verify your domain with a DNS TXT record

Complete the steps in this section to verify your domain with a DNS TXT record before you bring your IP address range to IPAM.

You can use DNS TXT records to validate that you control a public IP address range. DNS TXT records are a type of DNS record that contain information about your domain name. This feature enables you to bring IP addresses registered with any internet registry (such as JPNIC, LACNIC, and AFRINIC), not just those that support RDAP (Registration Data Access Protocol) record-based validations (such as ARIN, RIPE and APNIC).

Important

Before you can continue, you must have already created an IPAM in the Free or Advanced Tier. If you don't have an IPAM, complete Create an IPAM first.

Contents

- Step 1: Create a ROA if you don't have one
- Step 2. Create a verification token
- Step 3. Set up the DNS zone and TXT record

Step 1: Create a ROA if you don't have one

You must have a Route Origin Authorization (ROA) in your Regional Internet Registry (RIR) for IP address ranges you wish to advertise. If you don't have a ROA in your RIR, complete 3. Create a ROA object in your RIR in the *Amazon EC2 User Guide*. Ignore the other steps.

The most specific IPv4 address range that you can bring is /24. The most specific IPv6 address range that you can bring is /48 for CIDRs that are publicly advertisable and /60 for CIDRs that are not publicly advertisable.

Step 2. Create a verification token

A verification token is an AWS-generated random value that you can use to prove control of an external resource. For example, you can use a verification token to validate that you control a public IP address range when you bring an IP address range to AWS (BYOIP).

Complete the steps in this section to create a verification token which you'll need in a later step in this tutorial to bring your IP address range to IPAM. Use the instructions below for either the AWS console or the AWS CLI.

AWS Management Console

To create a verification token

1. Open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the AWS Management Console, choose the AWS Region where you created your IPAM.
3. In the left navigation pane, choose **IPAMs**.
4. Choose your IPAM and then choose the **Verification tokens tab**.
5. Select **Create verification token**.
6. After you create the token, leave this browser tab open. You'll need the **Token value**, **Token name** in the next step and the **Token ID** in a later step.

Note the following:

- Once you create a verification token, you can reuse the token for multiple BYOIP CIDRs that you provision from your IPAM within 72 hours. If you want to provision more CIDRs after 72 hours, you need a new token.
- You can create up to 100 tokens. If you reach the limit, delete expired tokens.

Command line

- Request that IPAM creates a verification token that you will use for the DNS configuration with create-ipam-external-resource-verification-token:

```
aws ec2 create-ipam-external-resource-verification-token --ipam-id ipam-id
```

This will return an IpamExternalResourceVerificationTokenId and token with `TokenName` and `TokenValue`, and the expiration time (`NotAfter`) of the token.

```
{
    "IpamExternalResourceVerificationToken": {
        "IpamExternalResourceVerificationTokenId": "ipam-ext-res-ver-
token-0309ce7f67a768cf0",
        "IpamId": "ipam-0f9e8725ac3ae5754",
        "TokenValue": "a34597c3-5317-4238-9ce7-50da5b6e6dc8",
        "TokenName": "86950620",
        "NotAfter": "2024-05-19T14:28:15.927000+00:00",
        "Status": "valid",
        "Tags": [],
        "State": "create-in-progress" }
}
```

Note the following:

- Once you create a verification token, you can reuse the token for multiple BYOIP CIDRs that you provision from your IPAM within 72 hours. If you want to provision more CIDRs after 72 hours, you need a new token.
- You can view your tokens using describe-ipam-external-resource-verification-tokens.
- You can create up to 100 tokens. If you reach the limit, you can delete expired tokens using delete-ipam-external-resource-verification-token.

Step 3. Set up the DNS zone and TXT record

Complete the steps in this section to set up the DNS zone and TXT record. If you are not using Route53 as your DNS, then follow the documentation provided by your DNS provider to set up a DNS Zone and add a TXT record.

If you are using Route53, note the following:

- To create a Reverse Lookup Zone in the AWS console, see Creating a public hosted zone in the *Amazon Route 53 Developer Guide* or use the AWS CLI command create-hosted-zone.
- To create a record in the Reverse Lookup Zone in the AWS console, see Creating records by using the Amazon Route 53 console in the *Amazon Route 53 Developer Guide* or use the AWS CLI command change-resource-record-sets.
- After you are done creating your hosted zone, delegate the hosted zone from your RIR to the name servers provided by Route53 (such as for LACNIC or APNIC).

Whether you are using another DNS provider or Route53, when you set up the TXT record, note the following:

- Record name should be your token name.
- Record type should be TXT.
- ResourceRecord Value should be the token value.

Example:

- **Name**: `86950620.113.0.203.in-addr.arpa`
- **Type**: TXT
- **ResourceRecords Value**: `a34597c3-5317-4238-9ce7-50da5b6e6dc8`

Where:

- `86950620` is the verification token name.
- `113.0.203.in-addr.arpa` is the Reverse Lookup Zone name.
- TXT is the record type.
- `a34597c3-5317-4238-9ce7-50da5b6e6dc8` is the verification token value.

> **Note**
>
> Depending on the size of the prefix to be brought to IPAM with BYOIP, one or more authentication records must be created in the DNS. These authentication records are of the record type TXT and must be placed into the reverse zone of the prefix itself or its parent prefix.

- For IPv4, authentication records need to align to ranges at an octet boundary that make up the prefix.
  - **Examples**
  - For 198.18.123.0/24, which is already aligned at an octet boundary, you would need to create a single authentication record at:
    - *token-name*.123.18.198.in-addr.arpa. IN TXT "*token-value*"
  - For 198.18.12.0/22, which itself is not aligned to octet boundary, you would need to create four authentication records. These records must cover the subnets 198.18.12.0/24, 198.18.13.0/24, 198.18.14.0/24, and 198.18.15.0/24 which are aligned at an octet boundary. The corresponding DNS entries must be:
    - *token-name*.12.18.198.in-addr.arpa. IN TXT "*token-value*"
    - *token-name*.13.18.198.in-addr.arpa. IN TXT "*token-value*"
    - *token-name*.14.18.198.in-addr.arpa. IN TXT "*token-value*"
    - *token-name*.15.18.198.in-addr.arpa. IN TXT "*token-value*"
  - For 198.18.0.0/16, which is already aligned at an octet boundary, you need to create a single authentication record:
    - *token-name*.18.198.in-addr.arpa. IN TXT "*token-value*"
- For IPv6, authentication records need to align to ranges at nibble boundary that make up the prefix. Valid nibble values are e.g. 32, 36, 40, 44, 48, 52, 56, and 60.
  - **Examples**
    - For 2001:0db8::/40, which is already aligned at nibble boundary, you need to create a single authentication record:
      - *token-name*.0.0.8.b.d.0.1.0.0.2.ip6.arpa TXT "*token-value*"
    - For 2001:0db8:80::/42, which is itself not aligned at nibble boundary, you need to create four authentication records. These records must cover the subnets 2001:db8:80::/44, 2001:db8:90::/44, 2001:db8:a0::/44, and 2001:db8:b0::/44 which are aligned at a nibble boundary. The corresponding DNS entries must be:
      - *token-name*.8.0.0.8.b.d.0.1.0.0.2.ip6.arpa TXT "*token-value*"
      - *token-name*.9.0.0.8.b.d.0.1.0.0.2.ip6.arpa TXT "*token-value*"
      - *token-name*.a.0.0.8.b.d.0.1.0.0.2.ip6.arpa IN TXT "*token-value*"
      - *token-name*.b.0.0.8.b.d.0.1.0.0.2.ip6.arpa IN TXT "*token-value*"

- For the non-advertised range 2001:db8:0:1000::/54, which is itself not aligned at a nibble boundary, you need to create four authentication records. These records must cover the subnets 2001:db8:0:1000::/56, 2001:db8:0:1100::/56, 2001:db8:0:1200::/56, and 2001:db8:0:1300::/56 which are aligned at a nibble boundary. The corresponding DNS entries must be:
  - `token-name.0.1.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa IN TXT "token-value"`
  - `token-name.1.1.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa IN TXT "token-value"`
  - `token-name.2.1.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa IN TXT "token-value"`
  - `token-name.3.1.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa IN TXT "token-value"`
- To validate the correct number of hexadecimal numbers between the *token-name* and the "ip6.arpa" string, multiply the number by four. The result should match the prefix length. For example, for a /56 prefix you should have 14 hexadecimal digits.

When you've completed these steps, continue with Bring your own IP to IPAM using both the AWS Management Console and the AWS CLI or Bring your own IP CIDR to IPAM using only the AWS CLI.

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

Bringing Your Own IP (BYOIP) to IPAM allows you to use your organization's existing IPv4 and IPv6 address ranges in AWS. This enables you to maintain consistent branding, improve network performance, enhance security, and simplify management by unifying on-premises and cloud environments under your own IP address space.

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

Note

Before you begin, you must have first verified domain control.

Once you bring an IPv4 address range to AWS, you can use all of the IP addresses in the range, including the first address (the network address) and the last address (the broadcast address).

Contents

- Bring your own IPv4 CIDR to IPAM using both the AWS Management Console and the AWS CLI
- Bring your own IPv6 CIDR to IPAM using the AWS Management Console

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 accounts in an AWS Organization.
  - Create an IPAM.
- 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 accounts in an AWS Organization. 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 and IAM roles
- Step 2: Create a top-level IPAM pool
- Step 3. Create a Regional pool within the top-level pool
- Step 4: Advertise the CIDR
- Step 5. Share the Regional pool
- Step 6: Allocate an Elastic IP address from the pool

- Step 7: Associate the Elastic IP address with an EC2 instance
- Step 8: Cleanup
- Alternative to Step 6

Step 1: Create AWS CLI named profiles and IAM roles

To complete this tutorial as a single AWS user, you can use AWS CLI named profiles to switch from one IAM role to another. Named profiles are collections of settings and credentials that you refer to when using the `--profile` option with the AWS CLI. For more information about how to create IAM roles and named profiles for AWS accounts, see Using an IAM role in the AWS CLI.

Create one role and one named profile 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.

After you have created the IAM roles and 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.

4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a **Description** for the pool.
6. Under **Source**, choose **IPAM scope**.
7. Under **Address family**, choose **IPv4**.
8. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.
9. Under **Locale**, choose **None**.

   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.

10. Under **Public IP source**, choose **BYOIP**.
11. Under **CIDRs to provision**, do one of the following:

    - If you verified your domain control with an X.509 certificate, you must include the CIDR and the BYOIP message and certificate signature that you created in that step so we can verify that you control the public space.
    - If you verified your domain control with a DNS TXT record, you must include the CIDR and IPAM verification token that you created in that step so we can verify that you control the public space.

    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.

> **Important**
>
> While most provisioning will be completed within two hours, it may take up to one week to complete the provisioning process for publicly advertisable ranges.

12. Leave **Configure this pool's allocation rule settings** unselected.
13. (Optional) Choose **Tags** for the pool.
14. 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. 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 part of one of the operating Regions you configured when you created the IPAM. For example, a locale of *us-east-1* means that *us-east-1* must be an operating Region for the IPAM. A locale of *us-east-1-scl-1* (a network border group used for Local Zones) means that the IPAM must have an operating Region of *us-east-1*.

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.
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a **Description** for the pool.
6. Under **Source**, choose the top-level pool that you created in the previous section.

7. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.
8. 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 for the pool should be one of the following:

   - An AWS Region where you want this IPAM pool to be available for allocations.
   - The network border group for an AWS Local Zone where you want this IPAM pool to be available for allocations (supported Local Zones). This option is only available for IPAM IPv4 pools in the public scope.
   - An AWS Dedicated Local Zone. To create a pool within an AWS Dedicated Local Zone, enter the AWS Dedicated Local Zone in the selector input.

   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. If the home Region of the IPAM is unavailable due to an outage and the pool has a locale different than the home Region of the IPAM, the pool can still be used to allocate IP addresses.

   Choosing a locale ensures there are no cross-region dependencies between your pool and the resources allocating from it.
9. Under **Service**, choose **EC2 (EIP/VPC)**. The service you select determines the AWS service where the CIDR will be advertisable. Currently, the only option is **EC2 (EIP/VPC)**, which means that the CIDRs allocated from this pool will be advertisable for the Amazon EC2 service (for Elastic IP addresses) and the Amazon VPC service (for CIDRs associated with VPCs).
10. Under **CIDRs to provision**, choose a CIDR to provision for the pool.

> **Note**
>
> When provisioning a CIDR to a Regional pool within the top-level pool, the most specific IPv4 CIDR you can provision is /24; more specific CIDRs (such as /25) are not permitted. After you create the Regional pool, you can create smaller pools (such as /25) within the same Regional pool. Note that if you share the Regional pool or pools within it, these pools can only be used in the locale set on the same Regional pool.

11. Enable **Configure this pool's allocation rule settings**. You have the same allocation rule options here as you did when you created the top-level pool. See Create a top-level IPv4 pool 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.
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: 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.

> **Note**
>
> The advertisement status doesn't not restrict your ability to allocate Elastic IP addresses. Even if your BYOIPv4 CIDR is not advertised, you can still can create EIPs from the IPAM pool.

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.
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 5. Share the Regional pool

Follow the steps in this section to share the IPAM pool using AWS Resource Access Manager (RAM).

Enable resource sharing in AWS RAM

After you create your IPAM, you'll want to share the regional pool with other accounts in your organization. Before you share an IPAM pool, complete the steps in this section to enable resource sharing with AWS RAM. If you are using the AWS CLI to enable resource sharing, use the `--profile` **`management-account`** option.

To enable resource sharing

1. Using the AWS Organizations management account, open the AWS RAM console at https://console.aws.amazon.com/ram/.
2. In the left navigation pane, choose **Settings**, choose **Enable sharing with AWS Organizations**, and then choose **Save settings**.

You can now share an IPAM pool with other members of the organization.

Share an IPAM pool using AWS RAM

In this section you'll share the regional pool with another AWS Organizations member account. For complete instructions on sharing IPAM pools, including information on the required IAM permissions, see Share an IPAM pool using AWS RAM. If you are using the AWS CLI to enable resource sharing, use the `--profile` **`ipam-account`** option.

To share an IPAM pool using AWS RAM

1. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. Choose the private scope, choose the IPAM pool, and choose **Actions** > **View details**.

4. Under **Resource sharing**, choose **Create resource share**. The AWS RAM console opens. You share the pool using AWS RAM.
5. Choose **Create a resource share**.
6. In the AWS RAM console, choose **Create a resource share** again.
7. Add a **Name** for the shared pool.
8. Under **Select resource type**, choose **IPAM pools,** and then choose the ARN of the pool you want to share.
9. Choose **Next**.
10. Choose the **AWSRAMPermissionIpamPoolByoipCidrImport** permission. The details of the permission options are out of scope for this tutorial, but you can find out more about these options in Share an IPAM pool using AWS RAM.
11. Choose **Next**.
12. Under **Principals** > **Select principal type**, choose **AWS account** and enter the account ID of the account that will be bringing an IP address range to IPAM and choose **Add** .
13. Choose **Next**.
14. Review the resource share options and the principals that you'll be sharing with, and then choose **Create**.
15. To allow the **`member-account`** account to allocate IP address CIDRS from the IPAM pool, create a second resource share with `AWSRAMDefaultPermissionsIpamPool`. The value for `--resource-arns` is the ARN of the IPAM pool that you created in the previous section. The value for `--principals` is the account ID of the **`member-account`**. The value for `--permission-arns` is the ARN of the `AWSRAMDefaultPermissionsIpamPool` permission.

Step 6: Allocate an Elastic IP address from the pool

Complete the steps in this section to allocate an Elastic IP address from the pool. Note that if you are using public IPv4 pools to allocate Elastic IP addresses, you can use the alternative steps in Alternative to Step 6 rather than the steps in this section.

> **Important**
>
> If you see an error related to not having permissions to call ec2:AllocateAddress, the managed permission currently assigned to the IPAM pool that was shared with you needs to be updated. Contact the person who created the resource share and ask them to update

the managed permission `AWSRAMPermissionIpamResourceDiscovery` to the default version. For more information, see Update a resource share in the *AWS RAM User Guide* .

AWS Management Console

Follow the steps in Allocate an Elastic IP address in the *Amazon EC2 User Guide* to allocate the address, but note the following:

- This step must be done by the member account.
- Ensure that the AWS Region you are in in the EC2 console matches the Locale option you chose when you created the Regional pool.
- When you choose the address pool, choose the option to **Allocate using an IPv4 IPAM pool** and choose the Regional pool you created.

Command line

Allocate an address from the pool with the allocate-address command. The `--region` you use must match the `-locale` option you chose when you created the pool in Step 2. Include the ID of the IPAM pool you created in Step 2 in `--ipam-pool-id`. Optionally, you can also choose a specific `/32` in your IPAM pool by using the `--address` option.

```
aws ec2 allocate-address --region us-east-1 --ipam-pool-id ipam-
pool-07ccc86aa41bef7ce
```

Example response:

```
{
    "PublicIp": "18.97.0.41",
    "AllocationId": "eipalloc-056cdd6019c0f4b46",
    "PublicIpv4Pool": "ipam-pool-07ccc86aa41bef7ce",
    "NetworkBorderGroup": "us-east-1",
    "Domain": "vpc"
}
```

For more information, see Allocate an Elastic IP address in the *Amazon EC2 User Guide*.

Step 7: Associate the Elastic IP address with an EC2 instance

Complete the steps in this section to associate the Elastic IP address with an EC2 instance.

AWS Management Console

Follow the steps in Associate an Elastic IP address in the *Amazon EC2 User Guide* to allocate an Elastic IP address from the IPAM pool, but note the following: When you use AWS Management Console option, the AWS Region you associate the Elastic IP address in must match the Locale option you chose when you created the Regional pool.

This step must be done by the member account.

Command line

This step must be done by the member account. Use the `--profile` **`member-account`** option.

Associate the Elastic IP address with an instance with the associate-address command. The `--region` you associate the Elastic IP address in must match the `--locale` option you chose when you created the Regional pool.

```
aws ec2 associate-address --region us-east-1 --instance-id i-07459a6fca5b35823 --
public-ip 18.97.0.41
```

Example response:

```
{
    "AssociationId": "eipassoc-06aa85073d3936e0e"
}
```

For more information, see Associate an Elastic IP address with an instance or network interface in the *Amazon EC2 User Guide*.

Step 8: 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* 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* 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: Delete any RAM shares 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 shares 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 shares and disable RAM integration with AWS Organizations.

Step 5: 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 to deprovision the CIDRs from the Regional pool and then the top-level pool, in that order.

Step 6: 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 to delete the Regional pool and then the top-level pool, in that order.

Alternative to Step 6

If you are using public IPv4 pools to allocate Elastic IP addresses, you can use the steps in this section rather than the steps in Step 6: Allocate an Elastic IP address from the pool.

Contents

- Step 1: Create a public IPv4 pool
- Step 2: Provision the public IPv4 CIDR to your public IPv4 pool
- Step 3: Allocate an Elastic IP address from the public IPv4 pool
- Alternative to Step 6 cleanup

Step 1: Create a public IPv4 pool

This step should be done by the member account that will provision an Elastic IP address.

> **Note**
>
> - This step must be done by the member account using the AWS CLI.
> - 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 2: 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. The `--netmask-length` is the amount of space out of the IPAM pool that you want to bring to your public pool. The value cannot be larger than the netmask length of the IPAM pool. The least specific `--netmask-length` you can define is 24.

Note

- If you are bringing a /24 CIDR range to IPAM to share across an AWS Organization, you can provision smaller prefixes to multiple IPAM pools, say /27 (using `-- netmask-length 27`), rather than provisioning the entire /24 CIDR (using `-- netmask-length 24`) as is shown in this tutorial.
- 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-public-ipv4-pools --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.

```
{
```

```
    "PublicIpv4Pools": [
        {
            "PoolId": "ipv4pool-ec2-09037ce61cf068f9a",
            "Description": "",
            "PoolAddressRanges": [
                {
                    "FirstAddress": "130.137.245.0",
                    "LastAddress": "130.137.245.255",
                    "AddressCount": 256,
                    "AvailableAddressCount": 255
                }
            ],
            "TotalAddressCount": 256,
            "TotalAvailableAddressCount": 255,
            "NetworkBorderGroup": "us-east-2",
            "Tags": []
        }
    ]
}
```


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 3: Allocate an Elastic IP address from the public IPv4 pool

Complete the steps in Allocate an Elastic IP address in the *Amazon EC2 User Guide* to allocate an 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.

Once you've completed these three steps, return to Step 7: Associate the Elastic IP address with an EC2 instance and continue until you complete the tutorial.

Alternative to Step 6 cleanup

Complete these steps to clean up public IPv4 pools created with the alternative to Step 9. You should complete these steps after you release the Elastic IP address during the standard cleanup process in Step 8: Cleanup.

Step 1: Deprovision the public IPv4 CIDR from your public IPv4 pool

> **Important**
>
> 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 2: 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**.

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.

If you do not need to advertise your IPv6 addresses over the Internet, you can provision a private GUA IPv6 address to an IPAM. For more information, see Enable provisioning private IPv6 GUA CIDRs.

Important

- This tutorial assumes you have already completed the steps in the following sections:
  - Integrate IPAM with accounts in an AWS Organization.
  - Create an IPAM.
- 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 accounts in an AWS Organization. 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
- Step 2. Create a Regional pool within the top-level pool
- Step 3. Share the Regional pool
- Step 4: Create a VPC
- Step 5: Advertise the CIDR
- Step 6: Cleanup

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 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.
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a **Description** for the pool.
6. Under **Source**, choose **IPAM scope**.
7. Under **Address family**, choose **IPv6**.
8. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.
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. If the home Region of the IPAM is unavailable due to an outage and the pool has a locale different than the home Region of the IPAM, the pool can still be used to allocate IP addresses.

   > **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 **Public IP source**, **BYOIP** is selected by default.
11. Under **CIDRs to provision**, do one of the following:

    - If you verified your domain control with an X.509 certificate, you must include the CIDR and the BYOIP message and certificate signature that you created in that step so we can verify that you control the public space.
    - If you verified your domain control with a DNS TXT record, you must include the CIDR and IPAM verification token that you created in that step so we can verify that you control the public space.

Note that when provisioning an IPv6 CIDR to a pool within the top-level pool, the most specific IPv6 address range that you can bring is /48 for CIDRs that are publicly advertisable and /60 for CIDRs that are not publicly advertisable.

> **Important**
>
> While most provisioning will be completed within two hours, it may take up to one week to complete the provisioning process for publicly advertisable ranges.

12. Leave **Configure this pool's allocation rule settings** unselected.
13. (Optional) Choose **Tags** for the pool.
14. 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 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.
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a description for the pool.
6. Under **Source**, choose the top-level pool that you created in the previous section.
7. Under **Resource planning**, leave **Plan IP space within the scope** selected. For more information about using this option to plan for subnet IP space within a VPC, see Tutorial: Plan VPC IP address space for subnet IP allocations.

8. 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. If the home Region of the IPAM is unavailable due to an outage and the pool has a locale different than the home Region of the IPAM, the pool can still be used to allocate IP addresses.

9. Under **Service**, choose **EC2 (EIP/VPC)**. The service you select determines the AWS service where the CIDR will be advertisable. Currently, the only option is **EC2 (EIP/VPC)**, which means that the CIDRs allocated from this pool will be advertisable for the Amazon EC2 service and the Amazon VPC service (for CIDRs associated with VPCs).

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 most specific IPv6 address range that you can bring is /48 for CIDRs that are publicly advertisable and /60 for CIDRs that are not publicly advertisable.

11. Enable **Configure this pool's allocation rule settings** 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.

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 3. Share the Regional pool

Follow the steps in this section to share the IPAM pool using AWS Resource Access Manager (RAM).

Enable resource sharing in AWS RAM

After you create your IPAM, you'll want to share the regional pool with other accounts in your organization. Before you share an IPAM pool, complete the steps in this section to enable resource sharing with AWS RAM. If you are using the AWS CLI to enable resource sharing, use the `--profile` **`management-account`** option.

To enable resource sharing

1. Using the AWS Organizations management account, open the AWS RAM console at https://console.aws.amazon.com/ram/.
2. In the left navigation pane, choose **Settings**, choose **Enable sharing with AWS Organizations**, and then choose **Save settings**.

You can now share an IPAM pool with other members of the organization.

Share an IPAM pool using AWS RAM

In this section you'll share the regional pool with another AWS Organizations member account. For complete instructions on sharing IPAM pools, including information on the required IAM permissions, see Share an IPAM pool using AWS RAM. If you are using the AWS CLI to enable resource sharing, use the `--profile` **`ipam-account`** option.

To share an IPAM pool using AWS RAM

1. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. Choose the private scope, choose the IPAM pool, and choose **Actions** > **View details**.
4. Under **Resource sharing**, choose **Create resource share**. The AWS RAM console opens. You share the pool using AWS RAM.
5. Choose **Create a resource share**.
6. In the AWS RAM console, choose **Create a resource share** again.
7. Add a **Name** for the shared pool.
8. Under **Select resource type**, choose **IPAM pools,** and then choose the ARN of the pool you want to share.
9. Choose **Next**.
10. Choose the **AWSRAMPermissionIpamPoolByoipCidrImport** permission. The details of the permission options are out of scope for this tutorial, but you can find out more about these options in Share an IPAM pool using AWS RAM.
11. Choose **Next**.
12. Under **Principals** > **Select principal type**, choose **AWS account** and enter the account ID of the account that will be bringing an IP address range to IPAM and choose **Add** .
13. Choose **Next**.
14. Review the resource share options and the principals that you'll be sharing with, and then choose **Create**.
15. To allow the **`member-account`** account to allocate IP address CIDRS from the IPAM pool, create a second resource share with `AWSRAMDefaultPermissionsIpamPool`. The value

for `--resource-arns` is the ARN of the IPAM pool that you created in the previous section. The value for `--principals` is the account ID of the **`member-account`**. The value for `--permission-arns` is the ARN of the `AWSRAMDefaultPermissionsIpamPool` permission.

Step 4: Create a VPC

Complete the steps in Create 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 5: 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.

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 6: 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 Delete 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 shares 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 shares 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 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 to delete the Regional pool and then the top-level pool, in that order.

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

Bringing Your Own IP (BYOIP) to IPAM allows you to use your organization's existing IPv4 and IPv6 address ranges in AWS. This enables you to maintain consistent branding, improve network performance, enhance security, and simplify management by unifying on-premises and cloud environments under your own IP address space.

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

Note

Before you begin, you must have first verified domain control.

Once you bring an IPv4 address range to AWS, you can use all of the IP addresses in the range, including the first address (the network address) and the last address (the broadcast address).

Contents

- Bring your own public IPv4 CIDR to IPAM using only the AWS CLI
- Bring your own IPv6 CIDR to IPAM using only the AWS CLI

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 accounts in an AWS Organization.
  - Create an IPAM.
- 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 accounts in an AWS Organization. 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 and IAM roles
- Step 2: Create an IPAM
- Step 3: Create a top-level IPAM pool

- Step 4: Provision a CIDR to the top-level pool
- Step 5: Create a Regional pool within the top-level pool
- Step 6: Provision a CIDR to the Regional pool
- Step 7: Advertise the CIDR
- Step 8: Share the Regional pool
- Step 9: Allocate an Elastic IP address from the pool
- Step 10: Associate the Elastic IP address with an EC2 instance
- Step 11: Cleanup
- Alternative to Step 9

Step 1: Create AWS CLI named profiles and IAM roles

To complete this tutorial as a single AWS user, you can use AWS CLI named profiles to switch from one IAM role to another. Named profiles are collections of settings and credentials that you refer to when using the `--profile` option with the AWS CLI. For more information about how to create IAM roles and named profiles for AWS accounts, see Using an IAM role in the AWS CLI.

Create one role and one named profile 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.

After you have created the IAM roles and 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.

Note

- If you verified your domain control with an X.509 certificate, you must include the CIDR and the BYOIP message and certificate signature that you created in that step so we can verify that you control the public space.
- If you verified your domain control with a DNS TXT record, you must include the CIDR and IPAM verification token that you created in that step so we can verify that you control the public space.

You only need to verify domain control when you provision the BYOIP CIDR to the top-level pool. For the Regional pool within the top-level pool, you can omit the domain ownership verification option.

This step must be done by the IPAM account.

> **Important**
>
> You only need to verify domain control when you provision the BYOIP CIDR to the top-level pool. For the Regional pool within the top-level pool, you can omit the domain control 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. To provision the CIDR with certificate information, use the following command example. In addition to replacing the values as needed in the example, ensure that you replace `Message` and `Signature` values with the `text_message` and `signed_message` values that you got in Verify your domain with an X.509 certificate.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-
pool-id ipam-pool-0a03d430ca3f5c035 --cidr 130.137.245.0/24 --
verification-method remarks-x509 --cidr-authorization-context
 Message="1|aws|470889052444|130.137.245.0/24|20250101|SHA256|
RSAPSS",Signature="W3gdQ9PZHLjPmrnGM~cvGx~KCIsMaU0P7ENO7VRnfSuf9NuJU5RUveQzus~QmF~Nx42j3z7d
hApR89Kt6GxRYOdRaNx8yt-uoZWzxct2yIhWngy-
du9pnEHBOX6WhoGYjWszPw0iV4cmaAX9DuMs8ASR83K127VvcBcRXElT5URr3gWEB1CQe3rmuyQk~gAdbXiDN-94-
oS9AZlafBbrFxRjFWRCTJhc7Cg3ASbRO-VWNci-
C~bWAPczbX3wPQSjtWGV3k1bGuD26ohUc02o8oJZQyYXRpgqcWGVJdQ__" --profile ipam-account
```

   To provision the CIDR with verification token information, use the following command example. In addition to replacing the values as needed in the example, ensure that you replace `ipam-ext-res-ver-token-0309ce7f67a768cf0` with the `IpamExternalResourceVerificationTokenId` token ID that you got in Verify your domain with a DNS TXT record.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-0a03d430ca3f5c035 --cidr 130.137.245.0/24 --verification-method dns-
```

```
token --ipam-external-resource-verification-token-id ipam-ext-res-ver-
token-0309ce7f67a768cf0 --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.

> **Important**
>
> While most provisioning will be completed within two hours, it may take up to one week to complete the provisioning process for publicly advertisable ranges.

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.

The locale for the pool should be one of the following:

- An AWS Region where you want this IPAM pool to be available for allocations.
- The network border group for an AWS Local Zone where you want this IPAM pool to be available for allocations (supported Local Zones). This option is only available for IPAM IPv4 pools in the public scope.
- An AWS Dedicated Local Zone. To create a pool within an AWS Dedicated Local Zone, enter the AWS Dedicated Local Zone in the selector input.

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. If the home Region of the IPAM is unavailable due to an outage and the pool has a locale different than the home Region of the IPAM, the pool can still be used to allocate IP addresses.

When you run the commands in this section, the value for `--region` must include the `--locale` option you entered when you created the pool that will be used for the BYOIP CIDR. For example, if you created the BYOIP pool with a locale of *us-east-1*, the `--region` should be *us-east-1*. If you created the BYOIP pool with a locale of *us-east-1-scl-1* (a network border group used for Local Zones), the `--region` should be *us-east-1* because that Region manages the locale *us-east-1-scl-1*.

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.

> **Important**
>
> When you create the pool, you must include `--aws-service ec2`. The service you select determines the AWS service where the CIDR will be advertisable. Currently, the only option is `ec2`, which means that the CIDRs allocated from this pool will be advertisable for the Amazon EC2 service (for Elastic IP addresses) and the Amazon VPC service (for CIDRs associated with VPCs).

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**
>
> When provisioning a CIDR to a Regional pool within the top-level pool, the most specific IPv4 CIDR you can provision is /24; more specific CIDRs (such as /25) are not permitted. After you create the Regional pool, you can create smaller pools (such as /25) within the same Regional pool. Note that if you share the Regional pool or pools within it, these pools can only be used in the locale set on the same Regional pool.

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: 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.

> **Note**
>
> The advertisement status doesn't not restrict your ability to allocate Elastic IP addresses. Even if your BYOIPv4 CIDR is not advertised, you can still can create EIPs from the IPAM pool.

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 8: Share the Regional pool

Follow the steps in this section to share the IPAM pool using AWS Resource Access Manager (RAM).

Enable resource sharing in AWS RAM

After you create your IPAM, you'll want to share the regional pool with other accounts in your organization. Before you share an IPAM pool, complete the steps in this section to enable resource sharing with AWS RAM. If you are using the AWS CLI to enable resource sharing, use the `--profile` **`management-account`** option.

To enable resource sharing

1. Using the AWS Organizations management account, open the AWS RAM console at https://console.aws.amazon.com/ram/.
2. In the left navigation pane, choose **Settings**, choose **Enable sharing with AWS Organizations**, and then choose **Save settings**.

You can now share an IPAM pool with other members of the organization.

Share an IPAM pool using AWS RAM

In this section you'll share the regional pool with another AWS Organizations member account. For complete instructions on sharing IPAM pools, including information on the required IAM permissions, see Share an IPAM pool using AWS RAM. If you are using the AWS CLI to enable resource sharing, use the `--profile` **`ipam-account`** option.

To share an IPAM pool using AWS RAM

1. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. Choose the private scope, choose the IPAM pool, and choose **Actions** > **View details**.
4. Under **Resource sharing**, choose **Create resource share**. The AWS RAM console opens. You share the pool using AWS RAM.

5. Choose **Create a resource share**.
6. In the AWS RAM console, choose **Create a resource share** again.
7. Add a **Name** for the shared pool.
8. Under **Select resource type**, choose **IPAM pools,** and then choose the ARN of the pool you want to share.
9. Choose **Next**.
10. Choose the **AWSRAMPermissionIpamPoolByoipCidrImport** permission. The details of the permission options are out of scope for this tutorial, but you can find out more about these options in Share an IPAM pool using AWS RAM.
11. Choose **Next**.
12. Under **Principals** > **Select principal type**, choose **AWS account** and enter the account ID of the account that will be bringing an IP address range to IPAM and choose **Add** .
13. Choose **Next**.
14. Review the resource share options and the principals that you'll be sharing with, and then choose **Create**.
15. To allow the **`member-account`** account to allocate IP address CIDRS from the IPAM pool, create a second resource share with `AWSRAMDefaultPermissionsIpamPool`. The value for `--resource-arns` is the ARN of the IPAM pool that you created in the previous section. The value for `--principals` is the account ID of the **`member-account`**. The value for `--permission-arns` is the ARN of the `AWSRAMDefaultPermissionsIpamPool` permission.

Step 9: Allocate an Elastic IP address from the pool

Complete the steps in this section to allocate an Elastic IP address from the pool. Note that if you are using public IPv4 pools to allocate Elastic IP addresses, you can use the alternative steps in Alternative to Step 9 rather than the steps in this section.

> **Important**
>
> If you see an error related to not having permissions to call ec2:AllocateAddress, the managed permission currently assigned to the IPAM pool that was shared with you needs to be updated. Contact the person who created the resource share and ask them to update the managed permission `AWSRAMPermissionIpamResourceDiscovery` to the default version. For more information, see Update a resource share in the *AWS RAM User Guide* .

AWS Management Console

Follow the steps in Allocate an Elastic IP address in the *Amazon EC2 User Guide* to allocate the address, but note the following:

- This step must be done by the member account.
- Ensure that the AWS Region you are in in the EC2 console matches the Locale option you chose when you created the Regional pool.
- When you choose the address pool, choose the option to **Allocate using an IPv4 IPAM pool** and choose the Regional pool you created.

Command line

Allocate an address from the pool with the allocate-address command. The `--region` you use must match the `-locale` option you chose when you created the pool in Step 2. Include the ID of the IPAM pool you created in Step 2 in `--ipam-pool-id`. Optionally, you can also choose a specific `/32` in your IPAM pool by using the `--address` option.

```
aws ec2 allocate-address --region us-east-1 --ipam-pool-id ipam-
pool-07ccc86aa41bef7ce
```

Example response:

```
{
    "PublicIp": "18.97.0.41",
    "AllocationId": "eipalloc-056cdd6019c0f4b46",
    "PublicIpv4Pool": "ipam-pool-07ccc86aa41bef7ce",
    "NetworkBorderGroup": "us-east-1",
    "Domain": "vpc"
}
```

For more information, see Allocate an Elastic IP address in the *Amazon EC2 User Guide*.

Step 10: Associate the Elastic IP address with an EC2 instance

Complete the steps in this section to associate the Elastic IP address with an EC2 instance.

AWS Management Console

Follow the steps in Associate an Elastic IP address in the *Amazon EC2 User Guide* to allocate an Elastic IP address from the IPAM pool, but note the following: When you use AWS Management Console option, the AWS Region you associate the Elastic IP address in must match the Locale option you chose when you created the Regional pool.

This step must be done by the member account.

Command line

This step must be done by the member account. Use the `--profile` **`member-account`** option.

Associate the Elastic IP address with an instance with the associate-address command. The `--region` you associate the Elastic IP address in must match the `--locale` option you chose when you created the Regional pool.

```
aws ec2 associate-address --region us-east-1 --instance-id i-07459a6fca5b35823 --
public-ip 18.97.0.41
```

Example response:

```
{
    "AssociationId": "eipassoc-06aa85073d3936e0e"
}
```

For more information, see Associate an Elastic IP address with an instance or network interface in the *Amazon EC2 User Guide*.

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 include 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 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 include 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": []
}
```

5. 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"

    }

}
```

6. Delete the RAM shares 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 shares 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 shares and disable RAM integration, use the `--profile` **`ipam-account`** and `--profile` **`management-account`** options.

7. 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"
    }
}
```


8. 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"

    }

}
```

9. 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"
    }
}
```

10. 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"
            }
        ],
```

```
    }
}
```

Alternative to Step 9

If you are using public IPv4 pools to allocate Elastic IP addresses, you can use the steps in this section rather than the steps in Step 9: Allocate an Elastic IP address from the pool.

Contents

- Step 1: Create a public IPv4 pool
- Step 2: Provision the public IPv4 CIDR to your public IPv4 pool
- Step 3: Create an Elastic IP address from the public IPv4 pool
- Alternative to Step 9 cleanup

Step 1: Create a public IPv4 pool

This step would typically be done by a different AWS account which wants to provision an Elastic IP address, such as 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 2: 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. The least specific `--netmask-length` you can define is 24.

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 3: 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"
        }
    ]
}
```

Alternative to Step 9 cleanup

Complete these steps to clean up public IPv4 pools created with the alternative to Step 9. You should complete these steps after you release the Elastic IP address during the standard cleanup process in Step 10: Cleanup.

1. 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": []
        }
    ]
}
```

2. 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"

    ]

}
```

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.

   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": []
           }
       ]
   }
   ```

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.

If you do not need to advertise your IPv6 addresses over the Internet, you can provision a private GUA IPv6 address to an IPAM. For more information, see Enable provisioning private IPv6 GUA CIDRs.

Important

- This tutorial assumes you have already completed the steps in the following sections:
  - Integrate IPAM with accounts in an AWS Organization.
  - Create an IPAM.
- 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 accounts in an AWS Organization. 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 and IAM roles
- Step 2: Create an IPAM
- Step 3: Create an IPAM pool
- Step 4: Provision a CIDR to the top-level pool
- Step 5: Create a Regional pool within the top-level pool
- Step 6: Provision a CIDR to the Regional pool
- Step 7. Share the Regional pool
- Step 8: Create a VPC using the IPv6 CIDR
- Step 9: Advertise the CIDR
- Step 10: Cleanup

Step 1: Create AWS CLI named profiles and IAM roles

To complete this tutorial as a single AWS user, you can use AWS CLI named profiles to switch from one IAM role to another. Named profiles are collections of settings and credentials that you refer to when using the `--profile` option with the AWS CLI. For more information about how to create IAM roles and named profiles for AWS accounts, see Using an IAM role in the AWS CLI.

Create one role and one named profile 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.

After you have created the IAM roles and 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 most specific IPv6 address range that you can bring is /48 for CIDRs that are publicly advertisable and /60 for CIDRs that are not publicly advertisable.

> **Note**
>
> - If you verified your domain control with an X.509 certificate, you must include the CIDR and the BYOIP message and certificate signature that you created in that step so we can verify that you control the public space.

- If you verified your domain control with a DNS TXT record, you must include the CIDR and IPAM verification token that you created in that step so we can verify that you control the public space.

You only need to verify domain control when you provision the BYOIP CIDR to the top-level pool. For the Regional pool within the top-level pool, you can omit the domain ownership option.

This step must be done by the IPAM account.

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

1. To provision the CIDR with certificate information, use the following command example. In addition to replacing the values as needed in the example, ensure that you replace `Message` and `Signature` values with the `text_message` and `signed_message` values that you got in Verify your domain with an X.509 certificate.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-07f2466c7158b50c4 --cidr 2605:9cc0:409::/48 --verification-method remarks-
x509 --cidr-authorization-context Message="1|aws|470889052444|2605:9cc0:409::/48|
20250101|SHA256|RSAPSS",Signature="FU26~vRG~NUGXa~akxd6dvdcCfvL88g8d~YAuai-
CR7HqMwzcgdS9RlpBGtfIdsRGyr77LmWyWqU9Xp1g2R1kSkfD00NiLKLcv9F63k6wdEkyFxNp7RAJDvF1mBwxmSgH~C
Vp6LON3yOOXMp4JENB9uM7sMlu6oeoutGyyhXFeYPzlGSRdcdfKNKaimvPCqVsxGN5AwSilKQ8byNqoa~G3dvs8ueSa
wispI~r69fq515UR19TA~fmmxBDh1huQ8DkM1rqcwveWow__" --profile ipam-account
```

To provision the CIDR with verification token information, use the following command example. In addition to replacing the values as needed in the example, ensure that you replace `ipam-ext-res-ver-token-0309ce7f67a768cf0` with the `IpamExternalResourceVerificationTokenId` token ID that you got in Verify your domain with a DNS TXT record.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-
pool-07f2466c7158b50c4 --cidr 2605:9cc0:409::/48 --verification-method
 dns-token --ipam-external-resource-verification-token-id ipam-ext-res-ver-
token-0309ce7f67a768cf0 --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.

> **Important**
>
> While most provisioning will be completed within two hours, it may take up to one week to complete the provisioning process for publicly advertisable ranges.

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.

> **Important**
>
> When you create the pool, you must include `--aws-service ec2`. The service you select determines the AWS service where the CIDR will be advertisable. Currently, the only option is `ec2`, which means that the CIDRs allocated from this pool will be advertisable for the Amazon EC2 service and the Amazon VPC service (for CIDRs associated with VPCs).

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 most specific IPv6 address range that you can bring is /48 for CIDRs that are publicly advertisable and /60 for CIDRs that are not publicly advertisable.

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. Share the Regional pool

Follow the steps in this section to share the IPAM pool using AWS Resource Access Manager (RAM).

Enable resource sharing in AWS RAM

After you create your IPAM, you'll want to share the regional pool with other accounts in your organization. Before you share an IPAM pool, complete the steps in this section to enable resource sharing with AWS RAM. If you are using the AWS CLI to enable resource sharing, use the `--profile` **`management-account`** option.

To enable resource sharing

1. Using the AWS Organizations management account, open the AWS RAM console at https://console.aws.amazon.com/ram/.
2. In the left navigation pane, choose **Settings**, choose **Enable sharing with AWS Organizations**, and then choose **Save settings**.

You can now share an IPAM pool with other members of the organization.

Share an IPAM pool using AWS RAM

In this section you'll share the regional pool with another AWS Organizations member account. For complete instructions on sharing IPAM pools, including information on the required IAM permissions, see Share an IPAM pool using AWS RAM. If you are using the AWS CLI to enable resource sharing, use the `--profile` **`ipam-account`** option.

To share an IPAM pool using AWS RAM

1. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.

3. Choose the private scope, choose the IPAM pool, and choose **Actions** > **View details**.
4. Under **Resource sharing**, choose **Create resource share**. The AWS RAM console opens. You share the pool using AWS RAM.
5. Choose **Create a resource share**.
6. In the AWS RAM console, choose **Create a resource share** again.
7. Add a **Name** for the shared pool.
8. Under **Select resource type**, choose **IPAM pools,** and then choose the ARN of the pool you want to share.
9. Choose **Next**.
10. Choose the **AWSRAMPermissionIpamPoolByoipCidrImport** permission. The details of the permission options are out of scope for this tutorial, but you can find out more about these options in Share an IPAM pool using AWS RAM.
11. Choose **Next**.
12. Under **Principals** > **Select principal type**, choose **AWS account** and enter the account ID of the account that will be bringing an IP address range to IPAM and choose **Add** .
13. Choose **Next**.
14. Review the resource share options and the principals that you'll be sharing with, and then choose **Create**.
15. To allow the **`member-account`** account to allocate IP address CIDRS from the IPAM pool, create a second resource share with `AWSRAMDefaultPermissionsIpamPool`. The value for `--resource-arns` is the ARN of the IPAM pool that you created in the previous section. The value for `--principals` is the account ID of the **`member-account`**. The value for `--permission-arns` is the ARN of the `AWSRAMDefaultPermissionsIpamPool` permission.

Step 8: 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 9: 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 10: 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 shares 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 shares 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 shares 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 a BYOIP IPv4 CIDR 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.

This tutorial assumes you have already successfully brought an IP address range to AWS using the process described in Bring your own IP addresses (BYOIP) in Amazon EC2 and now you want to

transfer that IP address range to IPAM. If you are bringing a new IP address to AWS for the first time, complete the steps in Tutorial: Bring your IP addresses to IPAM.

If you transfer a public IPv4 pool to IPAM, there is no impact on existing allocations. Once you transfer a public IPv4 pool to IPAM, depending on the resource type, you may be able to monitor the existing allocations. For more information, see Monitor CIDR usage by resource.

Note

- This tutorial assumes you have already completed the steps in Create an IPAM.
- 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 owns the BYOIP CIDR. In this tutorial, this account will be called the BYOIP CIDR owner account.

Contents

- Step 1: Create AWS CLI named profiles and IAM roles
- Step 2: Get your IPAM's public scope ID
- Step 3: Create an IPAM pool
- Step 4: Share the IPAM pool using AWS RAM
- Step 5: Transfer an existing BYOIP IPV4 CIDR to IPAM
- Step 6: View the CIDR in IPAM
- Step 7: Cleanup

Step 1: Create AWS CLI named profiles and IAM roles

To complete this tutorial as a single AWS user, you can use AWS CLI named profiles to switch from one IAM role to another. Named profiles are collections of settings and credentials that you refer to when using the `--profile` option with the AWS CLI. For more information about how to create IAM roles and named profiles for AWS accounts, see Using an IAM role in the AWS CLI.

Create one role and one named profile for each of the three 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.

After you have created the IAM roles and 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`** 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`** account. The IPAM pool you create must be a top-level pool with the `--locale` option matching the BYOIP CIDR AWS Region. You can only transfer a BYOIP to a top-level IPAM pool.

> **Important**
>
> When you create the pool, you must include `--aws-service ec2`. The service you select determines the AWS service where the CIDR will be advertisable. Currently, the only option is `ec2`, which means that the CIDRs allocated from this pool will be advertisable for the Amazon EC2 service (for Elastic IP addresses) and the Amazon VPC service (for CIDRs associated with VPCs).

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 --profile ipam-account --ipam-scope-
   id ipam-scope-0087d83896280b594 --description "top-level-pool" --locale us-west-2
    --aws-service ec2 --address-family ipv4
   ```

   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: Share the IPAM pool using AWS RAM

Follow the steps in this section to share an IPAM pool using AWS RAM so that another AWS account can transfer an existing BYOIP IPV4 CIDR to the IPAM pool and use the IPAM pool. This step should be performed by the **ipam-account** account.

To share an IPv4 address pool using the AWS CLI

1. View the AWS RAM permissions available for IPAM pools. You need both ARNs to complete the steps in this section.

```
aws ram list-permissions --region us-east-1 --profile ipam-account --resource-type
 ec2:IpamPool
```

```
{
    "permissions": [
        {
           "arn": "arn:aws:ram::aws:permission/AWSRAMDefaultPermissionsIpamPool",
           "version": "1",
           "defaultVersion": true,
           "name": "AWSRAMDefaultPermissionsIpamPool",
           "resourceType": "ec2:IpamPool",
           "status": "ATTACHABLE",
           "creationTime": "2022-06-30T13:04:29.335000-07:00",
           "lastUpdatedTime": "2022-06-30T13:04:29.335000-07:00",
           "isResourceTypeDefault": true
        },
        {
            "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionIpamPoolByoipCidrImport",
            "version": "1",
            "defaultVersion": true,
            "name": "AWSRAMPermissionIpamPoolByoipCidrImport",
            "resourceType": "ec2:IpamPool",
            "status": "ATTACHABLE",
            "creationTime": "2022-06-30T13:03:55.032000-07:00",
            "lastUpdatedTime": "2022-06-30T13:03:55.032000-07:00",
            "isResourceTypeDefault": false
        }
    ]
}
```

2. Create a resource share to enable the **byoip-owner-account** account to import BYOIP CIDRs to IPAM. The value for `--resource-arns` is the ARN of the IPAM pool that you created in the previous section. The value for `--principals` is the account ID of the BYOIP CIDR owner account. The value for `--permission-arns` is the ARN of the `AWSRAMPermissionIpamPoolByoipCidrImport` permission.

```
aws ram create-resource-share --region us-east-1 --profile ipam-account
 --name PoolShare2 --resource-arns arn:aws:ec2::123456789012:ipam-pool/
ipam-pool-0a03d430ca3f5c035 --principals 111122223333 --permission-arns
 arn:aws:ram::aws:permission/AWSRAMPermissionIpamPoolByoipCidrImport
```

```
{

    "resourceShare": {

        "resourceShareArn": "arn:aws:ram:us-east-1:123456789012:resource-
share/7993758c-a4ea-43ad-be12-b3abaffe361a",
        "name": "PoolShare2",

        "owningAccountId": "123456789012",

        "allowExternalPrincipals": true,

        "status": "ACTIVE",

        "creationTime": "2023-04-28T07:32:25.536000-07:00",

        "lastUpdatedTime": "2023-04-28T07:32:25.536000-07:00"

        }

}
```

3. (Optional) If you want to allow the **byoip-owner-account** account to allocate IP address CIDRS from the IPAM pool to public IPv4 pools after the transfer is complete, copy the ARN for `AWSRAMDefaultPermissionsIpamPool` and create a second resource share. The value for `--resource-arns` is the ARN of the IPAM pool that you created in the previous section. The value for `--principals` is the account ID of the BYOIP CIDR owner account. The value for `--permission-arns` is the ARN of the `AWSRAMDefaultPermissionsIpamPool` permission.

```
aws ram create-resource-share --region us-east-1 --profile ipam-account
 --name PoolShare1 --resource-arns arn:aws:ec2::123456789012:ipam-pool/
ipam-pool-0a03d430ca3f5c035 --principals 111122223333 --permission-arns
 arn:aws:ram::aws:permission/AWSRAMDefaultPermissionsIpamPool
```


```
{

    "resourceShare": {

        "resourceShareArn": "arn:aws:ram:us-east-1:123456789012:resource-
share/8d1e229b-2830-4cf4-8b10-19c889235a2f",
        "name": "PoolShare1",

        "owningAccountId": "123456789012",

        "allowExternalPrincipals": true,

        "status": "ACTIVE",

        "creationTime": "2023-04-28T07:31:25.536000-07:00",

        "lastUpdatedTime": "2023-04-28T07:31:25.536000-07:00"

        }

}
```


As a result of creating the resource share in RAM, the byoip-owner-account account can now move CIDRs to IPAM.

Step 5: 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-owner-account`** account.

Important

Once you bring an IPv4 address range to AWS, you can use all of the IP addresses in the range, including the first address (the network address) and the last address (the broadcast address).

To transfer the BYOIP CIDR to IPAM, the BYOIP CIDR owner must have these permissions in their IAM policy:

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

Note

You can use either the AWS Management Console or the AWS CLI for this step.

AWS Management Console

To transfer a BYOIP CIDR to the IPAM pool:

1. Open the IPAM console at https://console.aws.amazon.com/ipam/ as the **`byoip-owner-account`** account.
2. In the navigation pane, choose **Pools**.
3. Choose the top-level pool created and shared in this tutorial.
4. Choose **Actions** > **Transfer BYOIP CIDR**.
5. Choose **Transfer BYOIP CIDR**.
6. Choose your BYOIP CIDR.
7. Choose **Provision**.

Command line

Use the following AWS CLI commands 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 --profile byoip-owner-account
 --ipam-pool-id ipam-pool-0a03d430ca3f5c035 --ipam-pool-owner 123456789012 --
cidr 130.137.249.0/24
```

   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  --profile byoip-
owner-account --ipam-pool-id ipam-pool-0a03d430ca3f5c035 --ipam-pool-
owner 123456789012 --cidr 130.137.249.0/24
```

   The following example output shows the state.

```
{
    "ByoipCidr": {

        "Cidr": "130.137.249.0/24",

        "State": "complete-transfer"

    }

}
```

Step 6: 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** 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  --profile ipam-account --
ipam-pool-id ipam-pool-0d8f3646b61ca5987
```

  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": "111122223333"
        }
    ]
}
```

Step 7: 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** account.

To cleanup the resources created in this tutorial using the AWS CLI

1. To delete the IPAM pool shared resource, run the following command to get the first resource share ARN:

```
aws ram get-resource-shares --region us-east-1 --profile ipam-account --
name PoolShare1 --resource-owner SELF
```

```
{
    "resourceShares": [
        {
            "resourceShareArn": "arn:aws:ram:us-east-1:123456789012:resource-
share/8d1e229b-2830-4cf4-8b10-19c889235a2f",
            "name": "PoolShare1",
            "owningAccountId": "123456789012",
            "allowExternalPrincipals": true,
            "status": "ACTIVE",
            "creationTime": "2023-04-28T07:31:25.536000-07:00",
            "lastUpdatedTime": "2023-04-28T07:31:25.536000-07:00",
            "featureSet": "STANDARD"
        }
    ]
}
```

2. Copy the resource share ARN and use it to delete the IPAM pool resource share.

```
aws ram delete-resource-share --region us-east-1 --profile ipam-account
 --resource-share-arn arn:aws:ram:us-east-1:123456789012:resource-
share/8d1e229b-2830-4cf4-8b10-19c889235a2f
```

```
{
    "returnValue": true
}
```

3. If you created an additional resource share in Step 4: Share the IPAM pool using AWS RAM, repeat the previous two steps to get the second resource share ARN for `PoolShare2` and delete the second resource share.

4. 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  --profile ipam-account --
ipam-pool-id ipam-pool-0d8f3646b61ca5987
```

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": "111122223333"
        }
    ]
}
```

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 **`byoip-owner-account`** account.

```
aws ec2 deprovision-public-ipv4-pool-cidr --region us-east-1  --profile byoip-
owner-account --pool-id ipv4pool-ec2-0019eed22a684e0b3 --cidr 130.137.249.255/32
```

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

```
{
    "PoolId": "ipv4pool-ec2-0019eed22a684e0b3",

    "DeprovisionedAddresses": [

        "130.137.249.255"

    ]

}
```

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 **byoip-owner-account** account.

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

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

```
{
    "PublicIpv4Pools": [
        {
            "PoolId": "ipv4pool-ec2-0019eed22a684e0b3",
            "Description": "",
            "PoolAddressRanges": [],
            "TotalAddressCount": 0,
            "TotalAvailableAddressCount": 0,
            "NetworkBorderGroup": "us-east-1",
            "Tags": []
        }
    ]
}
```

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

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

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"
    }
}
```

Tutorial: Plan VPC IP address space for subnet IP allocations

Complete this tutorial to plan the VPC IP address space for allocating IP addresses to VPC subnets and monitor IP address-related metrics at the subnet and VPC level.

> **Note**
>
> This tutorial covers allocating private IPv4 address space in a private IPAM scope to VPCs and subnets. You can also complete this tutorial using an IPv6 CIDR range by creating the VPC with an Amazon-provided IPv6 CIDR block option on the VPC console.

Planning VPC IP address space for subnets enables you to do the following:

- **Plan and organize your VPC's IP addresses for allocation to subnets**: You can divide VPC IP address space into smaller CIDR blocks and provision those CIDR blocks to subnets with different business needs, such as if you're running workloads in development or production subnets.
- **Simplify IP address allocations for VPC subnets**: Once your VPC's address space is planned and organized, you can choose a netmask length rather than manually inputting a CIDR. For example, if a developer is creating a subnet for hosting development workloads, they need to choose a pool and a netmask length for the subnet and IPAM will automatically allocate the CIDR block to your subnet.

The following example shows the hierarchy of the pool and resource structure that you will create with this tutorial:

- Private scope
  - Resource planning pool (10.0.0.0/20)
    - Dev subnet pool (10.0.0.0/24)
      - Dev subnet (10.0.0.0/28)

- Prod subnet pool (10.0.0.1/24)
  - Prod subnet (10.0.0.16/28)

Important

- The resource planning pool can be used to allocate CIDRs to subnets or it can be used as a source pool in which you can create other pools. In this tutorial, we use the resource planning pool as a source pool for subnet pools.
- You can create multiple resource planning pools using the same VPC if the VPC has more than one CIDR provisioned to it; if a VPC has two CIDRs assigned to it, for example, you can create two resource planning pools, one from each CIDR. Each CIDR can be assigned to one pool at a time.

Step 1: Create a VPC

Complete the steps in this section to create a VPC to be used for subnet IP address planning. For more information about the IAM permissions that are required to create VPCs, see Amazon VPC policy examples in the *Amazon VPC User Guide*.

Note

You can use an existing VPC rather than creating a new one, but this tutorial focuses on the scenario where the VPC is configured with a manually-allocated CIDR block, not an IPAM-allocated automatically CIDR block.

To create a VPC

1. Using the IPAM admin account, open the VPC console at https://console.aws.amazon.com/vpc/.
2. Choose **Create VPC**.
3. Enter a name for the VPC, such as tutorial-vpc.
4. Choose **IPv4 CIDR manual input** and enter an IPv4 CIDR block. In this tutorial, we use 10.0.0.0/20.

5. Skip the option to add an IPv6 CIDR block.
6. Choose **Create VPC**.
7. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.
8. Choose **Resources** in the left navigation pane.
9. Wait for the VPC that you created to appear. This takes some time to happen and you may need to refresh the window to see it appear. The VPC must be discovered by IPAM before you continue to the next step.

Step 2: Create a resource planning pool

Complete the steps in this section to create a resource planning pool.

To create a resource planning pool

1. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. Choose the private scope.
4. Choose **Create pool**.
5. Under **IPAM scope**, leave the private scope selected.
6. (Optional) Add a **Name tag** for the pool, such as “Resource-planning-pool”.
7. Under **Source**, choose **IPAM scope**.
8. Under **Resource planning**, choose **Plan IP space within a VPC** and choose the VPC you created in the previous step. The VPC is the resource used to provision CIDRs to the resource planning pool.
9. Under **CIDRs to provision**, choose the VPC CIDR to provision for the resource pool. The CIDR you provision to the resource planning pool must match the CIDR provisioned to the VPC. In this tutorial, we use 10.0.0.0/20.
10. Choose **Create pool**.
11. Once the pool is created, choose the **CIDR** tab to see the state of the provisioned CIDR. Refresh the page and wait for the CIDR state to change from *Pending-provision* to *Provisioned* before you go to the next step.

Step 3: Create subnet pools

Complete the steps in this section to create two subnet pools that will be used for allocating IP space to subnets.

To create subnet pools

1. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.
2. In the navigation pane, choose **Pools**.
3. Choose the private scope.
4. Choose **Create pool**.
5. Under **IPAM scope**, leave the private scope selected.
6. (Optional) Add a **Name tag** for the pool, such as “dev-subnet-pool”.
7. Under **Source**, choose **IPAM pool** and select the resource planning pool you created in Step 3. The address family, Resource planning configuration, and Locale are automatically inherited from the source pool.
8. Under **CIDRs to provision**, choose the CIDR to provision for the subnet pool. In this tutorial, we use 10.0.0.0/24.
9. Choose **Create pool**.
10. Once the pool is created, choose the **CIDR tab** to see the state of the provisioned CIDR. Refresh the page and wait for the CIDR state to change from *Pending-provision* to *Provisioned* before you go to the next step.
11. Repeat this process to create another subnet called “prod-subnet-pool”.

At this point, if you want to make this subnet pool available to other AWS accounts, you can share the subnet pool. For instructions on how to do that, see Share an IPAM pool using AWS RAM. Then return here to complete the tutorial.

Step 4: Create subnets

Complete these steps to create two subnets.

To create subnets

1. Using the appropriate account, open the VPC console at https://console.aws.amazon.com/vpc/.
2. Choose **Subnets** > **Create subnet**.
3. Choose the VPC you created at the start of this tutorial.
4. Enter a name for the subnet, such as "tutorial-subnet".
5. (optional) Choose an **Availability Zone**.
6. Under **IPv4 CIDR block**, choose **IPAM-allocated IPV4 CIDR block** and choose the dev subnet pool and a /28 netmask.
7. Choose **Create subnet**.
8. Repeat this process to create another subnet. This time choose the prod subnet pool and a /28 netmask.
9. Return to the IPAM console and choose **Resources** in the left navigation pane.
10. Look for the subnet pools you created and wait for the subnets that you created to appear beneath it. This takes some time to happen and you may need to refresh the window to see it appear.

The tutorial is complete. You can create additional subnet pools as needed or you can launch in EC2 instance into one of the subnets.

IPAM publishes metrics related to IP address usage in subnets. You can set CloudWatch alarms on the SubnetIPUsage metric, thereby allowing you to take action when IP utilization thresholds are breached. If, for example, you have a /24 CIDR (256 IP addresses) assigned to a subnet and you want to be notified when 80% of the IPs have been utilized, you can set up a CloudWatch alarm to alert you when this threshold is reached. For more information on creating an alarm for subnet IP usage, see Quick tip for creating alarms.

Step 5: Cleanup

Complete these steps to delete the resources you created with this tutorial.

To clean up the resources

1. Using the IPAM admin account, open the IPAM console at https://console.aws.amazon.com/ipam/.

2. In the navigation pane, choose **Pools**.
3. Choose the private scope.
4. Choose the resource planning pool and choose **Action** > **Delete**.
5. Select **Cascade delete**. The resource planning pool and the subnet pools will be deleted. This will not delete the subnets themselves. They will stay with CIDRs provisioned to them, though the CIDRs will no longer be from an IPAM pool.
6. Choose **Delete**.
7. Delete the subnets.
8. Delete the VPC.

Cleanup is complete.

Allocate sequential Elastic IP addresses from an IPAM pool

IPAM enables you to provision Amazon-owned public IPv4 blocks to IPAM pools and allocate sequential Elastic IP addresses from those pools to AWS resources.

Contiguously-allocated Elastic IP addresses are public IPv4 addresses that are allocated sequentially. For example, if Amazon provides you a public IPv4 CIDR block of `192.0.2.0/30` and you allocate the four available public IPv4 addresses from that CIDR block, an example of four sequential Elastic IP addresses is `192.0.2.0`, `192.0.2.1`, `192.0.2.2`, and `192.0.2.3`.

Contiguously-allocated Elastic IP addresses enable you to simplify your security and networking rules in the following ways:

- **Security administration**: Using sequential IPv4 addresses reduces your firewall management overhead. You can add an entire prefix with a single rule and associate IPs from the same prefix as you scale, saving time and effort.
- **Enterprise access**: You can simplify the address space shared with your clients by using an entire CIDR block instead of a long list of individual public IPv4 addresses. This avoids the need to constantly communicate IP changes as your application scales on AWS.
- **Simplified IP management**: Using sequential IPv4 addresses simplifies public IP management for your central networking team, as it reduces the need to track individual public IPs and instead allows them to focus on a limited number of IP prefixes.

In this tutorial, you'll go through the steps required to allocate sequential Elastic IP addresses from an IPAM pool. You'll create an IPAM pool with an Amazon-provided contiguous public IPv4 CIDR block, allocate Elastic IP addresses from the pool, and learn how to monitor IPAM pool allocations.

Note

- There are charges associated with provisioning Amazon-owned public IPv4 CIDR blocks. For more information, see the **Amazon-provided contiguous IPv4 block** tab on the Amazon VPC pricing page.
- This tutorial assumes you want to create an IPAM using IPAM with a single account. If you want to share Amazon-owned contiguous public IPv4 blocks across accounts, first Integrate IPAM with accounts in an AWS Organization and then Share an IPAM pool using AWS RAM. If you integrate with AWS Organizations, you have the option to create a service control policy to prevent deprovisioning of the contig IPv4 blocks assigned to the pool.
- You cannot transfer sequential Elastic IP addresses allocated from an IPAM pool to other AWS accounts. Instead, IPAM allows you to share IPAM pools across AWS accounts by integrating IPAM with AWS Organizations (as mentioned above).
- There are limits on the number of Amazon-owned public IPv4 CIDR blocks you can provision and their size. For more information, see Quotas for your IPAM.

Contents

- Step 1: Create an IPAM
- Step 2: Create an IPAM pool and provision a CIDR
- Step 3: Allocate an Elastic IP address from the pool
- Step 4: Associate the Elastic IP address with an EC2 instance
- Step 5: Track and monitor pool usage
- Cleanup

Step 1: Create an IPAM

Complete the steps in this section to create an IPAM.

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. Create the IPAM in your main Region of operations.
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. Choose an **IPAM tier**. For more information about the features available in each tier and the costs associated with the tiers, see the IPAM tab on the Amazon VPC pricing page.
6. 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.

   > **Note**
   >
   > If you are creating an IPAM in the Free Tier, you can select multiple operating Regions for your IPAM, but the only IPAM feature that will be available across operating Regions is Public IP insights. You cannot use other features in the Free Tier, like BYOIP, across the IPAM's operating Regions. You can only use them in the IPAM's home Region. To use all IPAM features across operating Regions, create an IPAM in the Advanced Tier.

7. Choose **Create IPAM**.

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.

Create the IPAM with the create-ipam command:

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

Example response:

```
{
    "Ipam": {
        "OwnerId": "320805250157",
        "IpamId": "ipam-0755477df834ea06b",
        "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-0755477df834ea06b",
        "IpamRegion": "us-east-1",
        "PublicDefaultScopeId": "ipam-scope-01bc7290e4a9202f9",
        "PrivateDefaultScopeId": "ipam-scope-0a50983b97a7a583a",
        "ScopeCount": 2,
        "OperatingRegions": [
            {
                "RegionName": "us-east-1"
            }
        ],
        "State": "create-in-progress",
        "Tags": [],
        "DefaultResourceDiscoveryId": "ipam-res-disco-02cc5b34cc3f04f09",
        "DefaultResourceDiscoveryAssociationId": "ipam-res-disco-
assoc-06b3a4dccfc81f7c1",
        "ResourceDiscoveryAssociationCount": 1,
        "Tier": "advanced"
    }
}
```

You'll need the PublicDefaultScopeId in the next step. For more information about scopes, see How IPAM works.

Step 2: Create an IPAM pool and provision a CIDR

Complete the steps in this section to create an IPAM pool from which you'll allocate the Elastic IP addresses.

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. Choose the public scope. For more information about scopes, see How IPAM works.
4. Choose **Create pool**.
5. (Optional) Add a **Name tag** for the pool and a **Description** for the pool.
6. Under **Source**, choose **IPAM scope**.
7. Under **Address family**, choose **IPv4**.
8. Under **Resource planning**, leave **Plan IP space within the scope** selected.
9. Under **Locale**, choose the locale for the pool. The locale is the AWS Region where you want this IPAM pool to be available for allocations. The available options come from the operating Regions that you chose when you created your IPAM.
10. Under **Service**, choose **EC2 (EIP/VPC)**. The service you select determines the AWS service where the CIDR will advertised. Currently, the only option is **EC2 (EIP/VPC)**, which means that the CIDRs allocated from this pool will be advertised for the Amazon EC2 service (for Elastic IP addresses).
11. Under **Public IP source**, choose **Amazon-owned**.
12. Under **CIDR to provision**, choose **Add Amazon-owned public CIDR**. Choose a **Netmask** length between `/29` (8 IP addresses) and `/30` (4 IP addresses). You can add up to 2 CIDRs by default. For information about increasing the limits on Amazon-provided contiguous public IPv4 CIDRs, see Quotas for your IPAM.
13. Leave **Configure this pool's allocation rule settings** unselected.
14. (Optional) Choose **Tags** for the pool.
15. 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.

Command line

To create a pool

1. Create an IPAM pool with the create-ipam-pool command. The locale is the AWS Region where you want this IPAM pool to be available for allocations. The available options come from the operating Regions that you chose when you created your IPAM.

```
aws ec2 create-ipam-pool --region us-east-1 --ipam-scope-id ipam-
scope-01bc7290e4a9202f9 --address-family ipv4 --locale us-east-1 --aws-service
 ec2 --public-ip-source amazon
```

Example response with state `create-in-progress`:


```
{

    "IpamPool": {

        "OwnerId": "320805250157",

        "IpamPoolId": "ipam-pool-07ccc86aa41bef7ce",

        "IpamPoolArn": "arn:aws:ec2::320805250157:ipam-pool/ipam-
pool-07ccc86aa41bef7ce",
        "IpamScopeArn": "arn:aws:ec2::320805250157:ipam-scope/ipam-
scope-01bc7290e4a9202f9",
        "IpamScopeType": "public",

        "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-0755477df834ea06b",

        "IpamRegion": "us-east-1",

        "Locale": "us-east-1",

        "PoolDepth": 1,

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

        "AutoImport": false,

        "AddressFamily": "ipv4",
```

```
        "Tags": [],

        "AwsService": "ec2",

        "PublicIpSource": "amazon"

    }

}
```

2. Check that the pool was created successfully with the describe-ipam-pools command.

```
aws ec2 describe-ipam-pools --region us-east-1 --ipam-pool-ids ipam-
pool-07ccc86aa41bef7ce
```

Example response with state `create-complete`:

```
{

    "IpamPools": [
        {
            "OwnerId": "320805250157",
            "IpamPoolId": "ipam-pool-07ccc86aa41bef7ce",
            "IpamPoolArn": "arn:aws:ec2::320805250157:ipam-pool/ipam-
pool-07ccc86aa41bef7ce",
            "IpamScopeArn": "arn:aws:ec2::320805250157:ipam-scope/ipam-
scope-01bc7290e4a9202f9",
            "IpamScopeType": "public",
            "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-0755477df834ea06b",
            "IpamRegion": "us-east-1",
            "Locale": "us-east-1",
            "PoolDepth": 1,
            "State": "create-complete",
            "AutoImport": false,
            "AddressFamily": "ipv4",
            "Tags": [],
            "AwsService": "ec2",
            "PublicIpSource": "amazon"
        }
    ]
}
```

3. Provision a CIDR to the pool with the provision-ipam-pool-cidr command. Choose a `--netmask-length` between `/29` (8 IP addresses) and `/30` (4 IP addresses). You can add up to 2 CIDRs by default. For information about increasing the limits on Amazon-provided contiguous public IPv4 CIDRs, see Quotas for your IPAM.

```
aws ec2 provision-ipam-pool-cidr --region us-east-1 --ipam-pool-id ipam-pool-07ccc86aa41bef7ce --netmask-length 29
```

Example response with state `pending-provision`:

```
{

    "IpamPoolCidr": {
        "State": "pending-provision",
        "IpamPoolCidrId": "ipam-pool-cidr-01856e43994df4913b7bc6aac47adf983",
        "NetmaskLength": 29
    }
}
```

4. Ensure that this CIDR has been provisioned before you continue. You can view the state of provisioning using the get-ipam-pool-cidrs command.

```
aws ec2 get-ipam-pool-cidrs --region us-east-1 --ipam-pool-id ipam-pool-07ccc86aa41bef7ce
```

Example response with state `provisioned`:

```
{

    "IpamPoolCidrs": [
        {
            "Cidr": "18.97.0.40/29",
            "State": "provisioned",
            "IpamPoolCidrId": "ipam-pool-
cidr-01856e43994df4913b7bc6aac47adf983",
            "NetmaskLength": 29
        }
    ]
}
```

Step 3: Allocate an Elastic IP address from the pool

Complete the steps in this section to allocate an Elastic IP address from the pool.

AWS Management Console

Follow the steps in Allocate an Elastic IP address in the *Amazon EC2 User Guide* to allocate the address, but note the following:

- Ensure that the AWS Region you are in in the EC2 console matches the Locale option you chose when you created the pool in Step 2.
- When you choose the address pool, choose the option to **Allocate using an IPv4 IPAM pool** and choose the pool you created in Step 1.

Command line

Allocate an address from the pool with the allocate-address command. The `--region` you use must match the `-locale` option you chose when you created the pool in Step 2. Include the ID of the IPAM pool you created in Step 2 in `--ipam-pool-id`.

```
aws ec2 allocate-address --region us-east-1 --ipam-pool-id ipam-
pool-07ccc86aa41bef7ce
```

Example response:

```
{
    "PublicIp": "18.97.0.41",
    "AllocationId": "eipalloc-056cdd6019c0f4b46",
    "PublicIpv4Pool": "ipam-pool-07ccc86aa41bef7ce",
    "NetworkBorderGroup": "us-east-1",
    "Domain": "vpc"
}
```

Optionally, you can also choose a specific `/32` in your IPAM pool by using the `--address` option.

```
aws ec2 allocate-address --region us-east-1 --ipam-pool-id ipam-
pool-07ccc86aa41bef7ce --address 18.97.0.41
```

Example response:

```
{
    "PublicIp": "18.97.0.41",
    "AllocationId": "eipalloc-056cdd6019c0f4b46",
    "PublicIpv4Pool": "ipam-pool-07ccc86aa41bef7ce",
    "NetworkBorderGroup": "us-east-1",
    "Domain": "vpc"
}
```

For more information, see Allocate an Elastic IP address in the *Amazon EC2 User Guide*.

Step 4: Associate the Elastic IP address with an EC2 instance

Complete the steps in this section to associate the Elastic IP address with an EC2 instance.

AWS Management Console

Follow the steps in Associate an Elastic IP address in the *Amazon EC2 User Guide* to allocate an Elastic IP address from the IPAM pool, but note the following: When you use AWS Management Console option, the AWS Region you associate the Elastic IP address in must match the Locale option you chose when you created the pool in Step 2.

Command line

Associate the Elastic IP address with an instance with the associate-address command. The `--region` you associate the Elastic IP address in must match the `--locale` option you chose when you created the pool in Step 2.

```
aws ec2 associate-address --region us-east-1 --instance-id i-07459a6fca5b35823 --
public-ip 18.97.0.41
```

Example response:

```
{
    "AssociationId": "eipassoc-06aa85073d3936e0e"
}
```

For more information, see Associate an Elastic IP address with an instance or network interface in the *Amazon EC2 User Guide*.

Step 5: Track and monitor pool usage

Once you've allocated Elastic IP addresses from the IPAM pool, you can track and monitor IPAM pool allocations.

AWS Management Console

- View the IPAM pool details **Allocations** tab in the IPAM console. Any Elastic IP addresses allocated from the IPAM pool have a **Resource Type** of **EIP**.
- Use Public IP insights:
  - Under **Public IP types**, filter by **Amazon-owned EIPs**. This shows the total number of public IPv4 addresses allocated to Amazon-owned Elastic IP addresses. If you filter by this measure and scroll to **Public IP addresses** at the bottom of the page, you'll see the Elastic IP addresses you've allocated.
  - Under **EIP usage**, filter by **Associated Amazon-owned EIPs** or **Unassociated Amazon-owned EIPs**. This shows the total number of Elastic IP addresses that you have allocated in your AWS account and that you have or have not associated with an EC2 instance, network interface, or AWS resource. If you filter by this measure and scroll to **Public IP addresses** at the bottom of the page, you'll see details about the filtered resources.
  - Under **Amazon-owned IPv4 contiguous IPs usage**, monitor sequential public IPv4 address usage over time and related Amazon-owned IPv4 IPAM pools.
- Use Amazon CloudWatch to track and monitor metrics related to Amazon-provided contiguous public IPv4 blocks that have been provisioned to IPAM pools. For the available metrics specific to contiguous IPv4 blocks, see **Public IP Metrics** under IPAM metrics. In addition to viewing metrics, you can create alarms in Amazon CloudWatch to notify you when thresholds are reached. Creating alarms and setting up notifications with Amazon CloudWatch is outside the scope of this tutorial. For more information, see Using Amazon CloudWatch alarms in the *Amazon CloudWatch User Guide*.

Command line

- View the IPAM pool allocations with the get-ipam-pool-allocations command. Any Elastic IP addresses allocated from the IPAM pool have a **Resource Type** of **eip**.

```
aws ec2 get-ipam-pool-allocations --region us-east-1 --ipam-pool-id ipam-
pool-07ccc86aa41bef7ce
```

Example response:

```
{
     "IpamPoolAllocations": [
        {
            "Cidr": "18.97.0.40/32",
            "IpamPoolAllocationId": "ipam-pool-
alloc-0bd07df786e8148aba2763e2b6c1c44bd",
            "ResourceId": "eipalloc-0c9decaa541d89aa9",
            "ResourceType": "eip",
            "ResourceRegion": "us-east-1",
            "ResourceOwner": "320805250157"
        }
    ]
}
```

- Use Amazon CloudWatch to track and monitor metrics related to Amazon-provided contiguous public IPv4 blocks that have been provisioned to IPAM pools. For the available metrics specific to contiguous IPv4 blocks, see **Public IP Metrics** under IPAM metrics. In addition to viewing metrics, you can create alarms in Amazon CloudWatch to notify you when thresholds are reached. Creating alarms and setting up notifications with Amazon CloudWatch is outside the scope of this tutorial. For more information, see Using Amazon CloudWatch alarms in the *Amazon CloudWatch User Guide*.

The tutorial is now complete. You've created an IPAM pool with an Amazon-provided contiguous public IPv4 CIDR block, allocated Elastic IP addresses from the pool, and learned how to monitor IPAM pool allocations. Continue to the next section to delete the resources you've created in this tutorial.

Cleanup

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

Step 1: Disassociate the Elastic IP address

Complete the steps in Disassociate an Elastic IP address in the *Amazon EC2 User Guide* to disassociate the Elastic IP address.

Step 2: Release the Elastic IP address

Complete the steps in Release an Elastic IP address in the *Amazon EC2 User Guide* to release an Elastic IP address from the public IPv4 pool.

Step 3: Deprovision the CIDR from the IPAM pool

Complete the steps in Deprovision CIDRs from a pool to deprovision the Amazon-owned public CIDR from the IPAM pool. This step is required for pool deletion. You will be billed for the Amazon-provided contiguous IPv4 block until this step is complete.

Step 4: Delete the IPAM pool

Complete the steps in Delete a pool to delete the IPAM pool.

Step 5: Delete the IPAM

Complete the steps in Delete an IPAM to delete the IPAM.

The tutorial cleanup is complete.

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 about 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
- AWS managed policies for IPAM
- Example policy

Service-linked roles for IPAM

IPAM uses AWS Identity and Access Management (IAM) service-linked roles. A service-linked role is a unique type of IAM role. Service-linked roles are predefined by IPAM and include all the permissions that the service requires to call other AWS services on your behalf.

A service-linked role makes setting up IPAM easier because you don't have to manually add the necessary permissions. IPAM defines the permissions of its service-linked roles, and unless defined otherwise, only IPAM can assume its roles. The defined permissions include the trust policy and the permissions policy, and that permissions policy cannot be attached to any other IAM entity.

Service-linked role permissions

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.

Also attached to the service-linked role is an IAM trust 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 accounts in an AWS Organization 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, 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 a service-linked role description in the *IAM User Guide*.

Edit the service-linked role

You can't 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 and Delete an IPAM.
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 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": [
        {
            "Sid": "IPAMDiscoveryDescribeActions",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeAccountAttributes",
                "ec2:DescribeAddresses",
                "ec2:DescribeByoipCidrs",
                "ec2:DescribeIpv6Pools",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribePublicIpv4Pools",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSecurityGroupRules",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:DescribeVpnConnections",
                "ec2:GetIpamDiscoveredAccounts",
                "ec2:GetIpamDiscoveredPublicAddresses",
                "ec2:GetIpamDiscoveredResourceCidrs",
                "globalaccelerator:ListAccelerators",
                "globalaccelerator:ListByoipCidrs",
                "organizations:DescribeAccount",
                "organizations:DescribeOrganization",
                "organizations:ListAccounts",
                "organizations:ListDelegatedAdministrators",
                "organizations:ListChildren",
                "organizations:ListParents",
```

```
                "organizations:DescribeOrganizationalUnit"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CloudWatchMetricsPublishActions",
            "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.

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
AWSIPAMServiceRolePolicy	Actions added to the AWSIPAMServiceRolePolicy managed policy (`organizations:ListChildren` ,`organizations:ListParents` , and `organizations:Desc`	November 21, 2024

Change	Description	Date
	`ribeOrganizational Unit`) to enable IPAM to get the details of Organizational Units (OUs) in AWS Organizations so that customers can use IPAM at the OU level.	
AWSIPAMServiceRolePolicy	Action added to the AWSIPAMServiceRolePolicy managed policy (`ec2:GetIpamDiscoveredPublicAddresses`) to enable IPAM to get public IP addresses during resource discovery.	November 13, 2023
AWSIPAMServiceRolePolicy	Actions added to the AWSIPAMServiceRolePolicy managed policy (`ec2:DescribeAccountAttributes` , `ec2:DescribeNetworkInterfaces` , `ec2:DescribeSecurityGroups` , `ec2:DescribeSecurityGroupRules` , `ec2:DescribeVpnConnections` , `globalaccelerator:ListAccelerators` , and `globalaccelerator:ListByoipCidrs`) to enable IPAM to get public IP addresses during resource discovery.	November 1, 2023

Change	Description	Date
AWSIPAMServiceRolePolicy	Two actions added to the AWSIPAMServiceRolePolicy managed policy (`ec2:GetIpamDiscoveredAccounts` and `ec2:GetIpamDiscoveredResourceCidrs`) to enable IPAM to get the AWS accounts and resource CIDRs being monitored during resource discovery.	January 25, 2023
IPAM started tracking changes	IPAM started tracking changes for its AWS managed policies.	December 2, 2021

Example policy

The example policy in this section contains all the relevant AWS Identity and Access Management (IAM) actions for full IPAM usage. Depending on how you are using IPAM, you may not need to include all of the IAM actions. For a full experience using the IPAM console, you may need to include additional IAM actions for services such as AWS Organizations, AWS Resource Access Manager(RAM), and Amazon CloudWatch.

```
{
"Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AssociateIpamByoasn",
                "ec2:DeprovisionIpamByoasn",
                "ec2:DescribeIpamByoasn",
                "ec2:DisassociateIpamByoasn",
                "ec2:ProvisionIpamByoasn",
                "ec2:CreateIpam",
```

```
                "ec2:DescribeIpams",
                "ec2:ModifyIpam",
                "ec2:DeleteIpam",
                "ec2:CreateIpamScope",
                "ec2:DescribeIpamScopes",
                "ec2:ModifyIpamScope",
                "ec2:DeleteIpamScope",
                "ec2:CreateIpamPool",
                "ec2:DescribeIpamPools",
                "ec2:ModifyIpamPool",
                "ec2:DeleteIpamPool",
                "ec2:ProvisionIpamPoolCidr",
                "ec2:GetIpamPoolCidrs",
                "ec2:DeprovisionIpamPoolCidr",
                "ec2:AllocateIpamPoolCidr",
                "ec2:GetIpamPoolAllocations",
                "ec2:ReleaseIpamPoolAllocation",
                "ec2:CreateIpamResourceDiscovery",
                "ec2:DescribeIpamResourceDiscoveries",
                "ec2:ModifyIpamResourceDiscovery",
                "ec2:DeleteIpamResourceDiscovery",
                "ec2:AssociateIpamResourceDiscovery",
                "ec2:DescribeIpamResourceDiscoveryAssociations",
                "ec2:DisassociateIpamResourceDiscovery",
                "ec2:GetIpamResourceCidrs",
                "ec2:ModifyIpamResourceCidr",
                "ec2:GetIpamAddressHistory",
                "ec2:GetIpamDiscoveredResourceCidrs",
                "ec2:GetIpamDiscoveredAccounts",
                "ec2:GetIpamDiscoveredPublicAddresses"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "arn:aws:iam::*:role/aws-service-role/ipam.amazonaws.com/
AWSServiceRoleForIPAM",
            "Condition": {
                "StringLike": {
                    "iam:AWSServiceName": "ipam.amazonaws.com"
                }
            }
        }
```

```
    ]
}
```

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
Amazon-provided contiguous public IPv4 CIDR blocks	2	Yes. Contact the AWS Support Center as described in AWS service quotas in the *AWS General Reference*.
Amazon-provided contiguous public IPv4 CIDR block netmask length	/29	Acceptable size is between /29 and /30. To request an increase, contact the AWS Support Center as described in AWS service quotas in the *AWS General Reference*.
Amazon-provided IPv6 CIDR block netmask length	/52	Yes. Contact the AWS Support Center as described in AWS service quotas in the *AWS General Reference*.
Amazon-provided IPv6 CIDR blocks per Regional pool	1	Yes. Contact the AWS Support Center as described in AWS service quotas in

Name	Default	Adjustable
		the *AWS General Reference*.
Autonomous System Numbers (ASNs) that you can bring to IPAM	5	Yes. Contact the AWS Support Center as described in AWS service quotas in the *AWS General Reference*.
CIDRs per pool	50	Yes
IPAM administrators per organization	1	No
IPAMs per Region	1	No
Organizational unit exclusions per resource discovery	10	Yes. Contact the AWS Support Center as described in AWS service quotas in the *AWS General Reference*.
Pool depth (the number of pools within pools)	10	Yes
Pools per scope	50	Yes
Resource discovery associations per IPAM	5	Yes
Resource discoveries per Region	1	No
Resource utilization metrics	50	Yes. Contact the AWS Support Center as described in AWS service quotas in the *AWS General Reference*.

Name	Default	Adjustable
Scopes per IPAM	5	Yes. When you create an IPAM, a private and public default scope are created for you. If you want to create additional scopes, they will be private scopes. You cannot create additional public scopes.

Pricing for IPAM

Amazon VPC IP Address Manager (IPAM) is a service that helps you manage your IP address space across your AWS resources and on-premises networks. IPAM provides a centralized way to plan, monitor, and control the IP addresses used by your AWS and on-premises resources.

This section describes how to view pricing-related information and your current IPAM costs.

Contents

- View pricing information
- View your current costs and usage using AWS Cost Explorer

View pricing information

IPAM is offered in two tiers: Free and Advanced Tier. For more information about the features available in each tier and the costs associated with the tiers, see the **IPAM** tab on the Amazon VPC pricing page.

View your current costs and usage using AWS Cost Explorer

When you use the IPAM Advanced Tier, you pay an hourly price per active IP address managed by IPAM. If you want to view and analyze your IPAM costs and usage, you can use the AWS Cost Explorer.

1. Open the AWS Cost Management console at https://console.aws.amazon.com/cost-management/home.
2. Choose **Cost Explorer**.
3. Filter for IPAM usage by choosing **Usage type** and entering **`IPAddressManager`**.
4. Select one or more checkboxes. Each of them represents a different AWS Region.
5. Click **Apply**.

If, for example, you select *USE1-IPAddressManager-IP-Hours(Hrs)* and us-east-1 is your IPAM home Region, you'll see the number of active IP hours billed by IPAM in all Regions and the cost. If, say, the usage in hours is 18, this means that you could have 1 active IP address for 18 hours, 3 IP

addresses in 3 different Regions each active for 6 hours, or any combination of these that add up to 18 hours.

For more information about AWS Cost Explorer, see Analyzing your costs with AWS Cost Explorer in the *AWS Cost Management User Guide*.

Related information

While the AWS technical documentation site is a comprehensive resource, there are many other places to find information about AWS services. AWS blogs, whitepapers, case studies, and community forums can provide valuable insights, real-world examples, and alternative perspectives beyond the official technical details. Exploring these diverse sources can give you a more well-rounded understanding of AWS offerings.

The following related resources can help you as you work with Amazon VPC IP Address Manager:

- Amazon VPC IP Address Manager Best Practices: An AWS blog on best practices for planning and creating a scalable address scheme with Amazon VPC IP Address Manager.
- Network Address Management and Auditing at Scale with Amazon VPC IP Address Manager: An AWS blog that introduces Amazon VPC IP Address Manager and shows you how to use the service in the AWS console.
- Configure fine-grained access to your resources shared using AWS Resource Access Manager: An AWS blog that explains how to share an IPAM pool with the accounts in an AWS Organizations organization unit.
- Visualize enterprise IP address management and planning with CIDR map: An AWS blog that explains how to visualize your entire IPv4 and IPv6 landscape using the IPAM CIDR map in the IPAM console.

Document history for IPAM

The following table describes the releases for IPAM.

Feature	Description	Release Date
Enable cost distribution	When you enable cost distribution, you distribute the charges for active IP addresses to the accounts using the IP addresses rather than to the IPAM owner. This is useful for large organizations where the delegated IPAM admin manages the IP addresses centrally using IPAM and each account is responsible for their own usage, eliminating the need for manual billing calculations.	April 14, 2025
Exclude organizational units from IPAM	If your IPAM is integrated with AWS Organizations, you can now exclude organizational units from IPAM. IPAM will not manage the IP addresses in accounts in organizational unit exclusions.	November 21, 2024
AWS managed policy updates - Update to an existing policy	Existing AWSIPAMServiceRolePolicy updated.	November 21, 2024
Allocate sequential Elastic IP addresses from an IPAM pool	IPAM now enables you to provision Amazon-owned public IPv4 blocks to IPAM pools and allocate sequential Elastic IP addresses from those pools to AWS resources. Sequential Elastic IP addresses enable you to simplify your networking and security allowlisting needs.	August 28, 2024
Private IPv6 GUA and ULAs	You can now provision private IPv6 GUA and ULA ranges to an IPAM pool in a private scope. Private IPv6 addresses are only available in IPAM. For more information about	August 8, 2024

Feature	Description	Release Date
	private IPv6 addressing, see Private IPv6 addresses in the *Amazon VPC User Guide*.	
IPAM Free and Advanced Tiers	You can now choose between Free Tier and Advanced Tier for your IPAM.	November 17, 2023
Public IP insights	Previously, you could only view public IP insights in a single Region. You can now view public IP insights across Regions. In addition, you can now view public IP address insights in Amazon CloudWatch.	November 17, 2023
Plan VPC IP address space for subnet IP allocations	You can now use IPAM to plan for subnet IP space within a VPC and monitor IP address-r elated metrics at the subnet and VPC level.	November 17, 2023
Bring your own ASN (BYOASN)	You can now bring your own autonomous system number (ASN) to AWS.	November 17, 2023
AWS managed policy updates - Update to an existing policy	Existing AWSIPAMServiceRolePolicy updated.	November 17, 2023
AWS managed policy updates - Update to an existing policy	Existing AWSIPAMServiceRolePolicy updated.	November 1, 2023
Resource utilization metrics	IPAM now publishes IP utilization metrics for resources that the IPAM monitors to Amazon CloudWatch.	August 2, 2023

Feature	Description	Release Date
Public IP insights	Public IP insights shows you all public IPv4 addresses used by services in this Region in your account. You can use these insights to identify public IPv4 address usage and view recommendations to release unused Elastic IP addresses.	July 28, 2023
AWS managed policy updates - Update to an existing policy	Existing AWSIPAMServiceRolePolicy updated.	January 25, 2023
Integrate IPAM with accounts outside of your organization	You can now manage IP addresses outside of your organization from a single IPAM account and share IPAM pools with the accounts of other AWS Organizations.	January 25, 2023
Amazon-provided IPv6 contiguous CIDR block for IPAM pools	When you create an IPAM pool in the public scope, you can now provision an Amazon-provided IPv6 contiguous CIDR block to the pool. For more information, see Create IPv6 address pools in your IPAM.	January 25, 2023
Initial release	This release introduces Amazon VPC IP Address Manager.	December 2, 2021