aws

Research and Engineering Studio

Research and Engineering Studio: User Guide

Table of Contents

Overview

Research and Engineering Studio (RES) is an AWS supported, open source product that enables IT administrators to provide a web portal for scientists and engineers to run technical computing workloads on AWS. RES provides a single pane of glass for users to launch secure virtual desktops to conduct scientific research, product design, engineering simulations, or data analysis workloads. Users can connect to the RES portal using their existing corporate credentials and work on individual or collaborative projects.

Administrators can create virtual collaboration spaces called projects for a specific set of users to access shared resources and collaborate. Admins can build their own application software stacks (AMIs) and allow RES users to launch Windows or Linux virtual desktops, and enable access to project data through shared file-systems. Admins can assign software stacks and file-systems and restrict access to only those project users. Admins can use built-in telemetry to monitor the environment usage and troubleshoot user issues. They can also set budgets for individual projects to prevent overconsumption of resources. As the product is open source, customers can also customize the user-experience of the RES portal to suit their own needs.

RES is available at no additional charge, and you pay only for the AWS resources needed to run your applications.

This guide provides an overview of Research and Engineering Studio on AWS, its reference architecture and components, considerations for planning the deployment, and configuration steps for deploying RES to the Amazon Web Services (AWS) Cloud.

Features and benefits

Research and Engineering Studio on AWS provides the following features:

Web-based user interface

RES provides a web-based portal that administrators, researchers, and engineers can use to access and manage their research and engineering workspaces. Scientists and engineers do not need to have an AWS account or cloud expertise to use RES.

Project-based configuration

Use projects to define access permissions, allocate resources, and manage budgets for a set of tasks or activities. Assign specific software stacks (operating systems and approved applications)

and storage resources to a project for consistency and compliance. Monitor and manage spend on a per-project basis.

Collaboration tools

Scientists and engineers can invite other members of their project to collaborate with them, setting the permissions levels they want those colleagues to have. Those individuals can sign in to RES to connect to those desktops.

Integration with existing identity management infrastructure

Integrate with your existing identity management and directory services infrastructure to enable connection to the RES portal with a user's existing corporate identity and assign permissions to projects using existing user and group memberships.

Persistent storage and access to shared data

To provide users access to shared data across virtual desktop sessions, connect to your existing file systems or create new file systems within RES. Supported storage services include Amazon Elastic File System for Linux desktops and Amazon FSx for NetApp ONTAP for Windows and Linux desktops.

Monitoring and reporting

Use the analytics dashboard to monitor resource usage for instance types, software stacks, and operating system types. The dashboard also provides a breakdown of resource usage by projects for reporting.

Budget and cost management

Link AWS Budgets to your RES projects to monitor costs for each project. If you exceed your budget, you can limit the launch of VDI sessions.

Concepts and definitions

This section describes key concepts and defines terminology specific to this product:

File browser

A file browser is a part of the RES user interface where currently logged-in users can view their file system.

File system

The file system acts as a container for project data (often referred to as datasets). It provides a storage solution within a project's boundaries and improves collaboration and data access control.

Global administrator

An administrative delegate with access to RES resources that are shared across a RES environment. Scope and permissions span multiple projects. They can create or modify projects and assign project owners. They can delegate or assign permissions to project owners and project members. Sometimes the same person acts as the RES administrator depending on the size of the organization.

Project

A project is a logical partition within the application that serves as a distinct boundary for data and compute resources, ensuring governance over data flow and preventing data and VDI host sharing across projects.

Project-based permissions

Project-based permissions describes a logical partition of both data and VDI hosts in a system where multiple projects can exist. A user's access to data and VDI hosts within a project is determined by their associated role(s). A user must be assigned access (or project membership) for each project to which they require access. Otherwise, a user is unable to access project data and VDIs when they have not been granted membership.

Project member

An end user of RES resources (VDI, storage, etc). Scope and permissions are restricted to projects they are assigned to. They cannot delegate or assign any permissions.

Project owner

An administrative delegate with access to and ownership over a specific project. Scope and permissions are restricted to the project(s) they own. They can assign permissions to project members in the projects they own.

Software stack

Software stacks are [Amazon Machine Images (AMI)](#) with RES-specific metadata based on any operating system a user has selected to provision for their VDI host.

VDI hosts

Virtual desktop instance (VDI) hosts allow project members to access project-specific data and compute environments, ensuring secure and isolated workspaces.

For a general reference of AWS terms, see the [AWS glossary](#) in the AWS General Reference.

Architecture overview

This section provides an architecture diagram for the components deployed with this product.

Architecture diagram

Deploying this product with the default parameters deploys the following components in your AWS account.
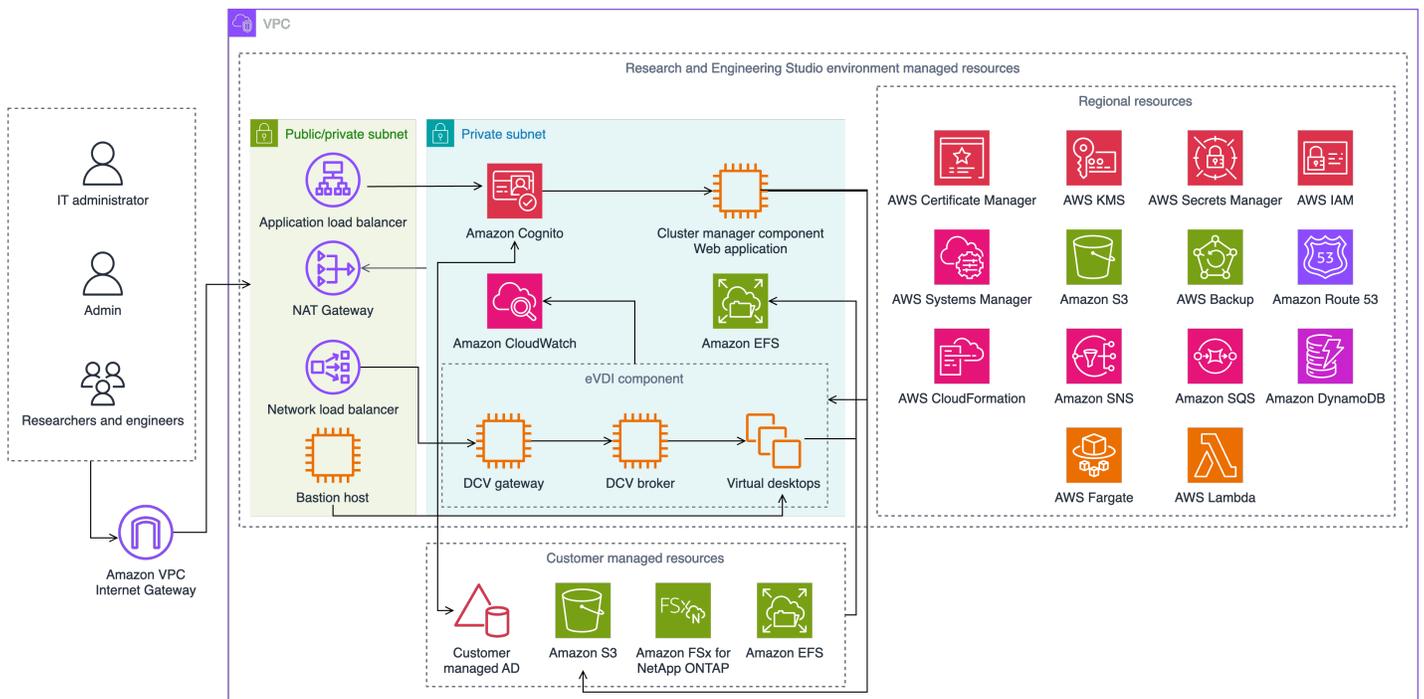
Figure 1: Research and Engineering Studio on AWS architecture

> ⓘ **Note**
>
> AWS CloudFormation resources are created from AWS Cloud Development Kit (AWS CDK) constructs.

The high-level process flow for the product components deployed with the AWS CloudFormation template is as follows:

1. RES installs components for the web portal as well as:

a. Engineering Virtual Desktop (eVDI) component for interactive workloads

b. Metrics component

   Amazon CloudWatch receives metrics from the eVDI components.

c. Bastion Host component

   Administrators may connect to the bastion host component using SSH to manage the underlying infrastructure.

2. RES installs components in private subnets behind a NAT gateway. Administrators access the private subnets via the Application Load Balancer (ALB) or the Bastion Host component.

3. Amazon DynamoDB stores the environment configuration.

4. AWS Certificate Manager (ACM) generates and stores a public certificate for the Application Load Balancer (ALB).

> ⓘ **Note**
>
> We recommend using AWS Certificate Manager to generate a trusted certificate for your domain.

5. Amazon Elastic File System (EFS) hosts the default /home file system mounted on all applicable infrastructure hosts and eVDI Linux sessions.

6. RES uses Amazon Cognito to create an initial bootstrap user called clusteradmin within and sends temporary credentials to the email address provided during installation. The clusteradmin must change the password with first-time login.

7. Amazon Cognito integrates with your organization's Active Directory and user identities for permissions management.

8. Security zones allow administrators to restrict access to specific components within the product based on permissions.

AWS services in this product

AWS service	Description
Amazon Elastic Compute Cloud	**Core.** Provides the underlying compute services to create virtual desktops with their chosen operating system and software stack.
Elastic Load Balancing	**Core.** Bastion, cluster-manager, and VDI hosts are created in Auto Scaling groups behind the load balancer. ELB balances traffic from the web portal across RES hosts.
Amazon Virtual Private Cloud	**Core.** All core product components are created within your VPC.
Amazon Cognito	**Core.** Manages user identities and authentication. Active Directory users are mapped to Amazon Cognito users and groups to authenticate access levels.
Amazon Elastic File System	**Core.** Provides the /home file system for the file browser and VDI hosts, as well as shared external file systems.
Amazon DynamoDB	**Core.** Stores configuration data such as users, groups, projects, file systems, and component settings.
AWS Systems Manager	**Core.** Stores documents for performing commands for VDI session management.
AWS Lambda	**Core.** Supports product functionalities such as updating settings within the DynamoDB table, starting Active Directory sync workflows, and updating the prefix list.

AWS service	Description
Amazon CloudWatch	**Supporting.** Provides metrics and activity logs for all Amazon EC2 hosts and Lambda functions.
Amazon Simple Storage Service	**Supporting.** Stores application binaries for host bootstrapping and configuration.
AWS Key Management Service	**Supporting.** Used for encryption at rest with Amazon SQS queues, DynamoDB tables, and Amazon SNS topics.
AWS Secrets Manager	**Supporting.** Stores service account credentials in Active Directory and self-signed certificates for VDIs.
AWS CloudFormation	**Supporting.** Provides a deployment mechanism for the product.
AWS Identity and Access Management	**Supporting.** Restricts the access level for hosts.
Amazon Route 53	**Supporting.** Creates private hosted zone for resolving the internal load balancer and the bastion host domain name.
Amazon Simple Queue Service	**Supporting.** Creates task queues to support asynchronous executions.
Amazon Simple Notification Service	**Supporting.** Supports the publication-subscriber model between VDI components such as the controller and hosts.
AWS Fargate	**Supporting.** Installs, updates, and deletes environments using Fargate tasks.
Amazon FSx File Gateway	**Optional.** Provides external shared file system.

AWS service	Description
Amazon FSx for NetApp ONTAP	**Optional.** Provides external shared file system.
AWS Certificate Manager	**Optional.** Generates a trusted certificate for your custom domain.
AWS Backup	**Optional.** Offers backup capabilities for Amazon EC2 hosts, file systems, and DynamoDB.

Create a demo environment

Follow the steps in this section to try out Research and Engineering Studio on AWS. This demo deploys a non-production environment with a minimal set of parameters using the Research and Engineering Studio on AWS demo environment stack template. It uses a Keycloak server for SSO.

Note that after you deploy the stack, you must follow the steps in Post deployment steps below to set up users in the environment before you login.

Create a one-click demo stack

This AWS CloudFormation stack creates all the components required by Research and Engineering Studio.

Time to deploy: ~90 minutes

Prerequisites

Topics

- Create an AWS account with an administrative user
- Create an Amazon EC2 SSH key pair
- Increase service quotas

Create an AWS account with an administrative user

You must have an AWS account with an administrative user:

1. Open https://portal.aws.amazon.com/billing/signup.
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

   When you sign up for an AWS account, an *AWS account root user* is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to a user, and use only the root user to perform tasks that require root user access.

Create an Amazon EC2 SSH key pair

If you do not have Amazon EC2 SSH key pair, you will need to create one. For more information, see Create a key pair using Amazon EC2 in the *Amazon EC2 User Guide.*

Increase service quotas

We recommend increasing the service quotas for:

- Amazon VPC
  - Increase the Elastic IP address quota per NAT gateway from five to eight
  - Increase the NAT gateways per Availability Zone from five to ten
- Amazon EC2
  - Increase the EC2-VPC Elastic IPs from five to ten

Your AWS account has default quotas, formerly referred to as limits, for each AWS service. Unless otherwise noted, each quota is Region-specific. You can request increases for some quotas, and other quotas cannot be increased. For more information, see the section called "Quotas for AWS services in this product".

Create resources and input parameters

1. Sign in to the AWS Management Console and open the AWS CloudFormation console at https://console.aws.amazon.com/cloudformation.

   > ⓘ **Note**
   >
   > Make sure you are in your administrator account.

2. Launch the template in the console.

3. Under **Parameters**, review the parameters for this product template and modify them as necessary.

Parameter	Default	Description
EnvironmentName	*<res-demo>*	A unique name given to your RES environment

Parameter	Default	Description
		starting with res- and no longer than 11 characters.
AdministratorEmail		The email address for the user completing setup of the product. This user additionally functions as a break-glass user if there is an Active Directory single sign on integration failure.
KeyPair		The key pair used to connect to infrastructure hosts.
ClientIPCidr	<0.0.0.0/0>	IP address filter which limits connection to the system. You can update the ClientIpCidr after deployment.
InboundPrefixList		*(Optional)* Provide a managed prefix list for IPs allowed to directly access the web UI and SSH into the bastion host.

Post deployment steps

1. Reset user passwords in `AWS Directory Service`– The demo stack creates four users with usernames which you can use: `admin1`, `user1`, `admin2`, and `user2`.

   a. Go to the Directory Service console.

   b. Select the Directory Id for your environment. You can get the Directory Id from the output of `<StackName>*DirectoryService*` stack.

   c. From the top right **Action** dropdown menu, select **Reset user password**.

      d.   For all the users you want to use, put the username and type in the password you want to have and select **Reset Password**.

2.   Once you have reset the user passwords, you will need to wait for Research and Engineering Studio to sync the users in the environment. Research and Engineering Studio syncs the users every hour at xx.00. You can either wait for that to happen or follow the steps listed in [User added in Active Directory, but missing from RES](#) to sync the users immediately.

Your deployment is now ready. Use the EnvironmentUrl you received in your email to access the UI, or you can also get the same URL from the output of the deployed stack. You may now login to the Research and Engineering Studio environment with the user and password that you reset the password for in Active Directory.

Plan your deployment

Cost

Research and Engineering Studio on AWS is available at no additional charge, and you pay only for the AWS resources needed to run your applications. For more information, see AWS services in this product.

> **ⓘ Note**
>
> You are responsible for the cost of the AWS services used while running this product. We recommend creating a budget through AWS Cost Explorer to help manage costs. Prices are subject to change. For full details, see the pricing webpage for each AWS service used in this product.

Security

When you build systems on AWS infrastructure, security responsibilities are shared between you and AWS. This shared responsibility model reduces your operational burden because AWS operates, manages, and controls the components including the host operating system, the virtualization layer, and the physical security of the facilities in which the services operate. For more information about AWS security, visit AWS Cloud Security.

IAM roles

AWS Identity and Access Management (IAM) roles allow customers to assign granular access policies and permissions to services and users on the AWS Cloud. This product creates IAM roles that grant the product's AWS Lambda functions and Amazon EC2 instances access to create Regional resources.

RES supports identity-based policies within IAM. When deployed, RES creates policies to define the administrator permission and access. The administrator implementing the product creates and manages end users and project leaders within the existing customer Active Directory integrated with RES. For more information, see Creating IAM policies in the *AWS Identity and Access Management User Guide*.

Your organization's administrator can manage user access with an active directory. When end users access the RES user interface, RES authenticates with Amazon Cognito.

Security groups

The security groups created in this product are designed to control and isolate network traffic between the Lambda functions, EC2 instances, file systems CSR instances, and remote VPN endpoints. We recommend that you review the security groups and further restrict access as needed once the product is deployed.

Data encryption

By default, Research and Engineering Studio on AWS (RES) encrypts customer data at rest and in transit using an RES owned key. When you deploy RES, you may specify an AWS KMS key. RES uses your credentials to grant key access. If you supply a customer owned and managed AWS KMS key, customer data at rest will be encrypted using that key.

RES encrypts customer data in transit using SSL/TLS. We require TLS 1.2, but recommend TLS 1.3.

Supported AWS Regions

This product uses services which are not currently available in all AWS Regions. You must launch this product in an AWS Region where all services are available. For the most current availability of AWS services by Region, see the AWS Regional Services List.

Research and Engineering Studio on AWS is supported in the following AWS Regions:

Region name	
US East (Ohio)	Canada (Central)
US East (N. Virginia)	Europe (Frankfurt)
US West (N. California)	Europe (Ireland)
US West (Oregon)	Europe (London)
Asia Pacific (Mumbai)	Europe (Milan)
Asia Pacific (Seoul)	Europe (Paris)

Region name	
Asia Pacific (Singapore)	Israel (Tel Aviv)
Asia Pacific (Sydney)	AWS GovCloud (US-West)
Asia Pacific (Tokyo)	

Quotas

Service quotas, also referred to as limits, are the maximum number of service resources or operations for your AWS account.

Quotas for AWS services in this product

Make sure you have sufficient quota for each of the services implemented in this product. For more information, see AWS service quotas.

For this product, we recommend raising quotas for the following services:

- Amazon Virtual Private Cloud
- Amazon EC2

To request a quota increase, see Requesting a Quota Increase in the *Service Quotas User Guide*. If the quota is not yet available in Service Quotas, use the limit increase form.

AWS CloudFormation quotas

Your AWS account has AWS CloudFormation quotas that you should be aware of when launching the stack in this product. By understanding these quotas, you can avoid limitation errors that would prevent you from deploying this product successfully. For more information, see AWS CloudFormation quotas in the in the *AWS CloudFormation User's Guide*.

Planning for resilience

The product deploys a default infrastructure with the minimum number and size of Amazon EC2 instances to operate the system. To improve resilience in large-scale production environments,

we recommend increasing the default minimum capacity settings within the infrastructure's Auto Scaling groups (ASG). Increasing the value from one instance to two instances provides the benefit of multiple Availability Zones (AZ) and reduces the time to restore system functionality in the event of unexpected data loss.

ASG settings can be customized within the Amazon EC2 console at https://console.aws.amazon.com/ec2/. The product creates four ASGs by default with each name ending with `-asg`. You can change the minimum and desired values to an amount appropriate for your production environment. Choose the group you want to modify, and then choose **Actions** and **Edit**. For more information on ASGs, see Scale the size of your Auto Scaling group in the *Amazon EC2 Auto Scaling User Guide*.

Deploy the product

> **ⓘ Note**
>
> This product uses AWS CloudFormation templates and stacks to automate its deployment. The CloudFormation templates describe the AWS resources included in this product and their properties. The CloudFormation stack provisions the resources that are described in the templates.

Before you launch the product, review the cost, architecture, network security, and other considerations discussed earlier in this guide.

Topics

- Prerequisites
- Create external resources
- Step 1: Launch the product
- Step 2: Sign in for the first time

Prerequisites

Topics

- Create an AWS account with an administrative user
- Create an Amazon EC2 SSH key pair
- Increase service quotas
- Create a public domain (optional)
- Create domain (GovCloud only)
- Provide external resources
- Configure LDAPS in your environment (optional)
- Configure a private VPC (optional)

Create an AWS account with an administrative user

You must have an AWS account with an administrative user:

1. Open https://portal.aws.amazon.com/billing/signup.

2. Follow the online instructions.

    Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

    When you sign up for an AWS account, an *AWS account root user* is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to a user, and use only the root user to perform [tasks that require root user access](#).

Create an Amazon EC2 SSH key pair

If you do not have Amazon EC2 SSH key pair, you will need to create one. For more information, see [Create a key pair using Amazon EC2](#) in the *Amazon EC2 User Guide*.

Increase service quotas

We recommend [increasing the service quotas](#) for:

- [Amazon VPC](#)
  - Increase the Elastic IP address quota per NAT gateway from five to eight
  - Increase the NAT gateways per Availability Zone from five to ten
- [Amazon EC2](#)
  - Increase the EC2-VPC Elastic IPs from five to ten

Your AWS account has default quotas, formerly referred to as limits, for each AWS service. Unless otherwise noted, each quota is Region-specific. You can request increases for some quotas, and other quotas cannot be increased. For more information, see [the section called "Quotas for AWS services in this product"](#).

Create a public domain (optional)

We recommend using a custom domain for the product in order to have a user-friendly URL. You will need to register a domain using Amazon Route 53 or another provider and import a certificate for the domain using AWS Certificate Manager. If you already have a public domain and certificate, you may skip this step.

1. Follow the directions to register a domain with Route53. You should receive a confirmation email.

2. Retrieve the hosted zone for your domain. This is created automatically by Route53.

   a. Open the Route53 console.

   b. Choose **Hosted zones** from the left navigation.

   c. Open the hosted zone created for your domain name and copy the **Hosted zone ID**.

3. Open AWS Certificate Manager and follow these steps to request a domain certificate. Ensure you are in the Region where you plan to deploy the solution.

4. Choose **List certificates** from the navigation, and find your certificate request. The request should be pending.

5. Choose your **Certificate ID** to open the request.

6. From the **Domains** section, choose **Create records in Route53**. It will take approximately ten minutes for the request to process.

7. Once the certificate is issued, copy the **ARN** from the **Certificate status** section.

Create domain (GovCloud only)

If you are deploying in the AWS GovCloud (US-West) Region, you will need to complete these prerequisite steps.

1. Deploy the Certificate AWS CloudFormation stack in the commercial-partition AWS Account where the public hosted domain was created.

2. From the **Certificate CloudFormation Outputs**, find and note the `CertificateARN` and `PrivateKeySecretARN`.

3. In the GovCloud partition account, create a secret with the value of the `CertificateARN` output. Note the new secret ARN and add two tags to the secret so `vdc-gateway` can access the secret value:

a.   res:ModuleName = virtual-desktop-controller

b.   res:EnvironmentName = [environment name] (This could be res-demo.)

4.   In the GovCloud partition account, create a secret with the value of the
     `PrivateKeySecretArn` output. Note the new secret ARN and add two tags to the secret so
     `vdc-gateway` can access the secret value:

a.   res:ModuleName = virtual-desktop-controller

b.   res:EnvironmentName = [environment name] (This could be res-demo.)

Provide external resources

When you deploy Research and Engineering Studio on AWS, there are external resources used by
the product you will need. RES expects those resources to exist when deployed.

- **Networking (VPC, Public, and Private Subnets)**

  This is where you will run the EC2 instances used to host the environment, the Active Directory
  (AD), and shared storage.

- **Storage (Amazon EFS)**

  The storage volumes contain files and data needed for the virtual desktop infrastructure (VDI).

- **Directory service (AWS Directory Service for Microsoft Active Directory)**

  The directory service authenticates users to the environment pages.

- **A secret that contains the service account password**

  Research and Engineering Studio accesses secrets that you provide, including the service account
  password, using AWS Secrets Manager.

> ⓘ **Tip**
>
> If you are deploying a demo environment and do not have these external resources
> available, you can use AWS High Performance Compute recipes to generate the external
> resources. See the following section, Create external resources, to deploy resources in your
> account.

> For demo deployments in the AWS GovCloud (US-West) Region, you will need to complete the prerequisite steps in Create domain (GovCloud only).

Configure LDAPS in your environment (optional)

If you plan to use LDAPS communication in your environment, you must complete these steps to create and attach certificates to the AWS Managed Microsoft AD (AD) domain controller to provide communication between AD and RES.

1. Follow the steps provided in How to enable server-side LDAPS for your AWS Managed Microsoft AD. You can skip this step if you have already enabled LDAPS.

2. After confirming that LDAPS is configured on the AD, export the AD certificate:

   a. Go to your Active Directory server.

   b. Open PowerShell as an administrator.

   c. Run `certmgr.msc` to open the certificate list.

   d. Open the certificate list by first opening the Trusted Root Certification Authorities and then Certificates.

   e. Select and hold (or right-click) the certificate with the same name as your AD server and choose **All tasks** and then **Export**.

   f. Choose **Base-64 encoded X.509 (.CER)** and choose **Next**.

   g. Select a directory and then choose **Next**.

3. Create a secret in AWS Secrets Manager:

   When creating your Secret in the Secrets Manager, choose **Other type of secrets** under **secret type** and paste your PEM encoded certificate in the **Plaintext** field.

4. Note the ARN created and input it as the `DomainTLSCertificateSecretARN` parameter in the section called "Step 1: Launch the product".

Configure a private VPC (optional)

Deploying Research and Engineering Studio in an isolated VPC offers enhanced security to meet your organization's compliance and governance requirements. However, the standard RES deployment relies on internet access for installing dependencies. To install RES in a private VPC, you will need to satisfy the following prerequisites:

Topics

-

-

-

-

Prepare Amazon Machine Images (AMIs)

1. Download [dependencies](#). To deploy in an isolated VPC, the RES infrastructure requires the availability of dependencies without having public internet access.

2. Create an IAM role with Amazon S3 read-only access and trusted identity as Amazon EC2.

   a. Open the IAM console at https://console.aws.amazon.com/iam/.

   b. From **Roles**, choose **Create role**.

   c. On the **Select trusted entity** page:

      - Under **Trusted entity type**, choose AWS service.

      - For **Use case** under **Service or use case**, select **EC2** and choose **Next**.

   d. On **Add permissions**, select the following permission policies and then choose **Next**:

      - AmazonS3ReadOnlyAccess

      - AmazonSSMManagedInstanceCore

      - EC2InstanceProfileForImageBuilder

   e. Add a **Role name** and **Description**, and then choose **Create role**.

3. Create the EC2 image builder component:

   a. Open the EC2 Image Builder console at https://console.aws.amazon.com/imagebuilder.

   b. Under **Saved resources**, choose **Components** and choose **Create component**.

   c. On the **Create component** page, enter the following details:

      - For **Component type**, choose **Build**.

      - For **Component details** choose:

Parameter	User entry
Image operating system (OS)	Linux
Compatible OS Versions	Amazon Linux 2
Component name	Choose a name such as: *<research-and-engineering-studio-infrastructure>*
Component version	We recommend starting with 1.0.0.
Description	Optional user entry.

d. On the **Create component** page, choose **Define document content**.

i.   Before entering the definition document content, you will need a file URI for the tar.gz file. Upload the tar.gz file provided by RES to an Amazon S3 bucket and copy the file's URI from the bucket properties.

ii. Enter the following:

> ⓘ **Note**
>
> AddEnvironmentVariables is optional, and you may remove it if you do not require custom environment variables in your infrastructure hosts.
> If you are setting up http_proxy and https_proxy environment variables, the no_proxy parameters are required to prevent the instance from using proxy to query localhost, instance metadata IP addresses, and the services that support VPC endpoints.

```
#  Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
#
#  Licensed under the Apache License, Version 2.0 (the "License"). You may
 not use this file except in compliance
#  with the License. A copy of the License is located at
#
#      http://www.apache.org/licenses/LICENSE-2.0
```

```
#
#  or in the 'license' file accompanying this file. This file is
 distributed on an 'AS IS' BASIS, WITHOUT WARRANTIES
#  OR CONDITIONS OF ANY KIND, express or implied. See the License for the
 specific language governing permissions
#  and limitations under the License.
name: research-and-engineering-studio-infrastructure
description: An RES EC2 Image Builder component to install required RES
 software dependencies for infrastructure hosts.
schemaVersion: 1.0

parameters:
  - AWSAccountID:
      type: string
      description: RES Environment AWS Account ID
  - AWSRegion:
      type: string
      description: RES Environment AWS Region
phases:
  - name: build
    steps:
        - name: DownloadRESInstallScripts
          action: S3Download
          onFailure: Abort
          maxAttempts: 3
          inputs:
              - source: '<s3 tar.gz file uri>'
                destination: '/root/bootstrap/res_dependencies/
res_dependencies.tar.gz'
                expectedBucketOwner: '{{ AWSAccountID }}'
        - name: RunInstallScript
          action: ExecuteBash
          onFailure: Abort
          maxAttempts: 3
          inputs:
              commands:
                  - 'cd /root/bootstrap/res_dependencies'
                  - 'tar -xf res_dependencies.tar.gz'
                  - 'cd all_dependencies'
                  - '/bin/bash install.sh'
        - name: AddEnvironmentVariables
          action: ExecuteBash
          onFailure: Abort
          maxAttempts: 3
```

```
            inputs:
                commands:
                    - |
                        echo -e "
                        http_proxy=http://<ip>:<port>
                        https_proxy=http://<ip>:<port>

 no_proxy=127.0.0.1,169.254.169.254,169.254.170.2,localhost,
{{ AWSRegion }}.res,{{ AWSRegion }}.vpce.amazonaws.com,
{{ AWSRegion }}.elb.amazonaws.com,s3.
{{ AWSRegion }}.amazonaws.com,s3.dualstack.
{{ AWSRegion }}.amazonaws.com,ec2.{{ AWSRegion }}.amazonaws.com,ec2.
{{ AWSRegion }}.api.aws,ec2messages.{{ AWSRegion }}.amazonaws.com,ssm.
{{ AWSRegion }}.amazonaws.com,ssmmessages.
{{ AWSRegion }}.amazonaws.com,kms.
{{ AWSRegion }}.amazonaws.com,secretsmanager.
{{ AWSRegion }}.amazonaws.com,sqs.
{{ AWSRegion }}.amazonaws.com,elasticloadbalancing.
{{ AWSRegion }}.amazonaws.com,sns.{{ AWSRegion }}.amazonaws.com,logs.
{{ AWSRegion }}.amazonaws.com,logs.
{{ AWSRegion }}.api.aws,elasticfilesystem.
{{ AWSRegion }}.amazonaws.com,fsx.{{ AWSRegion }}.amazonaws.com,dynamodb.
{{ AWSRegion }}.amazonaws.com,api.ecr.
{{ AWSRegion }}.amazonaws.com,.dkr.ecr.
{{ AWSRegion }}.amazonaws.com,kinesis.{{ AWSRegion }}.amazonaws.com,.data-
kinesis.{{ AWSRegion }}.amazonaws.com,.control-
kinesis.{{ AWSRegion }}.amazonaws.com,events.
{{ AWSRegion }}.amazonaws.com,cloudformation.
{{ AWSRegion }}.amazonaws.com,sts.
{{ AWSRegion }}.amazonaws.com,application-autoscaling.
{{ AWSRegion }}.amazonaws.com,monitoring.{{ AWSRegion }}.amazonaws.com
                        " > /etc/environment
```

    e.   Choose **Create component**.

4.   Create an Image Builder image recipe.

    a.   On the **Create recipe** page, enter the following:

Section	Parameter	User entry
Recipe details	**Name**	Enter an appropriate name such as res-recipe-linux-x 86.
	Version	Enter a version, typically starting with 1.0.0.
	Description	Add an optional descripti on.
Base image	**Select image**	Select managed images.
	OS	Amazon Linux
	Image origin	Quick start (Amazon-m anaged)
	Image name	Amazon Linux 2 x86
	Auto-versioning options	Use latest available OS version.
Instance configuration	**–**	Keep everything in the default settings, and make sure Remove SSM agent after pipeline execution is not selected.
Working directory	**Working directory path**	/root/bootstrap/re s_dependencies

Section	Parameter	User entry
Components	**Build components**	Search for and select the following: • Amazon-managed: aws-cli-version-2-linux • Amazon-managed: amazon-cloudwatch-agent-linux • Owned by you: Amazon EC2 component created previously. Put your AWS account ID and current AWS Region in the fields.
	Test components	Search for and select: • Amazon-managed: simple-boot-test-linux

    b.   Choose **Create recipe**.

5.   Create Image Builder infrastructure configuration.

    a.   Under **Saved resources**, choose **Infrastructure configurations**.

    b.   Choose **Create infrastructure configuration**.

    c.   On the **Create infrastructure configuration** page, enter the following:

Section	Parameter	User entry
General	**Name**	Enter an appropriate name such as res-infra-linux-x86.
	Description	Add an optional description.

Section	Parameter	User entry
	IAM role	Select the IAM role created previously.
AWS infrastructure	Instance type	Choose t3.medium.
	VPC, subnet, and security groups	Select an option that permits internet access and access to the Amazon S3 bucket. If you need to create a security group, you can create one from the Amazon EC2 console with the following inputs: • VPC: Select the same VPC being used for the infrastructure configuration. This VPC must have internet access. • Inbound rule: • Type: SSH • Source: Custom • CIDR block: 0.0.0.0/0

    d.   Choose **Create infrastructure configuration**.

6.   Create a new EC2 Image Builder pipeline:

    a.   Go to **Image pipelines**, and choose **Create image pipeline**.

    b.   On the **Specify pipeline details** page, enter the following and choose **Next**:

        • Pipeline name and optional description

        • For **Build schedule**, set a schedule or choose **Manual** if you want to start the AMI baking process manually.

    c.   On the **Choose recipe** page, choose **Use existing recipe** and enter the **Recipe name** created previously. Choose **Next**.

    d.   On the **Define image process** page, select the default workflows and choose **Next**.

    e.   On the **Define infrastructure configuration** page, choose **Use existing infrastructure configuration** and enter the name of the previously created infrastructure configuration. Choose **Next**.

    f.   On the **Define distribution settings** page, consider the following for your selections:

- The output image must reside in the same region as the deployed RES environment, so that RES can properly launch infrastructure host instances from it. Using service defaults, the output image will be created in the region where the EC2 Image Builder service is being used.

- If you want to deploy RES in multiple regions, you can choose **Create a new distribution settings** and add more regions there.

    g.   Review your selections and choose **Create pipeline**.

7.   Run the EC2 Image Builder pipeline:

    a.   From **Image pipelines**, find and select the pipeline you created.

    b.   Choose **Actions**, and choose **Run pipeline**.

    The pipeline may take approximately 45 minutes to an hour to create an AMI image.

8.   Note the AMI ID for the generated AMI and use it as the input for the InfrastructureHostAMI parameter in the section called "Step 1: Launch the product".

Set up VPC endpoints

To deploy RES and launch virtual desktops, AWS services require access to your private subnet. You must set up VPC endpoints to provide the required access, and you will need to repeat these steps for each endpoint.

1.   If endpoints have not previously been configured, follow the instructions provided in Access an AWS service using an interface VPC endpoint.

2.   Select one private subnet in each of the two availability zones.

AWS service	Service name
Application Auto Scaling	com.amazonaws.*region*.application-autoscaling

AWS service	Service name
AWS CloudFormation	com.amazonaws.*region*.cloudformation
Amazon CloudWatch	com.amazonaws.*region*.monitoring
Amazon CloudWatch Logs	com.amazonaws.*region*.logs
Amazon DynamoDB	com.amazonaws.*region*.dynamodb (Requires gateway endpoint)
Amazon EC2	com.amazonaws.*region*.ec2
Amazon ECR	com.amazonaws.*region*.ecr.api
	com.amazonaws.*region*.ecr.dkr
Amazon Elastic File System	com.amazonaws.*region*.elasticfilesystem
Elastic Load Balancing	com.amazonaws.*region*.elasticloadbalancing
Amazon EventBridge	com.amazonaws.*region*.events
Amazon FSx	com.amazonaws.*region*.fsx
AWS Key Management Service	com.amazonaws.*region*.kms
Amazon Kinesis Data Streams	com.amazonaws.*region*.kinesis-streams
Amazon S3	com.amazonaws.*region*.s3 (Requires a gateway endpoint that is created by default in RES.)
AWS Secrets Manager	com.amazonaws.*region*.secretsmanager
Amazon SES	com.amazonaws.*region*.email-smtp (Not supported in the following Availability Zones: use-1-az2, use1-az3, use1-az5, usw1-az2, usw2-az4, apne2-az4, cac1-az3, and cac1-az4.)
AWS Security Token Service	com.amazonaws.*region*.sts

AWS service	Service name
Amazon SNS	com.amazonaws.*region*.sns
Amazon SQS	com.amazonaws.*region*.sqs
AWS Systems Manager	com.amazonaws.*region*.ec2messages
	com.amazonaws.*region*.ssm
	com.amazonaws.*region*.ssmmessages

Connect to services without VPC endpoints

To integrate with services that do not support VPC endpoints, you can set up a proxy server in a public subnet of your VPC. Follow these steps to create a proxy server with the minimum necessary access for a Research and Engineering Studio deployment using AWS Identity Center as your identity provider.

1. Launch a Linux instance in the public subnet of the VPC you will use for your RES deployment.

   - Linux family – Amazon Linux 2 or Amazon Linux 3

   - Architecture – x86

   - Instance type – t2.micro or higher

   - Security group – TCP on port 3128 from 0.0.0.0/0

2. Connect to the instance to set up a proxy server.

   a. Open the http connection.

   b. Allow connection to the following domains from all relevant subnets:

      - .amazonaws.com (for generic AWS services)

      - .amazoncognito.com (for Amazon Cognito)

      - .awsapps.com (for Identity Center)

      - .signin.aws (for Identity Center)

      - .amazonaws-us-gov.com (for Gov Cloud)

   c. Deny all other connections.

    d.   Activate and start the proxy server.

    e.   Note the PORT on which the proxy server listens.

3.   Configure your route table to allow access to the proxy server.

    a.   Go to your VPC console and identify the route tables for the subnets you will be using for Infrastructure Hosts and VDI hosts.

    b.   Edit route table to allow all incoming connections to go to the proxy server instance created in the previous steps.

    c.   Do this for route tables for all the subnets (without internet access) which you are going to use for Infrastructure/VDIs.

4.   Modify the security group of the proxy server EC2 instance and make sure it allows inbound TCP connections on the PORT on which the proxy server is listening.

Set private VPC deployment parameters

In [the section called "Step 1: Launch the product"](#), you are expected to input certain parameters in the AWS CloudFormation template. Be sure to set the following parameters as noted to successfully deploy into the private VPC you just configured.

Parameter	Input
InfrastructureHostAMI	Use the infrastructure AMI ID created in [the section called "Prepare Amazon Machine Images (AMIs)"](#).
IsLoadBalancerInternetFacing	Set to false.
LoadBalancerSubnets	Choose private subnets without internet access.
InfrastructureHostSubnets	Choose private subnets without internet access.
VdiSubnets	Choose private subnets without internet access.

Parameter	Input
ClientIP	You can choose your VPC CIDR to allow access for all VPC IP addresses.

Create external resources

This CloudFormation stack creates networking, storage, active directory, and domain certificates (if a PortalDomainName is provided). You must have these external resources available to deploy the product.

You may download the recipes template before deployment.

Time to deploy: Approximately 40-90 minutes

1. Sign in to the AWS Management Console and open the AWS CloudFormation console at https://console.aws.amazon.com/cloudformation.

   > ℹ️ **Note**
   >
   > Make sure you are in your administrator account.

2. Launch the template in the console.

   If you are deploying in the AWS GovCloud (US-West) Region, launch the template in the GovCloud partition account.

3. Enter the template parameters:

Parameter	Default	Description
DomainName	`corp.res.com`	Domain used for the active directory. The default value is supplied in the `LDIF` file which sets up bootstrap users. If you would like to use the default users, leave the value as default. To change the value, update

Parameter	Default	Description
		and provide a separate LDIF file. This does not need to match the domain used for active directory.
SubDomain (GovCloud only)		**This parameter is optional for commercial regions, but required for GovCloud regions.** If you provide a SubDomain , the parameter will be prefixed to the DomainNam e provided. The provided Active Directory domain name will become a subdomain.
AdminPassword		The password for the active directory administrator (username Admin). This user is created in the active directory for the initial bootstrapping phase and is not used after. **Note:** The password for this user must meet the [password complexity requirements for active directory]().

Parameter	Default	Description
ServiceAccountPassword		Password used to create a service account (`ReadOnlyUser`). This account is used for synchronization. **Important:** as of Research and Engineering Studio release 2024.06 you must provide a Secret ARN which contains the plaintext password for the ServiceAccount. **Note:** The password for this user must meet the [password complexity requirements for active directory](#).
Keypair		Connects the administrative instances using an SSH client. **Note:** AWS Systems Manager Session Manager can also be used to connect to instances.

Parameter	Default	Description
LDIFS3Path	`aws-hpc-recipes/main/recipes/res/res_demo_env/assets/res.ldif`	The Amazon S3 path to an LDIF file imported during the bootstrapping phase of active directory setup. For more information, see LDIF Support. The parameter prepopulates with a file that creates a number of users in the active directory. To view the file, see the res.ldif file available in GitHub.
ClientIpCidr		The IP address from which you will access the site. For example, you can select your IP address and use `[IPADDRESS]/32` to only allow access from your host. You can update this post-deployment.
ClientPrefixList		Enter a prefix list to provide access to the active directory management nodes. For information on creating a managed prefix list, see Work with customer-managed prefix lists.

Parameter	Default	Description
EnvironmentName	res-*[environment name]*	If the `PortalDomainName` is provided, this parameter is used to add tags to the secrets generated so that they can be used within the environment. This will need to match the `EnvironmentName` parameter used when creating the RES stack. If you are deploying multiple environments in your account, this will need to be unique.
PortalDomainName		**For GovCloud deployments, do not enter this parameter. The certificates and secrets were manually created during the prerequisites.** The domain name in Amazon Route 53 for the account. If this is provided, then a public certificate and key file will be generated and uploaded to AWS Secrets Manager. If you have your own domain and certificates, this parameter and `EnvironmentName` can be left blank.

4. Acknowledge all checkboxes in **Capabilities**, and choose **Create stack**.

Step 1: Launch the product

Follow the step-by-step instructions in this section to configure and deploy the product into your account.

Time to deploy: Approximately 60 minutes

You can download the CloudFormation template for this product before deploying it.

If you are deploying in AWS GovCloud (US-West), use this template.

res-stack - Use this template to launch the product and all associated components. The default configuration deploys the RES main stack and authentication, frontend, and backend resources.

> ⓘ **Note**
>
> AWS CloudFormation resources are created from AWS Cloud Development Kit (AWS CDK) (AWS CDK) constructs.

The AWS CloudFormation template deploys Research and Engineering Studio on AWS in the AWS Cloud. You must meet the prerequisites before launching the stack.

1. Sign in to the AWS Management Console and open the AWS CloudFormation console at https://console.aws.amazon.com/cloudformation.

2. Launch the template .

   To deploy in AWS GovCloud (US-West), launch this template.

3. The template launches in the US East (N. Virginia) Region by default. To launch the solution in a different AWS Region, use the Region selector in the console navigation bar.

   > ⓘ **Note**
   >
   > This product uses the Amazon Cognito service, which is not currently available in all AWS Regions. You must launch this product in an AWS Region where Amazon Cognito is available. For the most current availability by Region, see the AWS Regional Services List.

4. Under **Parameters**, review the parameters for this product template and modify them as necessary. If you deployed the automated external resources, you can find these parameters in the **Outputs** tab of the external resources stack.

Parameter	Default	Description
EnvironmentName	*<res-demo>*	A unique name given to your RES environment starting with res- and no longer than 11 characters.
AdministratorEmail		The email address for the user completing setup of the product. This user additionally functions as a break-glass user if there is an active directory single sign on integration failure.
InfrastructureHostAMI	ami-*[numbers or letters only]*	*(Optional)* You may provide a custom AMI id to use for all the infrastructure hosts. The current supported base OS is Amazon Linux 2. For more information, see [Configure RES-ready AMIs](#).
SSHKeyPair		The key pair used to connect to infrastructure hosts.
ClientIP	*x.x.x*.0/24 or *x.x.x*.0/32	IP address filter which limits connection to the system. You can update the ClientIpCidr after deployment.

Parameter	Default	Description
ClientPrefixList		*(Optional)* Provide a managed prefix list for IPs allowed to directly access the web UI and SSH into the bastion host.
IAMPermissionBoundary		*(Optional)* You may provide a managed policy ARN that will be attached as a permission boundary to all roles created in RES. For more information, see [Setting custom permission boundaries](#).
VpcId		IP for the VPC where instances will launch.
IsLoadBalancerInternetFacing		Select true to deploy internet facing load balancer (Requires public subnets for load balancer). For deployments that need restricted internet access, select false.

Parameter	Default	Description
LoadBalancerSubnets		Select at least two subnets in different Availability Zones where load balancers will launch. For deployments that need restricted internet access, choose private subnets. For deployments that need internet access, choose public subnets. If more than two were created by the external networking stack, select all that were created.
InfrastructureHostSubnets		Select at least two private subnets in different Availability Zones where infrastructure hosts will launch. If more than two were created by the external networking stack, select all that were created.
VdiSubnets		Select at least two private subnets in different Availability Zones where VDI instances will launch. If more than two were created by the external networking stack, select all that were created.

Parameter	Default	Description
ActiveDirectoryName	*corp.res.com*	Domain for the active directory. It does not need to match the portal domain name.
ADShortName	*corp*	The short name for the active directory. This is also called the NetBIOS name.
LDAP Base	*DC=corp,DC=res,DC=com*	An LDAP path to the base within the LDAP hierarchy.
LDAPConnectionURI		A single ldap:// path that can be reached by the active directory's host server. If you deployed the automated external resources with the default AD domain, you can use ldap://corp.res.com.
ServiceAccountUserName	ServiceAccount	Username for a service account used to connect to AD. This account must have access to create computers within the ComputersOU.
ServiceAccountPasswordSecretArn		Provide a Secret ARN which contains the plaintext password for the ServiceAccount.
UsersOU		Organizational unit within AD for users that will sync.
GroupsOU		Organizational unit within AD for groups that will sync.

Parameter	Default	Description
SudoersOU		Organizational unit within AD for global sudoers.
SudoersGroupName	RESAdministrators	Group name that contains all users with sudoer access on instances at install and administrator access on RES.
ComputersOU		Organizational unit within AD that instances will join.
DomainTLSCertificateSecretARN		*(Optional)* Provide a domain TLS certificate secret ARN to enable TLS communication to AD.
EnableLdapIDMapping		Determines if UID and GID numbers are generated by SSSD or if the numbers provided by the AD are used. Set to True to use SSSD generated UID and GID, or False to use UID and GID provided by the AD. For most cases this parameter should be set to True.
DisableADJoin	False	To prevent Linux hosts from joining the directory domain, change to True. Otherwise, leave in the default setting of False.
ServiceAccountUserDN		Provide the distinguished name (DN) of the service account user in Directory.

Parameter	Default	Description
SharedHomeFilesystemID		An EFS ID to use for the shared home filesystem for Linux VDI hosts.
CustomDomainNamefo rWebApp		*(Optional)* Subdomain used by the web portal to provide links for the web portion of the system.
CustomDomainNameforVDI		*(Optional)* Subdomain used by the web portal to provide links for the VDI portion of the system.
ACMCertificateARNf orWebApp		*(Optional)* When using the default configuration, the product hosts the web application under the domain amazonaws .com. You may host the product services under your domain. If you deployed the automated external resources, this was generated for you and the information can be found in the Outputs of the res-bi stack. If you need to generate a certificate for your web application, see [Configuration guide](#).

Parameter	Default	Description
CertificateSecretARNforVDI		*(Optional)* This ARN secret stores the public certifica te for your web portal's public certificate. If you set a portal domain name for your automated external resources, you can find this value under the Outputs tab of the res-bi stack.
PrivateKeySecretARNforVDI		*(Optional)* This ARN secret stores the private key for your web portal's certificate. If you set a portal domain name for your automated external resources, you can find this value under the Outputs tab of the res-bi stack.

5.   Choose **Create stack** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should receive a CREATE_COMPLETE status in approximately 60 minutes.

Step 2: Sign in for the first time

Once the product stack has deployed in your account, you will receive an email with your credentials. Use the URL to sign in to your account and configure the workspace for other users.
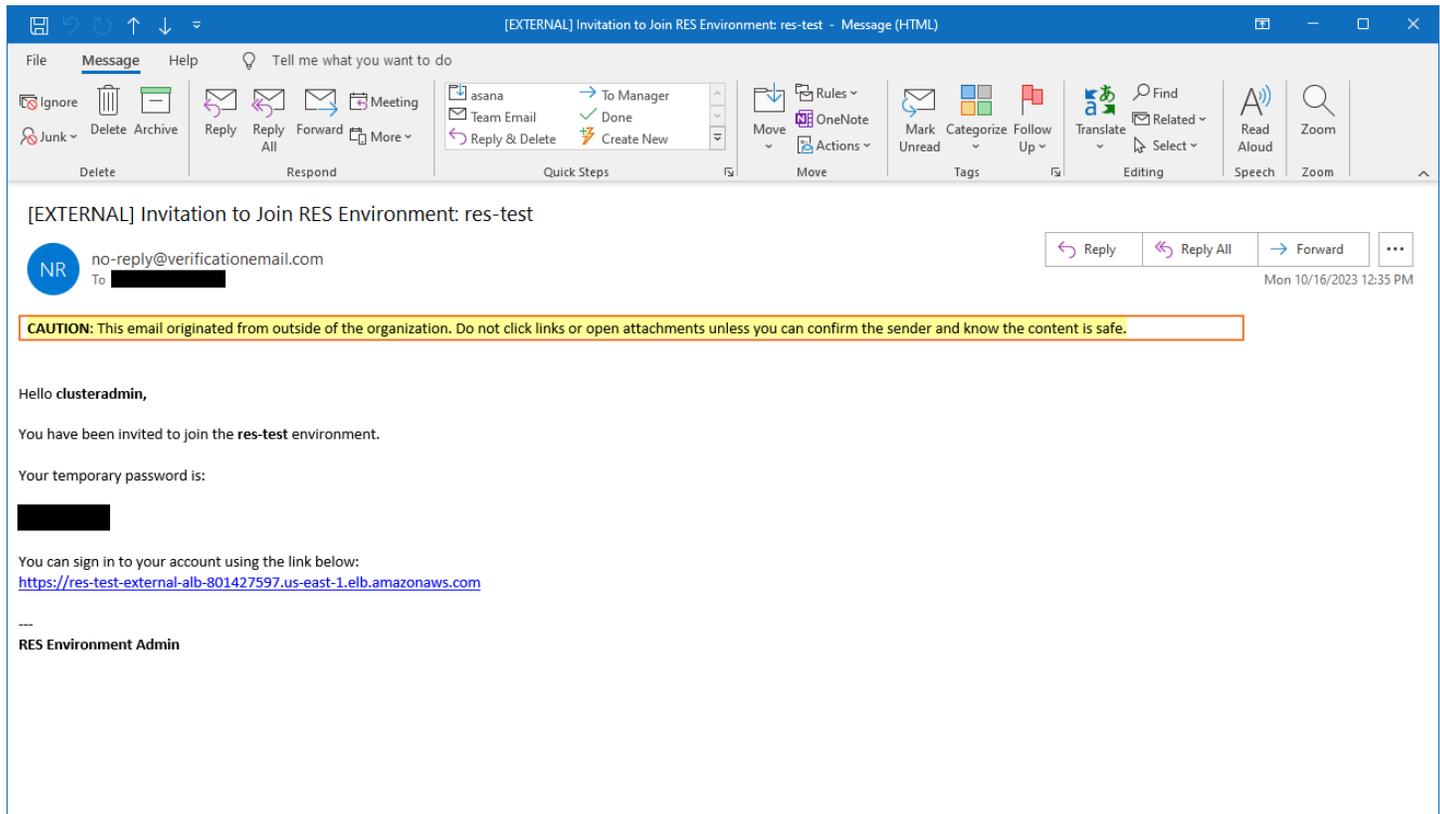
Once you have signed in for the first time, you can configure settings in the web portal to connect to the SSO provider. For post-deployment configuration information, see the *Configuration guide*.

Update the product

Research and Engineering Studio (RES) has two methods of updating the product which depend on if the version update is major or minor.

RES uses a date-based versioning scheme. A major release uses the year and month, and a minor release adds a sequence number when necessary. For example, version 2024.01 was released in January 2024 as a major release; version 2024.01.01 was a minor release update of that version.

Topics

- [Major version updates](#)
- [Minor version updates](#)

Major version updates

Research and Engineering Studio uses snapshots to support migration from a previous RES environment to the latest without losing your environment settings. You can also use this process to test and verify updates to your environment before onboarding users.

To update your environment with the latest version of RES:

1. Create a snapshot of your current environment. See [the section called "Create a snapshot"](#).

2. Redeploy RES with the new version. See [the section called "Step 1: Launch the product"](#).

3. Apply the snapshot to your updated environment. See [the section called "Apply a snapshot"](#).

4. Verify all data migrated successfully to the new environment.

Minor version updates

For minor version updates to RES, a new install is not required. You can update the existing RES stack by updating its AWS CloudFormation template. Check the version of your current RES environment in AWS CloudFormation before deploying the update. You can find the version number at the beginning of the template.

For example: `"Description": "RES_2024.1"`

To make a minor version update:

1. Download the latest AWS CloudFormation template in [the section called "Step 1: Launch the product"](#).

2. Open the AWS CloudFormation console at https://console.aws.amazon.com/cloudformation.

3. From **Stacks**, find and select the primary stack. It should appear as <code><stack-name></code>.

4. Choose **Update**.

5. Choose **Replace current template**.

6. For **Template source**, choose **Upload a template file**.

7. Choose **Choose file** and upload the template you downloaded.

8. On **Specify stack details**, choose **Next**. You do not need to update the parameters.

9. On **Configure stack options**, choose **Next**.

10. On **Review <stack-name>**, choose **Submit**.

Uninstall the product

You can uninstall the Research and Engineering Studio on AWS product from the AWS Management Console or by using the AWS Command Line Interface. You must manually delete the Amazon Simple Storage Service (Amazon S3) buckets created by this product. This product does not automatically delete <EnvironmentName>-shared-storage-security-group in case you have stored data to retain.

Using the AWS Management Console

1. Sign in to the AWS CloudFormation console.

2. On the **Stacks** page, select this product's installation stack.

3. Choose **Delete**.

Using AWS Command Line Interface

Determine whether the AWS Command Line Interface (AWS CLI) is available in your environment. For installation instructions, see What Is the AWS Command Line Interface in the *AWS CLI User Guide*. After confirming that the AWS CLI is available and configured to the administrator account in the Region where the product was deployed, run the following command.

```
$ aws cloudformation delete-stack --stack-name
<RES-stack-name>
```

Deleting the shared-storage-security-group

> ⚠ **Warning**
>
> The product retains this file system by default to protect against unintentional data loss. If you choose to delete the security group and associated file systems, any data retained within those systems will be permanently deleted. We recommend backing up data or reassigning the data to a new security group.

1.   Sign in to the AWS Management Console and open the Amazon EFS console at https://
     console.aws.amazon.com/efs/.

2.   Delete all file systems associated with *<RES-stack-name>*-shared-storage-security-group.
     Alternatively, you may reassign these file systems to another security group to maintain the
     data.

3.   Sign in to the AWS Management Console and open the Amazon EC2 console at https://
     console.aws.amazon.com/ec2/.

4.   Delete the *<RES-stack-name>*-shared-storage-security-group.

Deleting the Amazon S3 buckets

This product is configured to retain the product-created Amazon S3 bucket (for deploying in an
opt-in Region) if you decide to delete the AWS CloudFormation stack to prevent accidental data
loss. After uninstalling the product, you can manually delete this S3 bucket if you do not need to
retain the data. Follow these steps to delete the Amazon S3 bucket.

1.   Sign in to the AWS Management Console and open the Amazon S3 console at https://
     console.aws.amazon.com/s3/.

2.   Choose **Buckets** from the navigation pane.

3.   Locate the `stack-name` S3 buckets.

4.   Select each Amazon S3 bucket, then choose **Empty**. You must empty each bucket.

5.   Select the S3 bucket and choose **Delete**.

To delete S3 buckets using AWS CLI, run the following command:

```
$ aws s3 rb s3://<bucket-name> --force
```

> **ⓘ Note**
>
> The `--force` command empties the bucket of its contents.

Configuration guide

This configuration guide provides post-deployment instructions for a technical audience on how to further customize and integrate with the Research and Engineering Studio on AWS product.

Topics

- [Managing users and groups](#)

- [Creating subdomains](#)

- [Create an ACM certificate](#)

- [Amazon CloudWatch Logs](#)

- [Setting custom permission boundaries](#)

- [Configure RES-ready AMIs](#)

Managing users and groups

Research and Engineering Studio can use any SAML 2.0 compliant identity provider. If you deployed RES using the external resources or plan to use IAM Identity center, see [the section called "Setting up SSO with IAM Identity Center"](#). If you have your own SAML 2.0 compliant identity provider, see [the section called "Configuring your identity provider for single sign-on (SSO)"](#).

Topics

- [Setting up SSO with IAM Identity Center](#)

- [Configuring your identity provider for single sign-on (SSO)](#)

- [Setting passwords for users](#)

Setting up SSO with IAM Identity Center

If you do not already have an identity center connected to the managed active directory, start with [the section called "Set up an identity center"](#). If you already have an identity center connected with the managed active directory, start with [the section called "Connect to an identity center"](#).

> ⓘ **Note**
>
> If you are deploying to the AWS GovCloud (US-West) Region, set up SSO in the AWS
> GovCloud (US) partition account where you deployed Research and Engineering Studio.

Step 1: Set up an identity center

Enabling identity center

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.

2. Open the **Identity Center**.

3. Choose **Enable**.

4. Choose **Enable with AWS Organizations**.

5. Choose **Continue**.

> ⓘ **Note**
>
> Make sure you are in the same Region where you have your managed active directory.

Connecting identity center to managed active directory

After enabling identity center, complete these recommended set up steps:

1. From the navigation, choose **Settings**.

2. Under **Identity source**, choose **Actions** and choose **Change identity source**.

3. Under **Existing directories**, select your directory.

4. Choose **Next**.

5. Review your changes and enter **ACCEPT** in the confirmation box.

6. Choose **Change identity source**.

Syncing users and groups to identity center

Once the changes from [the section called "Connecting identity center to managed active directory"](#) complete, a green banner should appear.

1. In the confirmation banner, choose **Start guided setup**.

2. From **Configure attribute mappings**, choose **Next**.

3. Under the **User** section, enter the users you want to sync.

4. Choose **Add**.

5. Choose **Next**.

6. Review your changes and choose **Save configuration**.

7. The sync process may take a few minutes. If you receive a warning message about users not syncing, choose **Resume sync**.

Enabling users

1. From the menu, choose **Users**.

2. Choose the user(s) for which you want to enable access.

3. Choose **Enable user access**.

Step 2: Connect to an identity center

Setting up the application in Identity Center

1. Sign in to the AWS Management Console and open the IAM Identity Center at https://console.aws.amazon.com/singlesignon/.

2. Choose **Applications**.

3. Choose **Add application**.

4. Under **Setup preference**, choose **I have an application I want to set up**.

5. Under **Application type**, choose **SAML 2.0**.

6. Choose **Next**.

7. Enter the display name and description you would like to use.

8. Under **IAM Identity Center metadata**, copy the link for the **IAM Identity Center SAML metadata** file. You will need this when configuring the SSO with the RES portal.

9. Under **Application properties**, enter your **Application start URL**. For example, <your-portal-domain>/sso.

10. Under **Application ACS URL**, enter the redirect URL from the RES portal. To find this:

    a. Under **Environment management**, choose **General settings**.

    b. Choose the **Identity provider** tab.

    c. Under **Single Sign-On**, you will find the **SAML Redirect URL**.

11. Under **Application SAML audience**, enter the Amazon Cognito URN. To create the urn:

    a. From the RES portal, open **General Settings**.

    b. Once the **Identity provider** tab, locate the **User Pool ID**.

    c. Add the **User Pool ID** to this string:

    ```
    urn:amazon:cognito:sp:<user_pool_id>
    ```

12. Choose **Submit**.

Configuring attribute mappings for the application

1. From the **Identity Center**, open the details for your created application.

2. Choose **Actions** and choose **Edit attribute mappings**.

3. Under **Subject**, enter ${user:email}.

4. Under **Format**, choose emailAddress.

5. Choose **Add new attribute mapping**.

6. Under **User attribute in the application**, enter email.

7. Under **Maps to this string value or user attribute in IAM Identity Center**, enter ${user:email}.

8. Under **Format**, enter unspecified.

9. Choose **Save changes**.

Adding users to the application in Identity Center

1. From the Identity Center, open **Assigned users** for your created application and choose **Assign users**.

2. Select the users you want to assign application access.

3. Choose **Assign users**.

Setting up SSO within the RES environment

1. From the Research and Engineering Studio environment, open **General settings** under **Environment management**.

2. Open the **Identity provider** tab.

3. Under **Single Sign-On**, choose the edit button next to **Status**.

4. Complete the form with the following information:

   a. Choose **SAML**.

   b. Under **Provider name**, enter a user friendly name.

   c. Select **Enter metadata document endpoint URL**.

   d. Enter the URL you copied during the section called "Setting up the application in Identity Center"

   e. Under **Provider email attribute**, enter email.

   f. Choose **Submit**.

5. Refresh the page and check that the **Status** displays as enabled.

Configuring your identity provider for single sign-on (SSO)

Research and Engineering Studio integrates with any SAML 2.0 identity provider to authenticate user access to the RES portal. These steps provide directions to integrate with your chosen SAML 2.0 identity provider. If you intend to use IAM Identity Center, please see the section called "Setting up SSO with IAM Identity Center".

> **ⓘ Note**
>
> The user's email must match in the IDP SAML assertion and Active Directory. You will need to connect your identity provider with your Active Directory and periodically sync users.

Topics

- Configure your identity provider
- Configure RES to use your identity provider

- [Configuring your identity provider in a non-production environment](#)

- [Debugging SAML IdP issues](#)

Configure your identity provider

This section provides the steps to configure your identity provider with information from the RES Amazon Cognito user pool.

1. RES assumes that you have an AD (AWS Managed AD or a self-provisioned AD) with the user identities allowed to access the RES portal and projects. Connect your AD to your identity service provider and sync the user identities. Check your identity provider's documentation to learn how to connect your AD and sync user identities. For example, see [Using Active Directory as an identity source](#) in the *AWS IAM Identity Center User Guide*.

2. Configure a SAML 2.0 application for RES in your identity provider (IdP). This configuration requires the following parameters:

   - **SAML Redirect URL** — The URL that your IdP uses to send the SAML 2.0 response to the service provider.

     > **ⓘ Note**
     >
     > Depending on the IdP, the SAML Redirect URL might have a different name:
     >
     > - Application URL
     >
     > - Assertion Consumer Service (ACS) URL
     >
     > - ACS POST Binding URL

     To get the URL

     1. Sign in to RES as an **admin** or **clusteradmin**.

     2. Navigate to **Environment Management** ⇒ **General Settings** ⇒ **Identity Provider**.

     3. Choose **SAML Redirect URL**.

   - **SAML Audience URI** — The unique ID of the SAML audience entity on the service provider side.

> **ⓘ Note**
>
> Depending on the IdP, the SAML Audience URI might have a different name:
>
> - ClientID
>
> - Application SAML Audience
>
> - SP entity ID

Provide the input in the following format.

```
urn:amazon:cognito:sp:user-pool-id
```

To find your SAML Audience URI

1. Sign in to RES as an **admin** or **clusteradmin**.

2. Navigate to **Environment Management** ⇒ **General Settings** ⇒ **Identity Provider**.

3. Choose **User Pool Id**.

3. The SAML assertion posted to RES must have the following fields/claims set to the user's email address:

- SAML Subject or NameID

- SAML email

4. Your IdP adds fields/claims to the SAML assertion, based on the configuration. RES requires these fields. Most providers automatically fill these fields by default. Refer to the following field inputs and values if you have to configure them.

- **AudienceRestriction** — Set to `urn:amazon:cognito:sp:`*`user-pool-id`*. Replace *user-pool-id* with the ID of your Amazon Cognito user pool.

  ```
  <saml:AudienceRestriction>
      <saml:Audience> urn:amazon:cognito:sp:user-pool-id
  </saml:AudienceRestriction>
  ```

- **Response** — Set `InResponseTo` to `https://`*`user-pool-domain`*`/saml2/idpresponse`. Replace *user-pool-domain* with the domain name of your Amazon Cognito user pool.

```
<saml2p:Response
  Destination="http://user-pool-domain/saml2/idpresponse"
  ID="id123"
  InResponseTo="_dd0a3436-bc64-4679-a0c2-cb4454f04184"
  IssueInstant="Date-time stamp"
  Version="2.0"
  xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

- **SubjectConfirmationData** — Set `Recipient` to your user pool `saml2/idpresponse` endpoint and `InResponseTo` to the original SAML request ID.

```
<saml2:SubjectConfirmationData
  InResponseTo="_dd0a3436-bc64-4679-a0c2-cb4454f04184"
  NotOnOrAfter="Date-time stamp"
  Recipient="https://user-pool-domain/saml2/idpresponse"/>
```

- **AuthnStatement** — Configure as the following:

```
<saml2:AuthnStatement AuthnInstant="2016-10-30T13:13:28.152TZ"
  SessionIndex="32413b2e54db89c764fb96ya2k"
 SessionNotOnOrAfter="2016-10-30T13:13:28">
    <saml2:SubjectLocality />
    <saml2:AuthnContext>

 <saml2:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:Password</
saml2:AuthnContextClassRef>
    </saml2:AuthnContext>
</saml2:AuthnStatement>
```

5. If your SAML application has a logout URL field, set it to: <*domain-url*>/saml2/logout.

To get the domain URL

1. Sign in to RES as an **admin** or **clusteradmin**.

2. Navigate to **Environment Management** ⇒ **General Settings** ⇒ **Identity Provider**.

3. Choose **Domain URL**.

6. If your IdP accepts a signing certificate to establish trust with Amazon Cognito, download the Amazon Cognito signing certificate and upload it in your IdP.

To get the signing certificate

1. Open the Amazon Cognito console in the [Getting Started with the AWS Management Console](#)

2. Select your user pool. Your user pool should be res-*<environment name>*-user-pool.

3. Choose the **Sign-in experience** tab.

4. In the **Federated identity provider sign-in** section, choose **View signing certificate**.

Cognito user pool sign-in Info

Users can sign in using their email address, phone number, or user name. User attributes, group memberships, and security settings will be stored and configured in your user pool.

Cognito user pool sign-in options
User name
Email

User name requirements
User names are not case sensitive

Federated identity provider sign-in (1) Info

Your app users can sign-in through external social identity providers like Facebook, Google, Amazon, or Apple, and through your on-prem directories via SAML or Open ID Connect.

[Search identity providers by name]   ‹ 1 › ⚙

	Identity provider ▲	Identity provider type ▽	Created time ▽	Last updated time ▽
○	idc	SAML	2 weeks ago	3 hours ago

You can use this certificate to set up Active Directory IDP, add a `relying party trust`, and enable SAML support on this relying party.

> ⓘ **Note**
>
> This doesn't apply to Keycloak and IDC.

5. After the application setup is complete, download the SAML 2.0 application metadata XML or URL. You use it in the next section.

Configure RES to use your identity provider

To complete the single sign-on setup for RES

1. Sign in to RES as an **admin** or **clusteradmin**.

2. Navigate to **Environment Management** ⇒ **General Settings** ⇒ **Identity Provider**.

3. Under **Single Sign-On**, choose the edit icon next to the status indicator to open the **Single Sign On Configuration** page.

a. For **Identity Provider**, choose **SAML**.

b. For **Provider Name**, enter a unique name for your identity provider.

> ℹ️ **Note**
>
> The following names are not allowed:
>
> - Cognito
> - IdentityCenter

    c.   Under **Metadata Document Source**, choose the appropriate option and upload the metadata XML document or provide the URL from the identity provider.

    d.   For **Provider Email Attribute**, enter the text value `email`.

    e.   Choose **Submit**.

4.   Reload the **Environment Settings** page. Single sign-on is enabled if the configuration was correct.

Configuring your identity provider in a non-production environment

If you used the provided [external resources](#) to create a non-production RES environment and configured IAM Identity Center as your identity provider, you may want to configure a different identity provider such as Okta. The RES SSO enablement form asks for three configuration parameters:

1. Provider name — Cannot be modified
2. Metadata document or URL — Can be modified
3. Provider email attribute — Can be modified

To modify the metadata document and provider email attribute, do the following:

1.   Go to the Amazon Cognito console.

2.   From the navigation, choose **User pools**.

3.   Choose your user pool to view the **User pool overview**.

4.   From the **Sign-in experience** tab, go to **Federated identity provider sign-in** and open your configured identity provider.

5.   Generally, you will only be required to change the metadata and leave the attribute mapping unchanged. To update **Attribute mapping**, choose **Edit**. To update the **Metadata document**, choose **Replace metadata**.

Attribute mapping (1) Info   Edit

View, add, and edit attribute mappings between SAML and your user pool.

‹ 1 › ⚙

User pool attribute | **SAML attribute**

email   email

Metadata document Info   Replace metadata

View and update your SAML metadata. This document is issued by your SAML provider. It includes the issuer's name, expiration information, and keys that can be used to validate the response from the identity provider.

Metadata document source   **Metadata document endpoint URL**
Enter metadata document endpoint URL   https://portal.sso.us-west-2.amazonaws.com/saml/metadata
                                                     /MDg4ODM3NTczNjY0X2lucy04M2EyYTcyMGUzZTFlMDI4

---

6. If you edited the attribute mapping, you will need to update the `<environment name>.cluster-settings` table in DynamoDB.

    a. Open the DynamoDB console and choose **Tables** from the navigation.

    b. Find and select the `<environment name>.cluster-settings` table, and from the **Actions** menu choose **Explore items**.

    c. Under **Scan or query items**, go to **Filters** and enter the following parameters:

        - **Attribute name** — `key`

        - **Value** — `identity-provider.cognito.sso_idp_provider_email_attribute`

    d. Choose **Run**.

7. Under **Items returned**, find the `identity-provider.cognito.sso_idp_provider_email_attribute` string and choose **Edit** to modify the string to match your changes in Amazon Cognito.

▼ **Scan or query items**

◉ Scan

○ Query

Select a table or index

Table - res-jan19.cluster-settings ▼

Select attribute projection

All attributes ▼

▼ **Filters** **6**

Attribute name

🔍 key ✕

Type

String ▼

Condition

Equal to ▼

Value

identity-provider

Remove

Add filter

7

Run **Reset**

✓ Completed. Read capacity units consumed: 13 ✕

Items returned (1)

Actions ▼ Create item

Edit String ✕

email

Enter any string value.

Cancel Save

8

◁ 1 ▷ ⚙ ⛶

☐ | key *(String)*

☐ identity-provider.cognito.ss

▽ | version ▽

✎ 1

Debugging SAML IdP issues

SAML-tracer — You can use this extension for the Chrome browser to track SAML requests and check the SAML assertion values. For more information, see [SAML-tracer](#) at the Chrome web store.

SAML developer tools — OneLogin provides tools that you can use to decode the SAML encoded value and check the required fields in the SAML assertion. For more information, see [Base 64 Decode + Inflate](#) at the OneLogin web site.

Amazon CloudWatch Logs — You can check your RES logs in CloudWatch Logs for errors or warnings. Your logs are in a log group with the name format *res-environment-name*/cluster-manager.

Amazon Cognito documentation — For more information about SAML integration with Amazon Cognito, see [Adding SAML identity providers to a user pool](#) in the *Amazon Cognito Developer Guide*.

Setting passwords for users

1. From the [AWS Directory Service console,](#) select the directory for the created stack.

2. Under the **Actions** menu, choose **Reset user password**.

3. Choose the user and enter a new password.

4. Choose **Reset password**.

Creating subdomains

If you are using a custom domain, you will need to set up subdomains to support the web and VDI portions of your portal.

> ⓘ **Note**
>
> If you are deploying to the AWS GovCloud (US-West) Region, set up the web application and VDI subdomains in the commercial partition account hosting the domain public hosted zone.

1. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

2. Find the domain you created and choose **Create record**.

3. Enter web as the **Record name**.

4. Choose **CNAME** as the **Record type**.

5. For **Value**, enter the link you received in the initial email.

6. Choose **Create records**.

7. To create a record for the VDC, retrieve the NLB address.

a. Sign in to the AWS Management Console and open the AWS CloudFormation console at https://console.aws.amazon.com/cloudformation.

b. Choose <environment-name>-vdc.

c. Choose **Resources** and open <environmentname>-vdc-external-nlb.

d. Copy the DNS name from the NLB.

8. Sign in to the AWS Management Console and open the Route 53 console at https://console.aws.amazon.com/route53/.

9. Find your domain and choose **Create record**.

10. Under **Record name**, enter vdc.

11. Under **Record type**, select **CNAME**.

12. For the NLB, enter the DNS.

13. Choose **Create record**.

Create an ACM certificate

By default, RES hosts the web portal under an application load balancer using the domain amazonaws.com. To use your own domain, you will need to configure a public SSL/TLS certificate provided by you or requested from AWS Certificate Manager (ACM). If you use ACM, you will receive an AWS resource name you will need to provide as a parameter to encrypt the SSL/TLS channel between the client and web services host.

> ⓘ **Tip**
>
> If you are deploying the external resources demo package, you will need to enter your chosen domain in PortalDomainName when deploying the external resources stack in the section called "Create external resources".

To create a certificate for custom domains:

1. From the console, open AWS Certificate Manager to request a public certificate. If you are deploying in AWS GovCloud (US-West), create the certificate in your GovCloud partition account.

2. Choose **Request a public certificate**, and choose **Next**.

3. Under **Domain names**, request a certificate for both `*.PortalDomainName` and `PortalDomainName`.

4. Under **Validation method**, choose **DNS validation**.

5. Choose **Request**.

6. From the **Certificates** list, open your requested certificates. Each certificate will have **Pending validation** as the status.

> ⓘ **Note**
>
> If you do not see your certificates, refresh the list.

7. Do one of the following:

    - **Commercial deployment:** From the **Certificate details** for each requested certificate, choose **Create records in Route 53**. The status of the certificate should change to **Issued**.

    - **GovCloud deployment:** If you are deploying in AWS GovCloud (US-West), copy the CNAME key and value. From the commercial partition account, use the values to create a new record in the Public Hosted Zone. The status of the certificate should change to **Issued**.

8. Copy the new certificate ARN to input as the parameter for `ACMCertificateARNforWebApp`.

Amazon CloudWatch Logs

Research and Engineering Studio creates the following log groups in CloudWatch during installation. See the following table for default retentions:

CloudWatch Log groups	Retention
/aws/lambda/<installation-stack-name>-cluster-endpoints	Never expire
/aws/lambda/<installation-stack-name>-cluster-manager-scheduled-ad-sync	Never expire
/aws/lambda/<installation-stack-name>-cluster-settings	Never expire

CloudWatch Log groups	Retention
/aws/lambda/<installation-stack-name>-oauth-credentials	Never expire
/aws/lambda/<installation-stack-name>-self-signed-certificate	Never expire
/aws/lambda/<installation-stack-name>-update-cluster-prefix-list	Never expire
/aws/lambda/<installation-stack-name>-vdc-scheduled-event-transformer	Never expire
/aws/lambda/<installation-stack-name>-vdc-update-cluster-manager-client-scope	Never expire
/<installation-stack-name>/cluster-manager	3 months
/<installation-stack-name>/vdc/controller	3 months
/<installation-stack-name>/vdc/dcv-broker	3 months
/<installation-stack-name>/vdc/dcv-connection-gateway	3 months

If you would like to change the default retention for a log group, you can go to the CloudWatch console at https://console.aws.amazon.com/cloudwatch/ and follow the directions to *Change log data retention in CloudWatch Logs*.

Setting custom permission boundaries

As of 2024.04, you can optionally modify roles created by RES by attaching custom permission boundaries. A custom permission boundary may be defined as part of the RES AWS CloudFormation installation by supplying the permission boundary's ARN as part of the IAMPermissionBoundary parameter. No permission boundary is set on any RES roles if this parameter is left empty. Below is the list of actions that RES roles require to operate. Make sure that any permission boundary that you plan to use explicitly allows for the following actions:

```
[
    {
        "Effect": "Allow",
        "Resource": "*",
        "Sid": "ResRequiredActions",
        "Action": [
            "access-analyzer:*",
            "account:GetAccountInformation",
            "account:ListRegions",
            "acm:*",
            "airflow:*",
            "amplify:*",
            "amplifybackend:*",
            "amplifyuibuilder:*",
            "aoss:*",
            "apigateway:*",
            "appflow:*",
            "application-autoscaling:*",
            "appmesh:*",
            "apprunner:*",
            "aps:*",
            "athena:*",
            "auditmanager:*",
            "autoscaling-plans:*",
            "autoscaling:*",
            "backup-gateway:*",
            "backup-storage:*",
            "backup:*",
            "batch:*",
            "bedrock:*",
            "budgets:*",
            "ce:*",
            "cloud9:*",
            "cloudformation:*",
            "cloudfront:*",
            "cloudtrail-data:*",
            "cloudtrail:*",
            "cloudwatch:*",
            "codeartifact:*",
            "codebuild:*",
            "codeguru-profiler:*",
            "codeguru-reviewer:*",
            "codepipeline:*",
```

```
            "codestar-connections:*",
            "codestar-notifications:*",
            "codestar:*",
            "cognito-identity:*",
            "cognito-idp:*",
            "cognito-sync:*",
            "comprehend:*",
            "compute-optimizer:*",
            "cur:*",
            "databrew:*",
            "datapipeline:*",
            "datasync:*",
            "dax:*",
            "detective:*",
            "devops-guru:*",
            "dlm:*",
            "dms:*",
            "drs:*",
            "dynamodb:*",
            "ebs:*",
            "ec2-instance-connect:*",
            "ec2:*",
            "ec2messages:*",
            "ecr:*",
            "ecs:*",
            "eks:*",
            "elastic-inference:*",
            "elasticache:*",
            "elasticbeanstalk:*",
            "elasticfilesystem:*",
            "elasticloadbalancing:*",
            "elasticmapreduce:*",
            "elastictranscoder:*",
            "es:*",
            "events:*",
            "firehose:*",
            "fis:*",
            "fms:*",
            "forecast:*",
            "fsx:*",
            "geo:*",
            "glacier:*",
            "glue:*",
            "grafana:*",
```

```
            "guardduty:*",
            "health:*",
            "iam:*",
            "identitystore:*",
            "imagebuilder:*",
            "inspector2:*",
            "inspector:*",
            "internetmonitor:*",
            "iot:*",
            "iotanalytics:*",
            "kafka:*",
            "kafkaconnect:*",
            "kinesis:*",
            "kinesisanalytics:*",
            "kms:*",
            "lambda:*",
            "lightsail:*",
            "logs:*",
            "memorydb:*",
            "mgh:*",
            "mobiletargeting:*",
            "mq:*",
            "neptune-db:*",
            "organizations:DescribeOrganization",
            "osis:*",
            "personalize:*",
            "pi:*",
            "pipes:*",
            "polly:*",
            "qldb:*",
            "quicksight:*",
            "rds-data:*",
            "rds:*",
            "redshift-data:*",
            "redshift-serverless:*",
            "redshift:*",
            "rekognition:*",
            "resiliencehub:*",
            "resource-groups:*",
            "route53:*",
            "route53domains:*",
            "route53resolver:*",
            "rum:*",
            "s3:*",
```

```
                "sagemaker:*",
                "scheduler:*",
                "schemas:*",
                "sdb:*",
                "secretsmanager:*",
                "securityhub:*",
                "serverlessrepo:*",
                "servicecatalog:*",
                "servicequotas:*",
                "ses:*",
                "signer:*",
                "sns:*",
                "sqs:*",
                "ssm:*",
                "ssmmessages:*",
                "states:*",
                "storagegateway:*",
                "sts:*",
                "support:*",
                "tag:GetResources",
                "tag:GetTagKeys",
                "tag:GetTagValues",
                "textract:*",
                "timestream:*",
                "transcribe:*",
                "transfer:*",
                "translate:*",
                "vpc-lattice:*",
                "waf-regional:*",
                "waf:*",
                "wafv2:*",
                "wellarchitected:*",
                "wisdom:*",
                "xray:*"
        ]
    }
]
```

Configure RES-ready AMIs

With RES-ready AMIs, you can pre-install RES dependencies for virtual desktop instances (VDIs) on your custom AMIs. Using RES-ready AMIs improve boot times for VDI instances using the pre-baked

images. Using EC2 Image Builder, you can build and register your AMIs as new software stacks. For more information on Image Builder, see the Image Builder User Guide.

Before you begin, you must deploy the latest version of RES.

Topics

- Prepare IAM role to access RES environment
- Create EC2 Image Builder component
- Prepare your EC2 Image Builder recipe
- Configure EC2 Image Builder infrastructure
- Configure Image Builder image pipeline
- Run Image Builder image pipeline
- Register a new software stack in RES

Prepare IAM role to access RES environment

To access the RES environment service from EC2 Image Builder, you must create or modify an IAM role called RES-EC2InstanceProfileForImageBuilder. For information on configuring an IAM role for use in Image Builder, see AWS Identity and Access Management (IAM) in the *Image Builder User Guide*.

Your role requires:

- Trusted relationships include the Amazon EC2 service
- AmazonSSMManagedInstanceCore and EC2InstanceProfileForImageBuilder policies
- Custom RES policy with limited DynamoDB and Amazon S3 access to the deployed RES environment

  (This policy can be either a customer managed or customer inline policy document.)

Trusted relationship entity:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
```

```
                "Principal": {
                    "Service": "ec2.amazonaws.com"
                }
                "Action": "sts:AssumeRole"
            }
        ]
    }
```

RES policy:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RESDynamoDBAccess",
            "Effect": "Allow",
            "Action": "dynamodb:GetItem",
            "Resource": "arn:aws:dynamodb:{AWS-Region}:{AWS-Account-ID}:table/{RES-
EnvironmentName}.cluster-settings",
            "Condition": {
                "ForAllValues:StringLike": {
                    "dynamodb:LeadingKeys": [
                        "global-settings.gpu_settings.*",
                        "global-settings.package_config.*"
                    ]
                }
            }
        },
        {
            "Sid": "RESS3Access",
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::{RES-EnvironmentName}-cluster-{AWS-Region}-{AWS-
Account-ID}/idea/vdc/res-ready-install-script-packages/*"
        }
    ]
}
```

Create EC2 Image Builder component

Follow the directions to [Create a component using the Image Builder console](#) in the *Image Builder User Guide*.

Enter your component details:

1. For **Type**, choose **Build**.

2. For **Image operating system (OS)**, choose either Linux or Windows.

3. For **Component name**, enter a meaningful name such as `research-and-engineering-studio-vdi-<operating-system>`.

4. Enter your component's version number and optionally add a description.

5. For the **Definition document**, enter the following definition file. If you encounter any errors, the YAML file is space sensitive and is the most likely cause.

    Linux

    ```
    #  Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
    #
    #  Licensed under the Apache License, Version 2.0 (the "License"). You may not
     use this file except in compliance
    #  with the License. A copy of the License is located at
    #
    #      http://www.apache.org/licenses/LICENSE-2.0
    #
    #  or in the 'license' file accompanying this file. This file is distributed on
     an 'AS IS' BASIS, WITHOUT WARRANTIES
    #  OR CONDITIONS OF ANY KIND, express or implied. See the License for the
     specific language governing permissions
    #  and limitations under the License.
    name: research-and-engineering-studio-vdi-linux
    description: An RES EC2 Image Builder component to install required RES software
     dependencies for Linux VDI.
    schemaVersion: 1.0
    parameters:
      - AWSAccountID:
          type: string
          description: RES Environment AWS Account ID
      - RESEnvName:
          type: string
          description: RES Environment Name
      - RESEnvRegion:
          type: string
          description: RES Environment Region
      - RESEnvReleaseVersion:
          type: string
    ```

```
          description: RES Release Version

phases:
  - name: build
    steps:
        - name: PrepareRESBootstrap
          action: ExecuteBash
          onFailure: Abort
          maxAttempts: 3
          inputs:
              commands:
                  - 'mkdir -p /root/bootstrap/logs'
                  - 'mkdir -p /root/bootstrap/latest'
        - name: DownloadRESLinuxInstallPackage
          action: S3Download
          onFailure: Abort
          maxAttempts: 3
          inputs:
            - source: 's3://{{ RESEnvName }}-cluster-{{ RESEnvRegion }}-
{{ AWSAccountID }}/idea/vdc/res-ready-install-script-packages/linux/
res_linux_install_{{ RESEnvReleaseVersion }}.tar.gz'
              destination: '/root/bootstrap/
res_linux_install_{{ RESEnvReleaseVersion }}.tar.gz'
              expectedBucketOwner: '{{ AWSAccountID }}'
        - name: RunInstallScript
          action: ExecuteBash
          onFailure: Abort
          maxAttempts: 3
          inputs:
              commands:
                  - 'tar -xvf
 {{ build.DownloadRESLinuxInstallPackage.inputs[0].destination }} -C /root/
bootstrap/latest'
                  - '/bin/bash /root/bootstrap/latest/virtual-desktop-host-linux/
install.sh -r {{ RESEnvRegion }} -n {{ RESEnvName }} -g NONE'
        - name: FirstReboot
          action: Reboot
          onFailure: Abort
          maxAttempts: 3
          inputs:
              delaySeconds: 0
        - name: RunInstallPostRebootScript
          action: ExecuteBash
          onFailure: Abort
```

```
            maxAttempts: 3
            inputs:
                commands:
                    - '/bin/bash /root/bootstrap/latest/virtual-desktop-host-linux/
install_post_reboot.sh'
        - name: SecondReboot
            action: Reboot
            onFailure: Abort
            maxAttempts: 3
            inputs:
                delaySeconds: 0
```

Windows

```
#  Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
#
#  Licensed under the Apache License, Version 2.0 (the "License"). You may not
 use this file except in compliance
#  with the License. A copy of the License is located at
#
#      http://www.apache.org/licenses/LICENSE-2.0
#
#  or in the 'license' file accompanying this file. This file is distributed on
 an 'AS IS' BASIS, WITHOUT WARRANTIES
#  OR CONDITIONS OF ANY KIND, express or implied. See the License for the
 specific language governing permissions
#  and limitations under the License.
name: research-and-engineering-studio-vdi-windows
description: An RES EC2 Image Builder component to install required RES software
 dependencies for Windows VDI.
schemaVersion: 1.0
parameters:
  - AWSAccountID:
      type: string
      description: RES Environment AWS Account ID
  - RESEnvName:
      type: string
      description: RES Environment Name
  - RESEnvRegion:
      type: string
      description: RES Environment Region
  - RESEnvReleaseVersion:
      type: string
```

```
            description: RES Release Version

phases:
  - name: build
    steps:
        - name: CreateRESBootstrapFolder
          action: CreateFolder
          onFailure: Abort
          maxAttempts: 3
          inputs:
              - path: 'C:\Users\Administrator\RES\Bootstrap'
                overwrite: true
        - name: DownloadRESWindowsInstallPackage
          action: S3Download
          onFailure: Abort
          maxAttempts: 3
          inputs:
              - source: 's3://{{ RESEnvName }}-cluster-{{ RESEnvRegion }}-
{{ AWSAccountID }}/idea/vdc/res-ready-install-script-packages/windows/
res_windows_install_{{ RESEnvReleaseVersion }}.tar.gz'
                destination:
 '{{ build.CreateRESBootstrapFolder.inputs[0].path }}\res_windows_install_{{ RESEnvRelea
                expectedBucketOwner: '{{ AWSAccountID }}'
        - name: RunInstallScript
          action: ExecutePowerShell
          onFailure: Abort
          maxAttempts: 3
          inputs:
              commands:
                  - 'cd {{ build.CreateRESBootstrapFolder.inputs[0].path }}'
                  - 'Tar -xf
 res_windows_install_{{ RESEnvReleaseVersion }}.tar.gz'
                  - 'Import-Module .\virtual-desktop-host-windows\Install.ps1'
                  - 'Install-WindowsEC2Instance'
        - name: Reboot
          action: Reboot
          onFailure: Abort
          maxAttempts: 3
          inputs:
              delaySeconds: 0
```

6. Create any optional tags and choose **Create component**.

Prepare your EC2 Image Builder recipe

> ⓘ **Note**
>
> CentOS 7 is currently scheduled to reach end-of-life on 6/30/2024. Research and
> Engineering Studio version 2024.06 will be the last version to support CentOS 7.

An EC2 Image Builder recipe defines the base image to use as your starting point to create a new image, along with the set of components that you add to customize your image and verify that everything works as expected. You must either create or modify a recipe to construct the target AMI with the necessary RES software dependencies. For more information on recipes, see [Manage recipes](#).

RES supports the following image operating systems:

- Amazon Linux 2 (x86 and ARM64)
- CentOS 7 (x86 and ARM64)
- RHEL 7 (x86), 8 (x86), and 9 (x86)
- Ubuntu 22.04.3 (x86)
- Windows 2019, 2022 (x86)

Create a new recipe

1. Open the EC2 Image Builder console at https://console.aws.amazon.com/imagebuilder.

2. Under **Saved resources**, choose **Image recipes**.

3. Choose **Create image recipe**.

4. Enter a unique name and a version number.

5. Choose a base image supported by RES.

6. Under **Instance configuration**, install an SSM agent if one does not come pre-installed. Enter the information in **User data** and any other needed user data.

   > ⓘ **Note**
   >
   > For information on how to install an SSM agent, see:

- Manually installing SSM Agent on EC2 instances for Linux

- Manually installing and uninstalling SSM Agent on EC2 instances for Windows Server

7. For Linux based recipes, add the Amazon-managed `aws-cli-version-2-linux` build component to the recipe. RES installation scripts use the AWS CLI to provide VDI access to configuration values for the DynamoDB cluster-settings. Windows does not require this component.

8. Add the EC2 Image Builder component created for your Linux or Windows environment and enter any required parameter values. The following parameters are required inputs: AWSAccountID, RESEnvName, RESEnvRegion, and RESEnvReleaseVersion.

> ⚠️ **Important**
>
> For Linux environments, you must add these components in order with the `aws-cli-version-2-linux` build component added first.

9. (Recommended) Add the Amazon-managed `simple-boot-test-<linux-or-windows>` test component to verify that the AMI can be launched. This is a minimum recommendation. You may select other test components that meet your requirements.

10. Complete any optional sections if needed, add any other desired components, and choose **Create recipe**.

Modify a recipe

If you have an existing EC2 Image Builder recipe, you can use it by adding the following components:

1. For Linux based recipes, add the Amazon-managed `aws-cli-version-2-linux` build component to the recipe. RES installation scripts use the AWS CLI to provide VDI access to configuration values for the DynamoDB cluster-settings. Windows does not require this component.

2. Add the EC2 Image Builder component created for your Linux or Windows environment and enter any required parameter values. The following parameters are required inputs: AWSAccountID, RESEnvName, RESEnvRegion, and RESEnvReleaseVersion.

> ⚠ **Important**
>
> For Linux environments, you must add these components in order with the `aws-cli-version-2-linux` build component added first.

3. Complete any optional sections if needed, add any other desired components, and choose **Create recipe**.

Configure EC2 Image Builder infrastructure

You can use infrastructure configurations to specify the Amazon EC2 infrastructure that Image Builder uses to build and test your Image Builder image. For use with RES, you can choose to create a new infrastructure configuration, or use an existing one.

- To create a new infrastructure configuration, see Create an infrastructure configuration.

- To use an existing infrastructure configuration, Update an infrastructure configuration.

To configure your Image Builder infrastructure:

1. For **IAM role**, enter the role you previously configured in the section called "Prepare IAM role to access RES environment".

2. For **Instance type**, choose a type with at least 4 GB of memory and supports your chosen base AMI architecture. See Amazon EC2 Instance types.

3. For **VPC, subnet, and security groups**, you must permit internet access to download software packages. Access must also be allowed to the `cluster-settings` DynamoDB table and Amazon S3 cluster bucket of the RES environment.

Configure Image Builder image pipeline

The Image Builder image pipeline assembles the base image, components for building and testing, infrastructure configuration, and distribution settings. To configure an image pipeline for RES-ready AMIs, you can choose to create a new pipeline, or use an existing one. For more information, see Create and update AMI image pipelines in the *Image Builder User Guide*.

Create a new Image Builder pipeline

1. Open the Image Builder console at https://console.aws.amazon.com/imagebuilder.

2. From the navigation, choose **Image pipelines**.

3. Choose **Create image pipeline**.

4. Specify your pipeline details by entering a unique name, optional description, schedule, and frequency.

5. For **Choose recipe**, choose **Use existing recipe** and select the recipe created in the section called "Prepare your EC2 Image Builder recipe". Verify that your recipe details are correct.

6. For **Define image creation process**, choose either the default or custom workflow depending on the use case. In most cases, the default workflows are sufficient. For more information, see Configure image workflows for your EC2 Image Builder pipeline.

7. For **Define infrastructure configuration**, choose **Choose existing infrastructure configuration** and select the infrastructure configuration created in the section called "Configure EC2 Image Builder infrastructure". Verify that your infrastructure details are correct.

8. For **Define distribution settings**, choose **Create distribution settings using service defaults**. The output image must reside in the same AWS Region as your RES environment. Using service defaults, the image will be created in the Region where Image Builder is used.

9. Review the pipeline details and choose **Create pipeline**.

Modify an existing Image Builder pipeline

1. To use an existing pipeline, modify the details to use the recipe created in the section called "Prepare your EC2 Image Builder recipe".

2. Choose **Save changes**.

Run Image Builder image pipeline

To produce the output image configured, you must initiate the image pipeline. The building process can potentially take up to an hour depending on the number of components in the image recipe.

To run the image pipeline:

1. From **Image pipelines**, select the pipeline created in the section called "Configure Image Builder image pipeline".

2. From **Actions**, choose **Run pipeline**.

Register a new software stack in RES

1. Follow the directions in the section called "Software Stacks (AMIs)" to register a software stack.

2. For **AMI ID**, enter the AMI ID of the output image built in the section called "Run Image Builder image pipeline".

Administrator guide

This administrator guide provides additional instructions for a technical audience on how to further customize and integrate with the Research and Engineering Studio on AWS product.

Topics

- [Session management](#)
- [Environment management](#)
- [Secrets management](#)
- [Cost monitoring and control](#)
- [Permissions](#)

Session management

Session management provides a flexible and interactive environment for developing and testing sessions. As an administrative user, you can permit users to create and manage interactive sessions within their project environments.

Topics

- [Dashboard](#)
- [Sessions](#)
- [Software Stacks (AMIs)](#)
- [Permission Profiles](#)
- [Debugging](#)
- [Desktop settings](#)

Dashboard

Research and Engineering Studio   🔔   👤 demoadmin1 ▾

res-stage (us-west-2) ‹   RES › Virtual Desktop › Dashboard   ⓘ

Virtual Desktop Dashboard   ⑦ 🔄   **View Sessions**   ⑧

▾ **Home**
    Virtual Desktops
    Shared Desktops
    File Browser
    SSH Access

ADMIN ZONE

▾ **eVDI**
    Dashboard
    Sessions
    Software Stacks (AMIs)
    Permission Profiles
    Debug
    Settings

▸ **Environment Management**

Instance Types ①
Summary of all virtual desktop sessions by instance types.

3 sessions

m6a.large

▪ m6a.large

Session State ②
Summary of all virtual desktop sessions by state.

STOPPING

▪ STOPPING

Base OS ③
Summary of all virtual desktop sessions by Base OS.

Windows   Amazon Linu...

▪ Amazon Linux 2   ▪ Windows

Project ④
Summary of all virtual desktop sessions by Project Code

project1

▪ project1

Availability Zones ⑤
Summary of all virtual desktop sessions by Availability Zone.

us-west-2a

▪ us-west-2a

Software Stacks ⑥
Summary of all virtual desktop sessions by Software Stack.

Software Stacks

Amazon Linux 2 - x86_64

Windows - x86_64

0   0.5   1   1.5   2
No. of Sessions

▪ Sessions

The Session Management Dashboard provides administrators with a quick view into:

1. Instance types

2. Session states

3. Base OS

4. Projects

5. Availability zones

6. Software stacks

Additionally, administrators can:

7. Refresh the dashboard to update information.

8. Choose **View Sessions** to navigate to Sessions.

Sessions

Sessions displays all virtual desktops created within Research and Engineering Studio. From the Sessions page, you can filter and view session information or create a new session.

1. Use the menu to filter results by sessions created or updated within a specified time frame.

2. Select a session and use the Actions menu to:

    a. Resume Session(s)

    b. Stop/Hibernate Session(s)

    c. Force Stop/Hibernate Session(s)

    d. Terminate Session(s)

    e. Force Terminate Session(s)

    f. Session(s) Health

    g. Create Software Stack

3. Choose **Create Session** to create a new session.

4. Search for a session by name and filter by state and operating system.

5. Choose the **Session Name** to view more details.

Create a session

1. Choose **Create Session**. The Launch New Virtual Desktop modal opens.

2. Enter details for the new session.

3. (Optional.) Turn on **Show Advanced Options** to provide additional details such as subnet ID and DCV session type.

4. Choose **Submit**.

Launch New Virtual Desktop   ✕

Session Name

Enter a name for the virtual desktop

[   ]

Session Name is required. Use any characters and form a name of length between 3 and 24 characters, inclusive.

User

Select the user to create the session for

[🔍   ]

Project

Select the project under which the session will get created

[▼]

Operating System

Select the operating system for the virtual desktop

[Amazon Linux 2   ▼]

Software Stack

Select the software stack for your virtual desktop

[▼]

Enable Instance Hibernation

Hibernation saves the contents from the instance memory (RAM) to your Amazon Elastic Block Store (Amazon EBS) root volume. You can not change instance type if you enable this option.

(●)

Virtual Desktop Size

Select a virtual desktop instance type

[🔍   ]

Storage Size (GB)

Enter the storage size for your virtual desktop in GBs

[10   ]

Session details

From the **Sessions** list, choose the **Session Name** to view session details.

Software Stacks (AMIs)

> **Note**
>
> To run the provided CentSO7 software stack in AWS GovCloud (US), you will need to subscribe to the AMI within AWS Marketplace using your [linked standard account](#).

From the Software Stacks page, you can configure Amazon Machine Images (AMIs) and manage existing AMIs.

1. To search for an existing software stack, use the operating system drop-down to filter by OS.

2. Choose the name of a software stack to view details about the stack.

3. Once you select a software stack, use the **Actions** menu to edit the stack and assign the stack to a project.

4. The **Register Software Stack** button lets you create a new stack:

   1. Choose **Register Software Stack**.

   2. Enter details for the new software stack.

   3. Choose **Submit**.

Register new Software Stack ✕

Name
Enter a name for the software stack

[   ]

Use any characters and form a name of length between 3 and 24 characters, inclusive.

Description
Enter a user friendly description for the software stack

[   ]

AMI Id
Enter the AMI Id

[   ]

AMI Id must start with ami-xxx

Operating System
Select the operating system for the software stack

[Amazon Linux 2   ▼]

GPU Manufacturer
Select the GPU Manufacturer for the software stack

[N/A   ▼]

Min. Storage Size (GB)
Enter the min. storage size for your virtual desktop in GBs

[10   ]

Min. RAM (GB)
Enter the min. ram for your virtual desktop in GBs

[10   ]

Projects
Select applicable projects for the software stack

[▼]

Assign software stack to a project

When you are creating a new software stack, you can assign the stack to projects. If you need to add the stack to a project after the initial creation, do the following:

> ⓘ **Note**
>
> You can only assign software stacks to projects of which you are a member.

1. Select the software stack you need to add to a project from the Software Stacks page.

2. Choose **Actions**.

3. Choose **Edit**.

4. Use the **Projects** drop-down to select the project.

5. Choose **Submit**.

You can also edit the software stack from the stack details page.

View software stack details

From the **Software Stacks** list, choose the **Software Stack Name** to view details. From the details page, you can also choose **Edit** to edit the software stack.

Permission Profiles

Use **Permission Profiles** to create and manage reusable profiles for permissions.

Research and Engineering Studio   User Guide

1. Search for a permission profile.

2. Choose the **Profile ID** to view details.

3. When a profile is selected, use the **Actions** menu to edit the profile.

4. Choose **Create Permission Profile** to create a new profile.

Create a permission profile

1. Choose **Create Permission Profile**.

2. Enter details for the new profile and use the permission toggles to select permissions for the profile.

3. Choose **Submit**.

Register new Permission Profile ✕

Profile ID
Enter a Unique Profile ID for the Permission Profile

Title
Enter a user friendly Title for the Permission Profile

Description
Enter a user friendly description for the Permission Profile

Built In
All features

Display
Receive visual data from the NICE DCV server

Pointer
View NICE DCV server mouse position events and pointer shapes

Mouse
Input from the client mouse to the NICE DCV server

Keyboard
Input from the client keyboard to the NICE DCV server

Audio In
Send audio from the client to the NICE DCV server

Audio Out
Receive audio from the NICE DCV server to the client

Clipboard Copy
Copy data from the NICE DCV server to the client clipboard

Clipboard Paste
Copy data to the NICE DCV server from the client clipboard

File Upload
Upload files to the session storage

File Download
Download files from the session storage

USB
Use USB devices from the client

Printer
Create PDFs or XPS files from the NICE DCV server to the client

Smartcard
Read the smart card from the client

Stylus
Input from specialized USB devices, such as 3D pointing devices or graphic tablets

Keyboard SAS
Use the secure attention sequence (CTRL+Alt+Del). Note: Requires Keyboard permissions as well

Web Camera
Use the Web Camera connected to a client device in a session

Touch
Use native touch events from the client device

Screenshot
Save a screenshot of the remote desktop

Gamepad
Use gamepads connected to a client computer in a session

Unsupervised Access
Allow a user to connect to session without supervision

Cancel   **Submit**

Edit a permission profile

1. Select the permission profile you need to edit from the **Permission Profiles** page.

2. Choose **Actions**.

3. Choose **Edit Permission Profile**.

4. Edit the profile.

5. Choose **Submit**.

View permission profile details

From the **Permission Profiles** list, choose the **Profile ID** to view details. From the details page, you can also choose **Edit** to edit the permission profile.

Debugging

The debugging panel displays message traffic associated with the virtual desktops. You can use this panel to observe activity between hosts. The VD Host tab displays instance specific activity, and the VD Sessions tab displays in-progress session activity.

Desktop settings

You can use the Desktop Settings page for configuring resources associated with virtual desktops. The Server tab provides access to settings such as:

- DCV session idle timeout

- Idle timeout warning

- CPU utilization threshold

- Allowed sessions per user

Environment management

From the Environment management section of RES, administrative users can create and manage isolated environments for their research and engineering projects. These environments can include compute resources, storage, and other necessary components, all within a secure environment. Users can configure and customize these environments to meet the specific requirements of their projects, making it easier to experiment, test, and iterate on their solutions without impacting other projects or environments.

Topics

- [Projects](#)

- [Users](#)

- [Groups](#)

- [File Systems](#)

- [Environment status](#)

- [Snapshot management](#)

- [Environment settings](#)

Projects

Projects form a boundary for virtual desktops, teams, and budgets. When you create a project, you define its settings, such as the name, description, and environment configuration. Projects typically include one or more environments, which can be customized to meet the specific requirements of your project, such as the type and size of the compute resources, the software stack, and the networking configuration.

Topics

- [View projects](#)
- [Create a project](#)
- [Edit a project](#)
- [Add or remove tags from a project](#)
- [View file systems associated with a project](#)
- [Add a launch template](#)

View projects
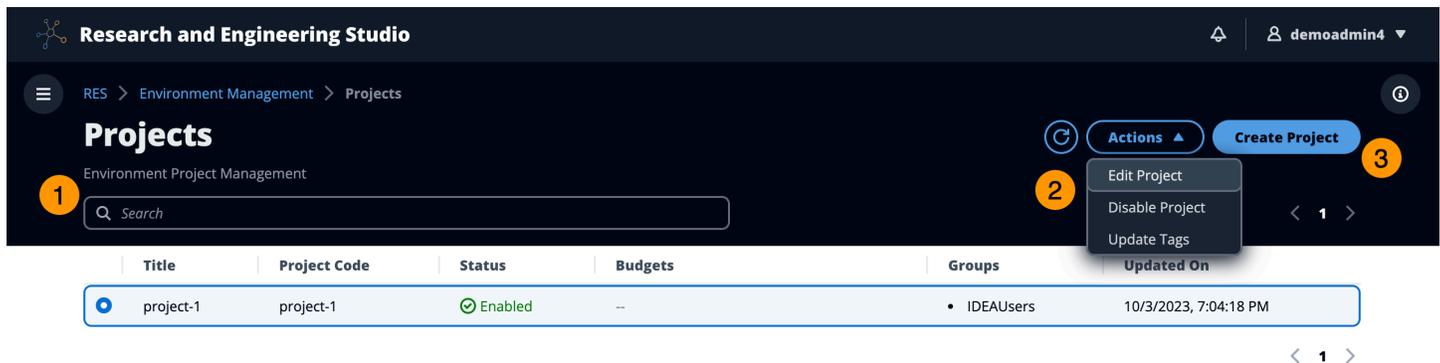
The Projects dashboard provides a list of projects available to you. From the Projects dashboard, you can:

1. You can use the search field to find projects.

2. When a project is selected, you can use the **Actions** menu to:

    a. Edit a project

    b. Disable or enable a project

    c. Update project tags

3. You can choose **Create Project** to create a new project.

Create a project

1. Choose **Create Project**.

2. Enter project details.

   The Project ID is a resource tag that can be used to track cost allocation in AWS Cost Explorer Service. For more information, see [Activating user-defined cost allocation tags](#).

   > ⚠️ **Important**
   >
   > The project ID cannot be changed after creation.

   For information on **Advanced Options**, see [Add a launch template](#).

3. (Optional) Turn on budgets for the project. For more information on budgets, see [Cost monitoring and control](#).

4. Assign users and/or groups the appropriate role ("Project Member" or "Project Owner"). See [Permissions](#) for the actions each role can take.

5. Choose **Submit**.

Create new Project

Project Definition

Title
Enter a user friendly project title

Project ID
Enter a project-id

Project ID can only use lowercase alphabets, numbers, hyphens (-), underscores (_), or periods (.). Must be between 3 and 40 characters long.

Description
Enter the project description

Enter Description ...

Do you want to enable budgets for this project?

Resource Configurations

Add file systems
Select applicable file systems for the Project

home [efs] ✕

▶ Advanced Options

Team Configurations

Groups
Select applicable ldap groups for the Project

group_1 ▼

Role
Choose a role for the group

Project Member ▼

Remove group

Add group

Users
Select applicable users for the Project

user1 ▼

Role
Choose a role for the user

Project Member ▼

Remove user

Add user

Cancel   Submit

Edit a project

1. Select a project in the project list.

2. From the **Actions** menu, choose **Edit Project**.

3. Enter your updates. If you intend to enable budgets, see Cost monitoring and control for more information. For information on **Advanced Options**, see Add a launch template.

4. Choose **Submit**.

Edit Project

Project Definition

Title
Enter a user friendly project title

> Project1

Project ID
Enter a project-id

> 100

Project ID can only use lowercase alphabets, numbers, hyphens (-), underscores (_), or periods (.). Must be between 3 and 40 characters long.

Description
Enter the project description

> Enter Description ...

Do you want to enable budgets for this project?

Resource Configurations

▼ **Advanced Options**

Add Policies
Select applicable policies for the Project

> ▼   ↻

Add Security Groups
Select applicable security groups for the Project

> ▼   ↻

▶ **Linux**

▶ **Windows**

Team Configurations

Groups
Select applicable ldap groups for the Project

> group_1 ▼

Role
Choose a role for the group

> Project Member ▼

[Remove group]

[Add group]

Users
Select applicable users for the Project

> user1 ▼

Role
Choose a role for the user

> Project Member ▼

[Remove user]

[Add user]

[Cancel] [Submit]

Add or remove tags from a project

Project tags will assign tags to all instances created under that project.

1. Select a project in the project list.

2. From the **Actions** menu, choose **Update Tags**.

3. Choose **Add Tags** and enter a value for **Key**.

4. To remove tags, choose **Remove** next to the tag you want to remove.

View file systems associated with a project

When a project is selected, you can expand the **File Systems** pane at the bottom of the screen to view file systems associated with the project.

Add a launch template

When creating or editing a project, you can add launch templates using the **Advanced Options** within the project configuration. Launch templates provide additional configurations, such as security groups, IAM policies, and launch scripts to all VDI instances within the project.

Add policies

You can add an IAM policy to control VDI access for all instances deployed under your project. To onboard a policy, tag the policy with the following key-value pair:

```
res:Resource/vdi-host-policy
```

For more information on IAM roles, see Policies and permissions in IAM.

Add security groups

You can add a security group to control the egress and ingress data for all VDI instances under your project. To onboard a security group, tag the security group with the following key-value pair:

```
res:Resource/vdi-security-group
```

For more information on security groups, see [Control traffic to your AWS resources using security groups](#) in the *Amazon VPC User Guide*.

Add launch scripts

You can add launch scripts that will initiate on all VDI sessions within your project. RES supports script initiation for Linux and Windows. For script initiation, you can choose either:

Run Script When VDI Starts

> This option initiates the script at the beginning of a VDI instance before any RES configurations or installations run.

Run Script when VDI is Configured

> This option initiates the script after RES configurations complete.

Scripts support the following options:

Script configuration	Example
S3 URI	s3://bucketname/script.sh
HTTPS URL	https://sample.samplecontent.com/sample
Local file	file:///user/scripts/example.sh

For **Arguments**, provide any arguments separated by a comma.

▼ **Linux**

Run Script When VDI Starts
Scripts that execute at the start of a VDI

Script Info

Arguments - *optional* Info

| s3://sample-res-scripts/sample.sh | 1,2 | **Remove Scripts** |

| https://sample.samplecontent.com/sample | | **Remove Scripts** |

| file:///root/bootstrap/latest/launch/script | 1,2 | **Remove Scripts** |

Add Scripts

Run Script when VDI is Configured
Scripts that execute after RES configurations are completed

Script Info

Arguments - *optional* Info

| s3://sample-res-scripts/sample.sh | 1,2 | **Remove Scripts** |

Add Scripts

▼ **Windows**

Run Script When VDI Starts
Scripts that execute at the start of a VDI

Script Info

Arguments - *optional* Info

| s3://sample-res-scripts/sample.sh | 1,2 | **Remove Scripts** |

Add Scripts

Run Script when VDI is Configured
Scripts that execute after RES configurations are completed

Script Info

Arguments - *optional* Info

| s3://sample-res-scripts/sample.sh | 1,2 | **Remove Scripts** |

Add Scripts

Example of a project configuration

Users

All users synced from your active directory will appear on the Users page. Users are synced by the cluster-admin user during configuration of the product. For more information on initial user configuration, see the *Configuration guide*.

> **ⓘ Note**
>
> Administrators can only create sessions for active users. By default, all users will be in an inactive state until they sign in to the product environment. If a user is inactive, ask them to sign in prior to creating a session for them.

From the **Users** page, you can:

1. Search for users.

2. When a username is selected, use the **Actions** menu to:

   a. Set as Admin user

   b. Disable user

Groups

All Groups synced from the active directory appear on the Groups page. For more information on group configuration and management, see the *Configuration guide*.

From the **Groups** page, you can:

1. Search for user groups.

2. When a user group is selected, use the **Actions** menu to disable or enable a group.

3. When a user group is selected, you can expand the **Users** pane at the bottom of the screen to view users in the group.

File Systems

From the File Systems page, you can:

1. Search for file systems.

2. When a file system is selected, use the **Actions** menu to:

    a. Add the file system to a project

    b. Remove the file system from a project

3. Onboard a new file system.

4. Create a file system.

5. When a file system is selected, you can expand the pane at the bottom of the screen to view file system details.

Create a file system

1. Choose **Create File System**.

2. Enter the details for the new file system.

3. Provide Subnet IDs from the VPC. You can find the IDs in the **Environment Management** > **Settings** > **Network** tab.

4. Choose **Submit**.

Create new File System ✕

Title

Enter a user friendly file system title

[   ]

Eg. EFS 01

Name

Enter a file system name

[   ]

File System name can only use lowercase alphabets, numbers and underscore (_). Must be between 3 and 18 characters long.

File System Provider

Select applicable file system type

[EFS   ▼]

Projects

Select applicable project

[▼]

⟳

Subnet ID 1

Enter subnet id to create mount target

[   ]

Subnet ID 2

Enter second subnet to create mount target

[   ]

Subnet ID 1 and Subnet ID 2 should be in two different AZs

Mount Directory

Enter directory to mount the file system

Onboard a file system

1. Choose **Onboard File System**.

2. Select a file system from the drop down. The modal will expand with additional detail entries.
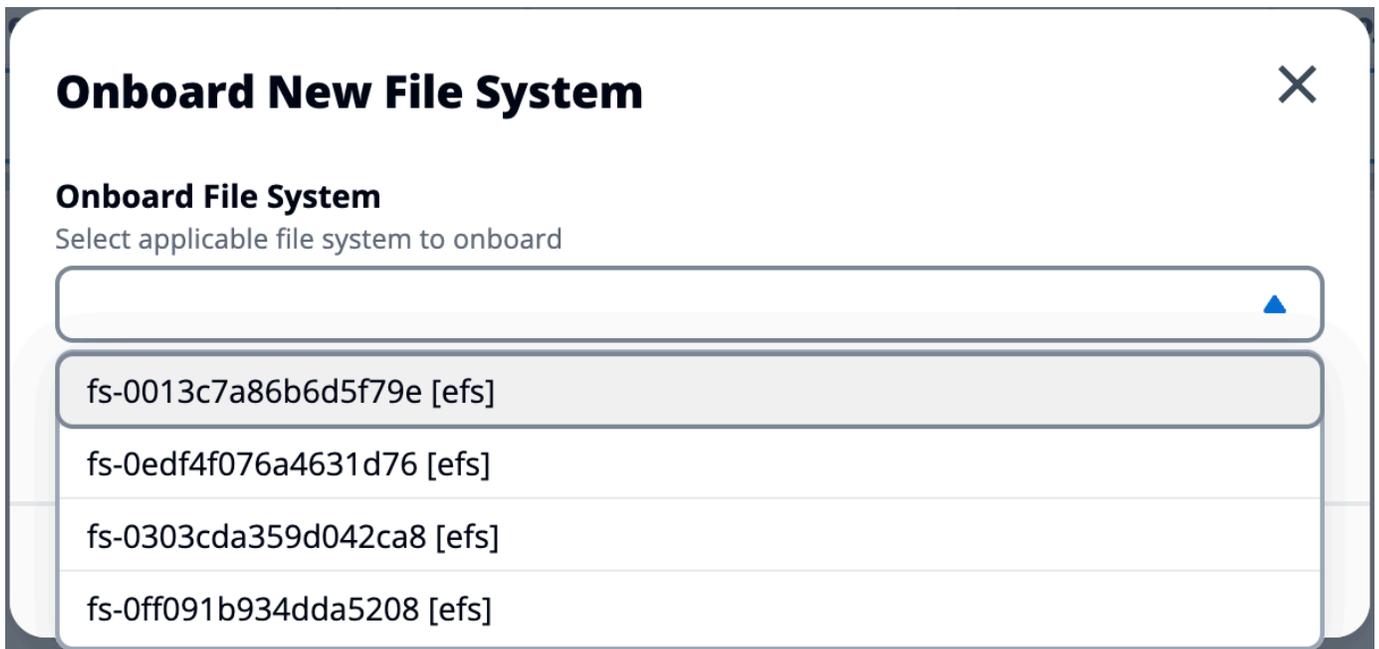
Onboard New File System   ✕

Onboard File System
Select applicable file system to onboard

[▲]

fs-0013c7a86b6d5f79e [efs]

fs-0edf4f076a4631d76 [efs]

fs-0303cda359d042ca8 [efs]

fs-0ff091b934dda5208 [efs]

3. Enter file system details.

4. Choose **Submit**.

Onboard New File System ✕

Onboard File System

Select applicable file system to onboard

    fs-0edf4f076a4631d76 [efs]   ▼

↻

Title

Enter a user friendly file system title

File System Name

Enter a file system name

File System name cannot contain white spaces or special characters. Only use lowercase alphabets, numbers and underscore (_). Must be between 3 and 18 characters long.

Mount Directory

Enter directory to mount the file system

Mount directory cannot contain white spaces or special characters. Only use lowercase alphabets, numbers, and hyphens (-). Must be between 3 and 18 characters long. Eg. /efs-01

Cancel   Submit

Environment status

The Environment Status page displays the deployed software and hosts within the product. It includes information such as software version, module names, and other system information.

Snapshot management

Snapshot management simplifies the process of saving and migrating data between environments, ensuring consistency and accuracy. With snapshots, you can save your environment state and migrate data into a new environment with the same state.

From the Snapshot management page, you can:

1. View all created snapshots and their status.

2. Create a snapshot. Before you can create a snapshot, you will need to create a bucket with the appropriate permissions.

3. View all applied snapshots and their status.

4. Apply a snapshot.

Create a snapshot

Before you can create a snapshot, you must provide an Amazon S3 bucket with the necessary permissions. For information on creating a bucket, see Creating a bucket. We recommend enabling bucket versioning and server access logging. These settings can be enabled from the bucket's **Properties** tab after provisioning.

> **ⓘ Note**
>
> This Amazon S3 bucket's lifecycle will not be managed within the product. You will need to manage the bucket lifecycle from the console.

To add permissions to the bucket:

1. Choose the bucket you created from the **Buckets** list.

2. Choose the **Permissions** tab.

3. Under **Bucket policy**, choose **Edit**.

4. Add the following statement to the bucket policy. Replace these values with your own:

   - AWS_ACCOUNT_ID

   - RES_ENVIRONMENT_NAME

   - AWS_REGION

   - S3_BUCKET_NAME

> **⚠ Important**
>
> There are limited version strings supported by AWS. For more information, see https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_elements_version.html

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Export-Snapshot-Policy",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::{AWS_ACCOUNT_ID}:role/{RES_ENVIRONMENT_NAME}-
cluster-manager-role-{AWS_REGION}}"
            },
            "Action": [
                "s3:GetObject",
```

```
                    "s3:ListBucket",
                    "s3:AbortMultipartUpload",
                    "s3:PutObject",
                    "s3:PutObjectAcl"
                ],
                "Resource": [
                    "arn:aws:s3:::{S3_BUCKET_NAME}",
                    "arn:aws:s3:::{S3_BUCKET_NAME}/*"
                ]
            },
            {
                "Sid": "AllowSSLRequestsOnly",
                "Action": "s3:*",
                "Effect": "Deny",
                "Resource": [
                    "arn:aws:s3:::{S3_BUCKET_NAME}",
                    "arn:aws:s3:::{S3_BUCKET_NAME}/*"
                ],
                "Condition": {
                    "Bool": {
                        "aws:SecureTransport": "false"
                    }
                },
                "Principal": "*"
            }
        ]
    }
```

To create the snapshot:

1. Choose **Create Snapshot**.

2. Enter the name of the Amazon S3 bucket you created.

3. Enter the path where you would like the snapshot stored within the bucket. For example,
   october2023/23.

4. Choose **Submit**.

Create New Snapshot   ✕

S3 Bucket Name
Enter the name of an existing S3 bucket where the snapshot should be stored.

[   ]

S3 bucket name can only contain lowercase alphabets, numbers, dots (.), and hyphens (-).

Snapshot Path
Enter a path at which the snapshot should be stored in the provided S3 bucket.

[   ]

Snapshot path can only contain forward slashes, dots (.), exclamations (!), asterisks (*), single
quotes ('), parentheses (), and hyphens (-).

Cancel   **Submit**

5. After five to ten minutes, choose **Refresh** on the Snapshots page to check the status. A
    snapshot will not be valid until the status changes from IN_PROGRESS to COMPLETED.

Apply a snapshot

Once you have created a snapshot of an environment, you can apply that snapshot to a new
environment to migrate data. You will need to add a new policy to the bucket allowing the
environment to read the snapshot.

Applying a snapshot copies data such as user permissions, projects, software stacks, permission
profiles, and file systems with their associations to a new environment. User sessions will not be
replicated. When the snapshot is applied, it checks the basic information of each resource record
to determine if it already exists. For duplicate records, the snapshot skips resource creation in the
new environment. For records that are similar, such as share a name or key, but other basic resource
information varies, it will create a new record with a modified name and key using the following
convention: `RecordName_SnapshotRESVersion_ApplySnapshotID`. The `ApplySnapshotID`
looks like a timestamp and identifies each attempt to apply a snapshot.

During the snapshot application, the snapshot checks for the availability of resources. Resource not available to the new environment will not be created. For resources with a dependent resource, the snapshot checks for the availability of the dependent resource. If the dependent resource is not available, it will create the main resource without the dependent resource.

If the new environment is not as expected or fails, you can check the CloudWatch logs found in the log group `/res-<env-name>/cluster-manager` for details. Each log will have the [apply snapshot] tag. Once you have applied a snapshot, you can check its status from the the section called "Snapshot management" page.

To add permissions to the bucket:

1. Choose the bucket you created from the **Buckets** list.

2. Choose the **Permissions** tab.

3. Under **Bucket policy**, choose **Edit**.

4. Add the following statement to the bucket policy. Replace these values with your own:

   - AWS_ACCOUNT_ID

   - RES_ENVIRONMENT_NAME

   - AWS_REGION

   - S3_BUCKET_NAME

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Export-Snapshot-Policy",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::{AWS_ACCOUNT_ID}:role/{RES_ENVIRONMENT_NAME}-
cluster-manager-role-{AWS_REGION}"
            },
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::{S3_BUCKET_NAME}",
                "arn:aws:s3:::{S3_BUCKET_NAME}/*"
```

```
            ]
        },
        {
            "Sid": "AllowSSLRequestsOnly",
            "Action": "s3:*",
            "Effect": "Deny",
            "Resource": [
                "arn:aws:s3:::{S3_BUCKET_NAME}",
                "arn:aws:s3:::{S3_BUCKET_NAME}/*"
            ],
            "Condition": {
                "Bool": {
                    "aws:SecureTransport": "false"
                }
            },
            "Principal": "*"
        }
    ]
}
```

To apply snapshot:

1. Choose **Apply snapshot**.

2. Enter the name of the Amazon S3 bucket containing the snapshot.

3. Enter the file path to the snapshot within the bucket.

4. Choose **Submit**.

Apply a Snapshot   ✕

S3 Bucket Name

Enter the name of the S3 bucket where the snapshot to be applied is stored.

[   ]

S3 bucket name can only contain lowercase alphabets, numbers, dots (.), and hyphens (-).

Snapshot Path

Enter the path at which the snapshot to be applied is stored in the provided S3 bucket.

[   ]

Snapshot path can only contain forward slashes, dots (.), exclamations (!), asterisks (*), single quotes ('), parentheses (), and hyphens (-).

Cancel   **Submit**

5. After five to ten minutes, choose **Refresh** on the Snapshot management page to check the status.

Environment settings

Environment settings displays product configuration details, such as:

- General

  Displays information such as the Administrator Username and email for the user who provisioned the product. You can edit the web portal title and copyright text.

- Identity Provider

  Displays information such as Single Sign-On status.

- Network

  Displays VPC ID, Prefix list IDs for access.

- Directory Service

Displays active directory settings and service account secrets manager ARN for username and password.

Secrets management

Research and Engineering Studio maintains the following secrets using AWS Secrets Manager. RES creates secrets automatically during environment creation. Secrets entered by the administrator during environment creation are entered as parameters.

Secret name	Description	RES generated	Admin entered
<envname>-sso-client-secret	Single Sign-On OAuth2 Client Secret for environment	✓	
<envname>-vdc-client-secret	vdc ClientSecret	✓	
<envname>-vdc-client-id	vdc ClientId	✓	
<envname>-vdc-gateway-certificate-private-key	Self-Signed certificate private key for domain	✓	
<envname>-vdc-gateway-certificate-certificate	Self-Signed certificate for domain	✓	
<envname>-cluster-manager-client-secret	cluster-manager ClientSecret	✓	
<envname>-cluster-manager-client-id	cluster-manager ClientId	✓	
<envname>-external-private-key	Self-Signed certificate private key for domain	✓	
<envname>-external-certificate	Self-Signed certificate for domain	✓	
<envname>-internal-private-key	Self-Signed certificate private key for domain	✓	

Secret name	Description	RES generated	Admin entered
<envname>-internal-certificate	Self-Signed certificate for domain	✓	
<envname>-directoryservice-ServiceAccountUsername			✓
<envname>-directoryservice-ServiceAccountPassword			✓

The following secret ARN values are contained in the <envname>-cluster-settings table in DynamoDB:

Key	Source
identity-provider.cognito.sso_client_secret	
vdc.dcv_connection_gateway.certificate.certificate_secret_arn	stack
vdc.dcv_connection_gateway.certificate.private_key_secret_arn	stack
cluster.load_balancers.internal_alb.certificates.private_key_secret_arn	stack
directoryservice.root_username_secret_arn	
vdc.client_secret	stack
cluster.load_balancers.external_alb.certificates.certificate_secret_arn	stack
cluster.load_balancers.internal_alb.certificates.certificate_secret_arn	stack
directoryservice.root_password_secret_arn	

Key	Source
cluster.secretsmanager.kms_key_id	
cluster.load_balancers.external_alb.certificates.private_key_se cret_arn	stack
cluster-manager.client_secret	

Cost monitoring and control

> **ⓘ Note**
>
> Associating Research and Engineering Studio projects to AWS Budgets is not supported in AWS GovCloud (US).

We recommend creating a budget through AWS Cost Explorer to help manage costs. Prices are subject to change. For full details, see the pricing webpage for each of the the section called "AWS services in this product".

To assist with cost tracking, you can associate RES projects to budgets created within AWS Budgets. You will first need to activate the environment tags within the billing cost allocation tags.

1. Sign in to the AWS Management Console and open the AWS Billing console at https://console.aws.amazon.com/billing/.

2. Choose **Cost allocation tags**.

3. Search for and select the `res:Project` and `res:EnvironmentName` tags.

4. Choose **Activate**.

> ⓘ **Note**
>
> It may take up to a day for RES tags to appear following deployment.

To create a budget for RES resources:

1. From the Billing console, choose **Budgets**.

2. Choose **Create a budget**.

3. Under **Budget setup**, choose **Customize (advanced)**.

4. Under **Budget types**, choose **Cost budget - Recommended**.

5. Choose **Next**.

6. Under **Details**, enter a meaningful **Budget name** for your budget to distinguish it from other budgets in your account. For example, [EnvironmentName]-[ProjectName]-[BudgetName].

7. Under **Set budget amount**, enter the amount budgeted for your project.

8. Under **Budget scope**, choose **Filter specific AWS cost dimensions**.

9. Choose **Add filter**.

10. Under **Dimension**, choose **Tag**.

11. Under **Tag**, select **res:Project**.

> ⓘ **Note**
>
> It may take up to two days for tags and values to become available. You can create a budget once the project name becomes available.

12. Under **Values**, select the project name.

13. Choose **Apply filter** to attach the project filter to the budget.

14. Choose **Next**.

15. (Optional.) Add an alert threshold.

16. Choose **Next**.

17. (Optional.) If an alert was configured, use **Attach actions** to configure desired actions with the alert.

18. Choose **Next**.

19. Review the budget configuration and confirm the correct tag was set under **Additional budget parameters**.

20. Choose **Create budget**.

Now that the budget has been created, you can enable the budget for projects. To turn on budgets for a project, see the section called "Edit a project". Virtual desktops will be blocked from launching if the budget is exceeded. If the budget is exceeded while a desktop is launched, the desktop will continue to operate.

If you need to change your budget, return to the console to edit the budget amount. It may take up to fifteen minutes for the change to take effect within RES. Alternatively, you may edit a project to disable a budget.

Permissions

	Project Member	Project Owner	Global Administrator	Scope
Add users as project member/ project owner		X	X	Project owner: Projects they own

|	Project Member	Project Owner	Global Administrator	Scope
				Global Administrator: Any project
Add groups as project member/ project owner		X	X	Project owner: Projects they own Global Administrator: Any project
Remove users		X	X	Project owner: Projects they own Global Administrator: Any project
Remove groups		X	X	Project owner: Projects they own Global Administrator: Any project
Start/Stop VDI instances	X	X	X	Project member/ Project Owner: VDI instances they own when they are part of a project. Global Administrator: Any VDI instances.

Use the product

This section offers guidance to users on using virtual desktops to collaborate with other users.

Topics

- [Virtual desktops](#)
- [Shared desktops](#)
- [File browser](#)
- [SSH access](#)

Virtual desktops

The virtual desktop interface (VDI) module allows users create and manage Windows or Linux virtual desktops on AWS. Users can launch Amazon EC2 instances with their favorite tools and application pre-installed and configured.
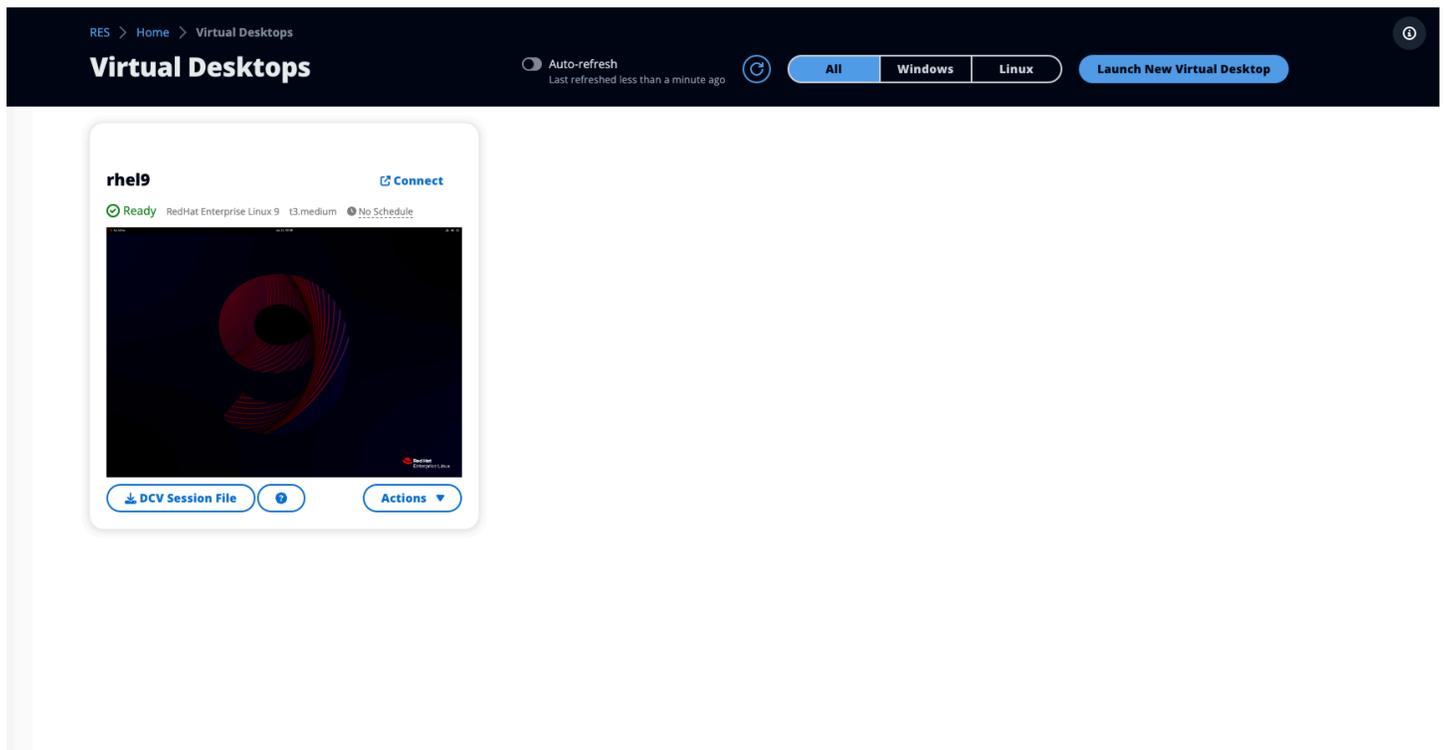
Supported operating systems

> **ⓘ Note**
>
> CentOS 7 is currently scheduled to reach end-of-life on 6/30/2024. Research and Engineering Studio version 2024.06 will be the last version to support CentOS 7.

RES currently supports launching virtual desktops using the following operating systems:

- Amazon Linux 2 (x86 and ARM64)
- CentOS 7 (x86 and ARM64)
- RHEL 7 (x86), 8 (x86), and 9 (x86)
- Ubuntu 22.04.03 (x86)
- Windows 2019, 2022 (x86)

Launch a new desktop

1. From the menu, choose **My Virtual Desktops**.
2. Choose **Launch New Virtual Desktop**.
3. Enter the details for your new desktop.
4. Choose **Submit**.

A new card with your desktop information appears instantly, and your desktop will be ready to use within 10-15 minutes. Startup time depends on the selected image. RES detects GPU instances and installs the relevant drivers.

Access your desktop

To access a virtual desktop, choose the card for the desktop and connect using either web or DCV client.

Web connection

Accessing your desktop through the web browser is the easiest method of connection.

- Choose **Connect**, or choose the thumbnail to access your desktop directly through your browser.

DCV connection

Accessing your desktop through a DCV client offers the best performance. To access via DCV:

1. Choose **DCV Session File** to download the .dcv file. You will need a DCV client installed on your system.

2. For installation instructions, choose the **?** icon.
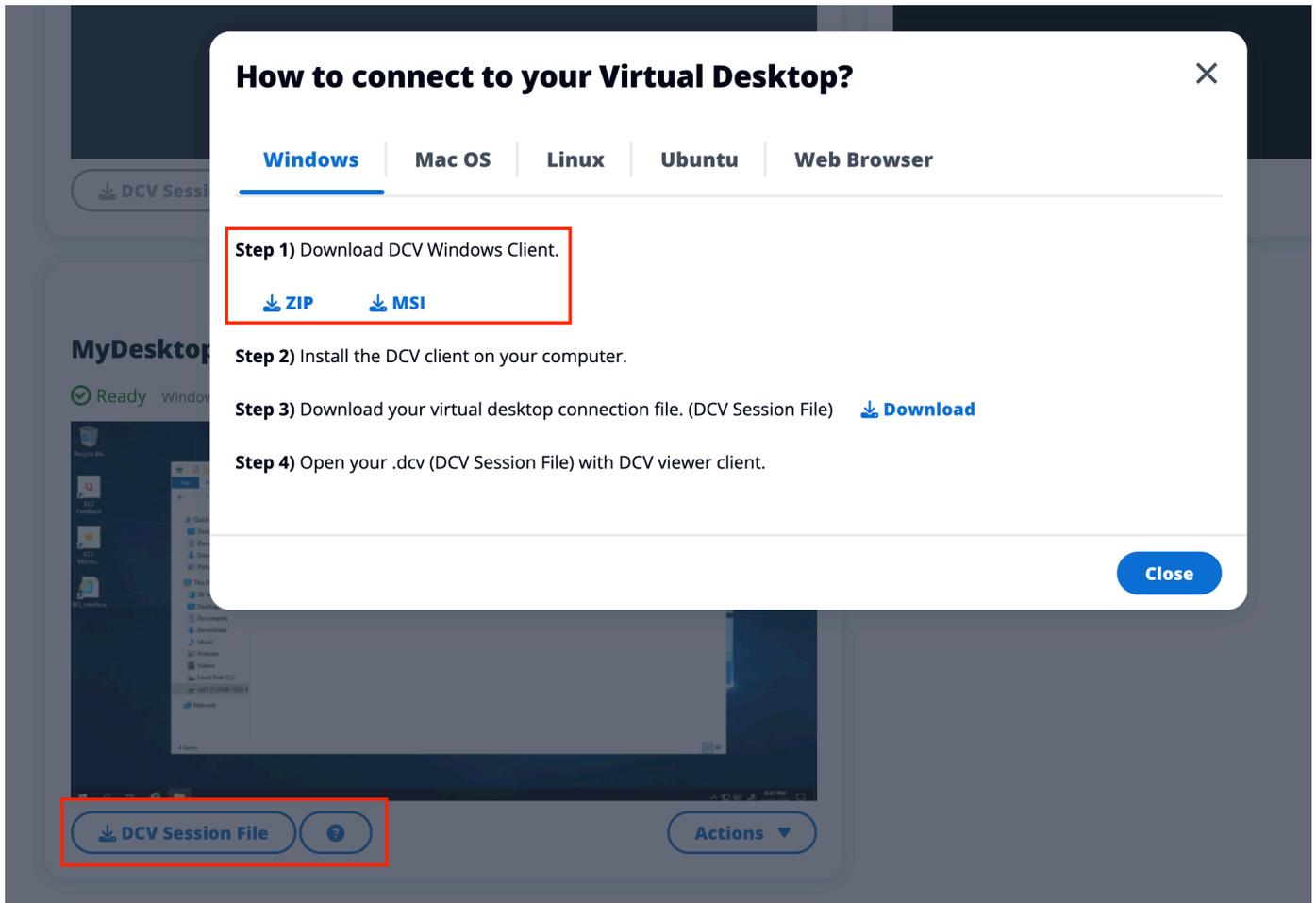
Control your desktop state

To control your desktop's state:

1. Choose **Actions**.

2. Choose **Virtual Desktop State**. You have four states to choose from:

    *   **Stop**

        A stopped session will not suffer data loss, and you can restart a stopped session at any time.

    *   **Reboot**

        Reboots current session.

    *   **Terminate**

Permanently ends a session. Terminating a session may cause data loss if you are using ephemeral storage. You should backup your data to the RES filesystem before terminating.

- **Hibernate**

    Your desktop state will be saved in memory. When you restart the desktop, your applications will resume but any remote connections may be lost. Not all instances support hibernation, and the option is only available if it was enabled during instance creation. To verify if your instance supports this state, see Hibernation prerequisites.

Modify a virtual desktop

You can update the hardware of your virtual desktop or change the session name.

1. Before making changes to the instance size, you must stop the session:

    a.   Choose **Actions**.

    b.   Choose **Virtual Desktop State**.

    c.   Choose **Stop**.

    > ⓘ **Note**
    >
    > You cannot update the desktop size for hibernated sessions.

2. Once you have confirmed the desktop has stopped, choose **Actions** and then choose **Update Session**.

3. Change the session name or choose the desktop size you would like.

4. Choose **Submit**.

5. Once your instances updates, restart your desktop:

    a.   Choose **Actions**.

    b.   Choose **Virtual Desktop State**.

    c.   Choose **Start**.

Retrieve session information

1. Choose **Actions**.

2. Choose **Show Info**.

Schedule virtual desktops

By default, virtual desktops do not have a schedule and will stay active until you stop or terminate the session. Desktops also stop if idle to prevent accidental stops. An idle state is determined by no active connection and CPU usage below 15% for at least 15 minutes. You can configure a schedule to automatically start and stop your desktop.

1. Choose **Actions**.

2. Choose **Schedule**.

3. Set your schedule for each day.

4. Choose **Save**.

Schedule for windows-session ✕

Setup a schedule to start/stop your virtual desktop to save and manage costs. The schedule operates at the cluster timezone setup by your cluster administrator.

> ⓘ **Cluster Time: October 20, 2023 4:32 PM (America/New_York)**

Monday

No Schedule	▲

Working Hours (09:00 - 17:00)	
Stop All Day	
Start All Day	
Custom Schedule	
No Schedule	✓

Thursday

No Schedule	▼

Friday

No Schedule	▼

Saturday

Stop All Day	▼

Sunday

Stop All Day	▼

Cancel   **Save**

Schedule virtual desktops
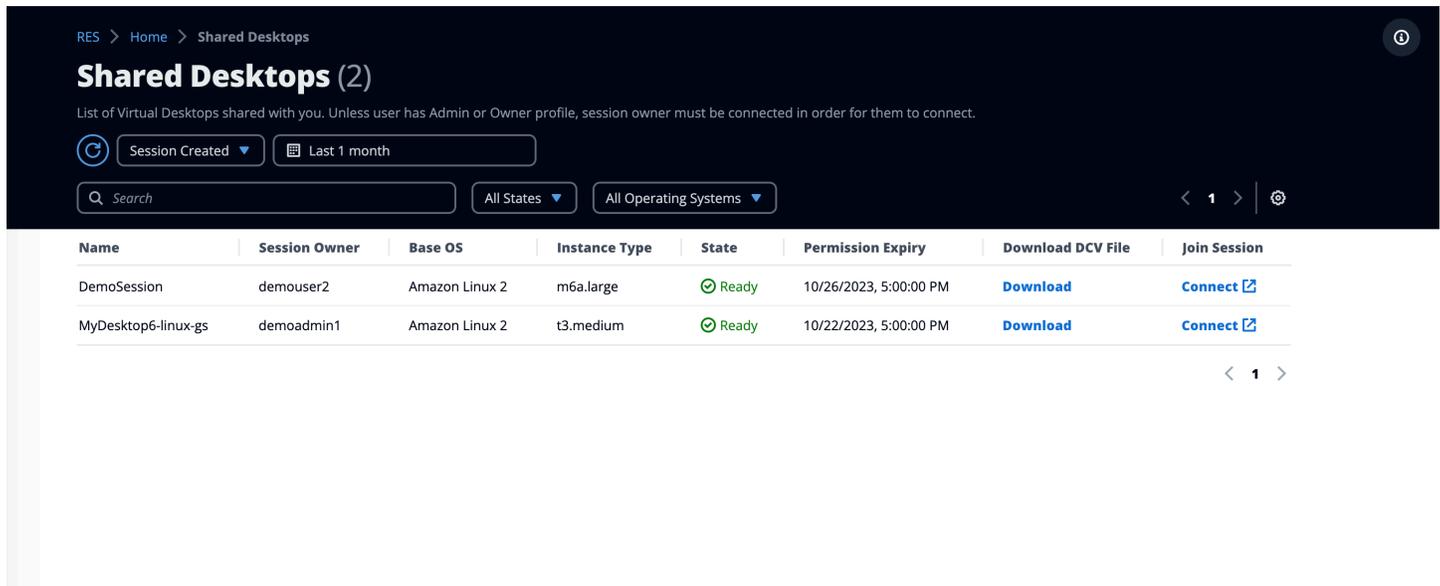
Shared desktops

On Shared Desktops, you can see the desktops that have been shared with you. In order to connect to a desktop, the session owner must be connected as well unless you are an Admin or Owner.
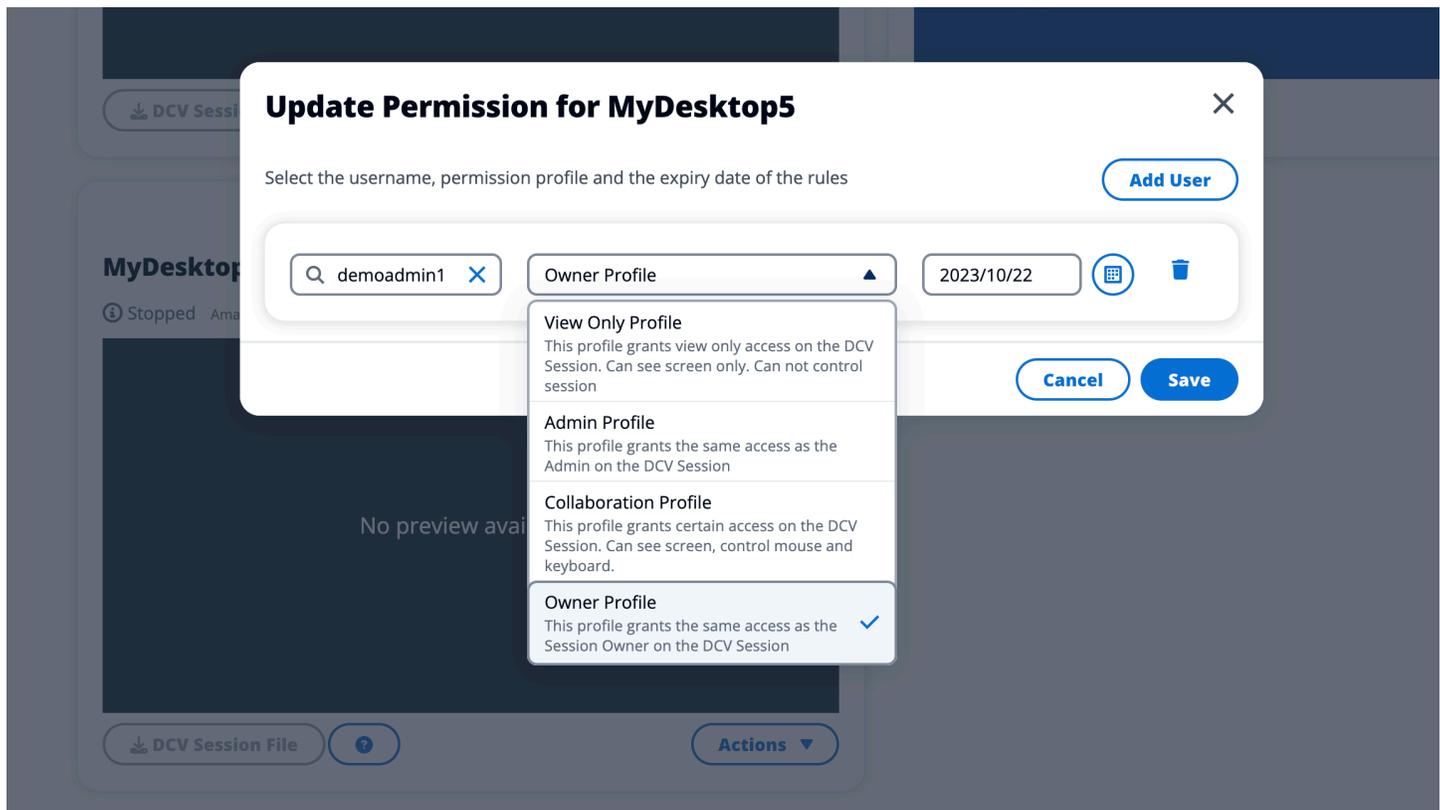

While sharing a session, you can configure permissions for your collaborators. For example, you can give read-only access to a teammate with whom you are collaborating.

Share a desktop

1. From your desktop session, choose **Actions**.

2. Choose **Session Permissions**.

3. Choose the user and permission level. You may also set an expiration time.

4. Choose **Save**.

For more information on permissions, see [the section called "Permission Profiles"](#).

Access a shared desktop

From Shared Desktops, you can view the desktops shared with you and connect to an instances. You can join by either web browser or DCV. To connect, follow the directions in [the section called "Access your desktop"](#).

File browser

File browser allows you to access filesystems through the web portal. You can manage all available files you have permission to access on the underlying filesystem. Backend storage (Amazon EFS) is available for all Linux nodes. For Linux and Windows nodes, FSx for ONTAP is available. Updating files on your virtual desktop is the same as updating a file through the terminal or web-based file browser.
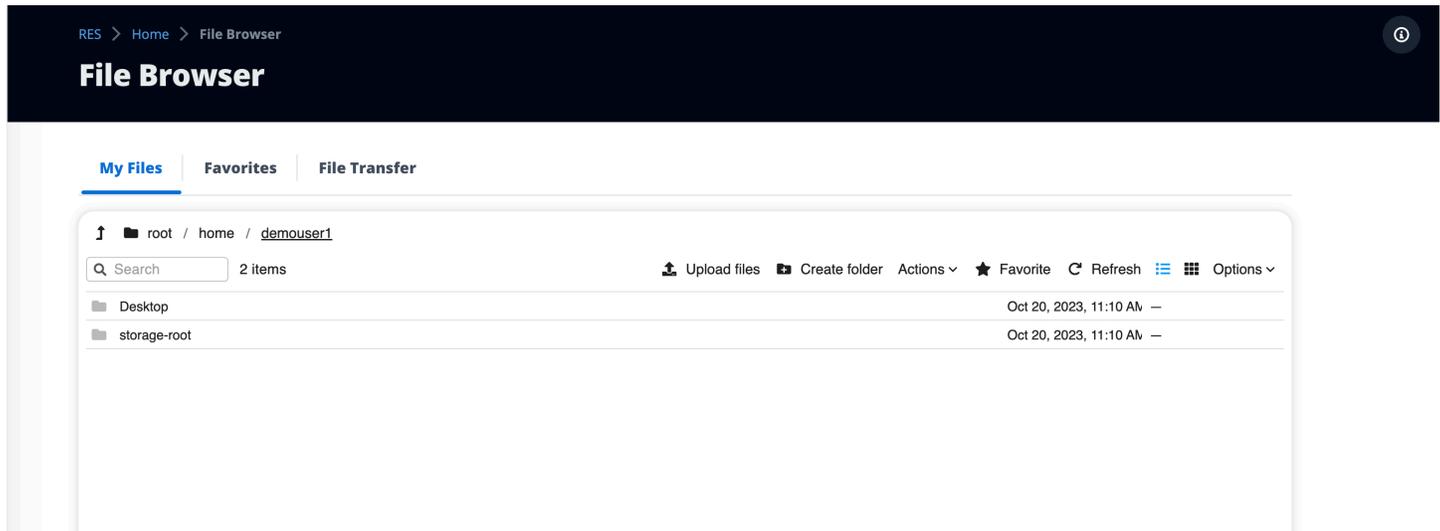
Upload file(s)

1. Choose **Upload file**.

2. Either drop files or browse for files to upload.

3. Choose **Upload (n) files**.

Delete file(s)

1. Select the file(s) you want to delete.

2. Choose **Actions**.

3. Choose **Delete files**.

Alternatively, you can also right-click any file or folder and choose **Delete files**.

Manage favorites

To pin important files and folders, you can add them to Favorites.

1. Select a file or folder.

2. Choose **Favorite**.

Alternatively, you can right-click any file or folder and choose **Favorite**.

> ⓘ **Note**
>
> Favorites are stored to the local browser. If you change your browser or clear the cache, you will need to re-pin your favorites.

Edit files

You can edit the content of text-based files within the web portal.

1. Choose the file you want to update. A modal will open with the file's content.

2. Make your updates and choose **Save**.

Transfer files

Use File Transfer to use external file transfer applications to transfer files. You can select from the following applications and follow the on-screen directions to transfer files.

- FileZilla (Windows, MacOS, Linux)

- WinSCP (Windows)

- AWS Transfer for FTP (Amazon EFS)

SSH access

To use SSH to access the bastion host:

1. From the RES menu, choose **SSH access**.

2. Follow the onscreen directions to use either SSH or PuTTY for access.

Troubleshooting

This document contains information about how to monitor the system and how to troubleshoot specific issues that may occur. If you cannot find the solution to an issue, you may be able to find additional troubleshooting topics on GitHub.

Topics

- Installation issues
- Identity management issues

Installation issues

Topics

- AWS CloudFormation stack fails to create with message "WaitCondition received failed message. Error:States.TaskFailed"
- Email notification not received after AWS CloudFormation stacks create successfully
- Instances cycling or vdc-controller in failed state
- Environment CloudFormation stack fails to delete due to dependent object error
- Error encountered for CIDR block parameter during environment creation
- CloudFormation stack creation failure during environment creation
- Creation of external resources (demo) stack fails with AdDomainAdminNode CREATE_FAILED

AWS CloudFormation stack fails to create with message "WaitCondition received failed message. Error:States.TaskFailed"

To identify the issue, examine the Amazon CloudWatch log group named `<stack-name>-InstallerTasksCreateTaskDefCreateContainerLogGroup<nonce>-<nonce>`. If there are multiple log groups with the same name, examine the first available. An error message within the logs will provide more information on the issue.

> ⓘ **Note**
>
> Confirm that the parameter values do not have spaces.

Email notification not received after AWS CloudFormation stacks create successfully

If an email invitation was not received after the AWS CloudFormation created successfully, verify the following:

1. Confirm the email address parameter was entered correctly.

   If the email address is incorrect or cannot be accessed, delete and redeploy the Research and Engineering Studio environment.

2. Check Amazon EC2 console for evidence of cycling instances.

   If there are Amazon EC2 instances with the <envname> prefix appearing as terminated and then are replaced with a new instance, there may be an issue with the network or Active Directory configuration.

3. If you deployed the AWS High Performance Compute recipes to create your external resources, confirm that the VPC, private and public subnets, and other selected parameters were created by the stack.

   If any of the parameters are incorrect, you may need to delete and redeploy the RES environment. For more information, see *Uninstall the product*.

4. If you deployed the product with your own external resources, confirm the networking and Active Directory match the expected configuration.

   Confirming that infrastructure instances successfully joined the Active Directory is critical. Try the steps in the section called "Instances cycling or vdc-controller in failed state" to resolve the issue.

Instances cycling or vdc-controller in failed state

The most probable cause of this issue is the inability for resource(s) to connect or join the Active Directory.

To verify the issue:

1. From the command line, start a session with SSM on the running instance of the vdc-controller.

2. Run sudo su -.

3. Run `systemctl status sssd`.

If the status is inactive, failed, or you see errors in the logs, then the instance was unable to join
Active Directory.

```
[root@ip-               ]# systemctl status sssd
● sssd.service – System Security Services Daemon
   Loaded: loaded (/usr/lib/systemd/system/sssd.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2023-11-14 12:12:19 UTC; 1 weeks 0 days ago
 Main PID: 31248 (sssd)          Might see "inactive"/"failed" here
   CGroup: /system.slice/sssd.service
           ─31248 /usr/sbin/sssd -i --logger=files
           ─31249 /usr/libexec/sssd/sssd_be --domain corp.res.com --uid 0 --gid 0 --logger=files
           ─31251 /usr/libexec/sssd/sssd_nss --uid 0 --gid 0 --logger=files
           ─31252 /usr/libexec/sssd/sssd_pam --uid 0 --gid 0 --logger=files

Nov 21 15:27:19 ip-10-3-144-194.ec2.internal sssd_be[31249]: GSSAPI client step 1
Nov 21 15:27:19 ip-10-3-144-194.ec2.internal sssd_be[31249]: GSSAPI client step 2
Nov 21 15:42:19 ip-10-3-144-194.ec2.internal sssd_be[31249]: GSSAPI client step 1
Nov 21 15:42:19 ip-10-3-144-194.ec2.internal sssd_be[31249]: GSSAPI client step 1
Nov 21 15:42:19 ip-10-3-144-194.ec2.internal sssd_be[31249]: GSSAPI client step 1   Might see errors
Nov 21 15:42:19 ip-10-3-144-194.ec2.internal sssd_be[31249]: GSSAPI client step 2   highlighted in
Nov 21 15:57:19 ip-10-3-144-194.ec2.internal sssd_be[31249]: GSSAPI client step 1      RED here
Nov 21 15:57:19 ip-10-3-144-194.ec2.internal sssd_be[31249]: GSSAPI client step 1
Nov 21 15:57:19 ip-10-3-144-194.ec2.internal sssd_be[31249]: GSSAPI client step 1
Nov 21 15:57:19 ip-10-3-144-194.ec2.internal sssd_be[31249]: GSSAPI client step 2
```

SSM error log

To solve the issue:

- From the same command line instance, run `cat /root/bootstrap/logs/userdata.log`
  to investigate the logs.

The issue could be one of three possible root causes.

Root cause 1: Incorrect ldap connection details entered

Review the logs. If you see the following repeated multiple times, the instance was unable to join
the Active Directory.

```
+ local AD_AUTHORIZATION_ENTRY=
+ [[ -z '' ]]
+ [[ 0 -le 180 ]]
+ local SLEEP_TIME=34
+ log_info '(0 of 180) waiting for AD authorization, retrying in 34 seconds ...'
++ date '+%Y-%m-%d %H:%M:%S,%3N'
+ echo '[2024-01-16 22:02:19,802] [INFO] (0 of 180) waiting for AD authorization,
  retrying in 34 seconds ...'
```

```
[2024-01-16 22:02:19,802] [INFO] (0 of 180) waiting for AD authorization, retrying in
 34 seconds ...
+ sleep 34
+ (( ATTEMPT_COUNT++ ))
```

1. Verify the parameter values for the following were entered correctly during RES stack creation.

   - directoryservice.ldap_connection_uri

   - directoryservice.ldap_base

   - directoryservice.users.ou

   - directoryservice.groups.ou

   - directoryservice.sudoers.ou

   - directoryservice.computers.ou

   - directoryservice.name

2. Update any incorrect values in the DynamoDB table. The table is found in the DynamoDB console under **Tables**. The table name should be **[stack name]**`.cluster-settings`.

3. After updating the table, delete the cluster-manager and vdc-controller currently running the environment instances. Auto scaling will start new instances using the latest values from the DynamoDB table.

Root cause 2: Incorrect ServiceAccount username entered

If the logs return `Insufficient permissions to modify computer account`, the ServiceAccount name entered during stack creation could be incorrect.

1. From the AWS Console, open Secrets Manager.

2. Search for `directoryserviceServiceAccountUsername`. The secret should be **[stack name]**`-directoryservice-ServiceAccountUsername`.

3. Open the secret to view the details page. Under **Secret Value**, choose **Retrieve secret value** and choose **Plaintext**.

4. If the value was updated, delete the currently running cluster-manager and vdc-controller instances of the environment. Auto scaling will start new instances using the latest value from Secrets Manager.

Root cause 3: Incorrect ServiceAccount password entered

If the logs display `Invalid credentials`, the ServiceAccount password entered during stack creation might be incorrect.

1. From the AWS Console, open Secrets Manager.

2. Search for `directoryserviceServiceAccountPassword`. The secret should be **[stack name]**`-directoryservice-ServiceAccountPassword`.

3. Open the secret to view the details page. Under **Secret Value**, choose **Retrieve secret value** and choose **Plaintext**.

4. If you forgot the password or you are unsure if the entered password is correct, you can reset the password in Active Directory and Secrets Manager.

    a. To reset the password in AWS Managed Microsoft AD:

        i.   Open the AWS Console and go to AWS Directory Service.

        ii. Select the **Directory ID** for your RES directory, and choose **Actions**.

        iii. Choose **Reset user password**.

        iv. Enter the ServiceAccount username.

        v.   Enter a new password, and choose **Reset password**.

    b. To reset the password in Secrets Manager:

        i.   Open the AWS Console and go to Secrets Manager.

        ii. Search for `directoryserviceServiceAccountPassword`. The secret should be **[stack name]**`-directoryservice-ServiceAccountPassword`.

        iii. Open the secret to view the details page. Under **Secret Value**, choose **Retrieve secret value** and choose **Plaintext**.

        iv. Choose **Edit**.

        v.   Set a new password for the ServiceAccount user and choose **Save**.

5. If the value was updated, delete the currently running cluster-manager and vdc-controller instances of the environment. Auto scaling will start new instances using the latest value.

Environment CloudFormation stack fails to delete due to dependent object error

If the deletion of the **[env-name]**-vdc CloudFormation stack fails due to a dependent object error such as the vdcdcvhostsecuritygroup, this could be due to an Amazon EC2 instance that was launched into a RES-created subnet or security group using the AWS Console.

To resolve the issue, find and terminate all Amazon EC2 instances launched in this manner. You can then resume the environment deletion.

Error encountered for CIDR block parameter during environment creation

When creating an environment, an error appears for the CIDR block parameter with a response status of [FAILED].

Example of error:

```
Failed to update cluster prefix list:
        An error occurred (InvalidParameterValue) when calling the
 ModifyManagedPrefixList operation:
        The specified CIDR (52.94.133.132/24) is not valid. For example, specify a CIDR
 in the following form: 10.0.0.0/16.
```

To resolve the issue, the expected format is x.x.x.0/24 or x.x.x.0/32.

CloudFormation stack creation failure during environment creation

Creating an environment involves a series of resource creation operations. In some Regions, a capacity issue may occur which causes a CloudFormation stack creation to fail.

If this occurs, delete the environment and retry the creation. Alternatively, you can retry the creation in a different Region.

Creation of external resources (demo) stack fails with AdDomainAdminNode CREATE_FAILED

If the demo environment stack creation fails with the following error, it may be due to Amazon EC2 patching occurring unexpectedly during the provisioning after instance launch.

```
AdDomainAdminNode CREATE_FAILED   Failed to receive 1 resource signal(s) within the
  specified duration
```

To determine the cause of failure:

1. In the SSM State Manager, check if patching is configured and if it is configured for all instances.

2. In the SSM RunCommand/Automation execution history, check if execution of a patching related SSM document coincides with an instance launch.

3. In the log files for the environment's Amazon EC2 instances, review the local instance logging to determine if the instance rebooted during provisioning.

If the issue was caused by patching, delay patching for the RES instances at least 15 minutes post-launch.

Identity management issues

Most issues with single sign-on (SSO) and identity management occur due to misconfiguration. For information on setting up your SSO configuration, see:

- the section called "Setting up SSO with IAM Identity Center"

- the section called "Configuring your identity provider for single sign-on (SSO)"

To troubleshoot other issues related to identity management, see the following troubleshooting topics:

Topics

- When logging into the environment, I immediately return to the login page

- "User not found" error when trying to log in

- User added in Active Directory, but missing from RES

- User unavailable when creating a session

- Size limit exceeded error in CloudWatch cluster-manager log

When logging into the environment, I immediately return to the login page

This issue occurs when your SSO integration is misconfigured. To determine the issue, check the controller instance logs and review the configuration settings for errors.

To check the logs:

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. From **Log groups**, find the group named /*<environment-name>*/cluster-manager.
3. Open the log group to search for any errors in the log streams.

To check the configuration settings:

1. Open the DynamoDB console at https://console.aws.amazon.com/dynamodb/.
2. From **Tables**, find the table named *<environment-name>*.cluster-settings.
3. Open the table and choose **Explore table items**.
4. Expand the filters section, and enter the following variables:

   - **Attribute name** – key
   - **Condition** – contains
   - **Value** – sso

5. Choose **Run**.
6. In the returned string, verify that the SSO configuration values are correct. If they are incorrect, change the sso_enabled key's value to **False**.

   ### Edit item

   You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. Learn more [↗]

   Attributes

⊞ Attribute name	Value
key *- Partition key*	identity-provider.cognito.sso_enabled
value	◯ True ⦿ False

7. Return to the RES user interface to reconfigure the SSO.

"User not found" error when trying to log in

If you receive the error "User not found" when logging in to the RES interface, the user is present in Active Directory, but not present in RES. If you recently added the user to AD, they are possibly not synced to RES. RES syncs hourly, so you may need to wait and check that the user was added after the next sync. To sync immediately, follow the steps in the section called "User added in Active Directory, but missing from RES".

If the user is present in RES:

1. Ensure the attribute mapping is configured correctly. For more information, see the section called "Configuring your identity provider for single sign-on (SSO)".

2. Ensure that the SAML subject and SAML email both map to the user's email address.

User added in Active Directory, but missing from RES

If you have added a user to the Active Directory but they are missing in RES, the AD sync needs to be triggered. The AD sync is performed hourly by a Lambda function to import AD entries to the RES environment. Occasionally, there is a delay until the next sync process runs after adding new users or groups. You can initiate the sync manually from the Amazon Simple Queue Service.

Initiate the sync process manually:

1. Open the Amazon SQS console at https://console.aws.amazon.com/sqs/.

2. From **Queues**, select `<environment-name>-cluster-manager-tasks.fifo`.

3. Choose **Send** and receive messages.

4. For **Message body**, enter:

   `{ "name": "adsync.sync-from-ad", "payload": {} }`

5. For **Message group ID**, enter: `adsync.sync-from-ad`

6. For **Message deduplication ID**, enter a random alpha-numeric string. This entry must be different from all calls within five minutes or the request will be ignored.

User unavailable when creating a session

If you are an administrator creating a session, but find that a user who is in the Active Directory is not available when creating a session, the user may need to log in for the first time. Sessions can only be created for active users. Active users must log into the environment at least once.

Size limit exceeded error in CloudWatch cluster-manager log

```
2023-10-31T18:03:12.942-07:00 ldap.SIZELIMIT_EXCEEDED: {'msgtype': 100, 'msgid': 11,
 'result': 4, 'desc': 'Size limit exceeded', 'ctrls': []}
```

If you receive this error in the CloudWatch cluster-manager log, the ldap search may have returned too many user records. To fix this issue, increase your IDP's ldap search result limit.

Notices

Each Amazon EC2 instance comes with two Remote Desktop Services (Terminal Services) licenses for administration purposes. This [information](#) is available to help you provision these licenses for your administrators. You can also use [AWS Systems Manager Session Manager](#), which enables remoting into Amazon EC2 instances without RDP and without a need for RDP licenses. If additional Remote Desktop Services licenses are needed, Remote Desktop user CALs should be purchased from Microsoft or a Microsoft license reseller. Remote Desktop users CALs with active Software Assurance have License Mobility benefits and can be brought to AWS default (shared) tenant environments. For information on bringing licenses without Software Assurance or License Mobility benefits, please see [this section](#) of the FAQ.

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents AWS current product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided "as is" without warranties, representations, or conditions of any kind, whether express or implied. AWS responsibilities and liabilities to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Research and Engineering Studio on AWS is licensed under the terms of the of the Apache License Version 2.0 available at [The Apache Software Foundation](#).

Revisions

For more information, see the [CHANGELOG.md](#) file in the GitHub repository.

Date	Change
November 2023	Initial release
December 2023	GovCloud directions and templates added
January 2024	Release version 2024.01
February 2024	Release version 2024.01.01 — updated deployment template
March 2024	Additional troubleshooting topics, CloudWatch Logs retention, uninstall minor versions
April 2024	Release version 2024.04 — RES-ready AMIs and project launch templates
June 2024	- Release version 2024.06 — Ubuntu support, Project owner permissions. - User Guide: added [Create a demo environment](#)