AWS Innovation Sandbox

Implementation Guide

AWS Innovation Sandbox: Implementation Guide

Table of Contents

Provision isolated, self-contained environments to securely evaluate, explore, and build proof-of-concept (POC) applications that run on AWS

Publication date: *August 2021*

The AWS Innovation Sandbox solution provisions isolated, self-contained, environments to help developers, security professionals, and infrastructure teams to securely evaluate, explore, and build proof-of-concepts (POCs) using AWS services and third-party applications that run on AWS.

The sandbox environment implements security controls (p. 7) to manage access and permissions through a browser-based Amazon AppStream 2.0 connection, minimizing the risk of data exfiltration from the user's network environment.

This solution includes the following key features:

- **Account isolation:** Create sandbox accounts within an existing AWS Organizations with networking isolation to keep existing accounts secure.
- **Secure guardrails:** Secure controls with custom AWS Identity and Access Management (IAM) roles to allow users to experiment freely while restricting administrative changes to the sandbox account.
- **Detective controls:** Amazon CloudTrail logs are activated, stored, and secured to ensure sandbox activities' auditing.
- **Data movement restrictions:** Prevents users from uploading data directly from their local machines. Data access is controlled by AWS Innovation Sandbox administrators.

This solution also creates IAM roles (p. 7) that allow elevated access to the sandbox account to allow environment customization, as needed.

This implementation guide describes architectural considerations and configuration steps for deploying AWS Innovation Sandbox in the Amazon Web Services (AWS) Cloud. It includes links to AWS CloudFormation templates that launch and configure the AWS services required to deploy this solution using AWS best practices for security and availability.

This guide is intended for IT architects, developers, DevOps, data analysts, and marketing technology professionals who have practical experience architecting in the AWS Cloud.

Cost

You are responsible for the cost of the AWS services used while running this solution. As of May 2021, the cost for running this solution with the default settings in US East (N. Virginia) Region and creating the default AWS Innovation Sandbox accounts in your AWS Organizations account (deploying the `aws-innovation-sandbox` CloudFormation template) is approximately **$70.43 per month**. This cost estimate also accounts for Amazon Virtual Private Cloud (Amazon VPC), AWS Transit Gateway, and Amazon Simple Storage Service (Amazon S3) that is initiated in the solution's accounts. Refer to Table 1 for the cost breakdown.

Table 1: Cost estimate for the solution's accounts

AWS service	Cost per month
Costs for management account	
• AWS Transit Gateway • Amazon S3 • NAT Gateway	• $18.45 • $0.16 • ~$33.21
Costs for sandbox account	
• AWS Transit Gateway • Amazon S3	• $18.45 • $0.16
Total:	**~$70.43 / month**

For additional pricing information for AWS Transit Gateway and Amazon S3, refer to the AWS Transit Gateway pricing and Amazon VPC pricing pages.

Cost for running AppStream 2.0

Additional charges accrue when Amazon AppStream 2.0 is initiated (deploying the `aws-innovation-sandbox-appstream` CloudFormation template). Charges will vary depending on the following factors:

- The Amazon Elastic Compute Cloud (Amazon EC2) instance type you choose to deploy, which is charged at an hourly rate.
- The total time that the AppStream 2.0 instance is running.
- The size of the AWS CloudTrail logs and Amazon Virtual Private Cloud (Amazon VPC) flow logs you choose to store in the Amazon Simple Storage Service (Amazon S3) buckets.
- The number of developers and other end users that receives access to the sandbox account.

A set user fee is for each authorized user that launches a streaming session from an Amazon AppStream 2.0 fleet is also incurred during the month. User fees are not incurred for administrators connecting to and using image builders to create images.

The following table provides cost estimates for AppStream 2.0 configuration with different numbers of end users given access to the sandbox account. These developers run on-demand Amazon EC2 instances five hours per day from Monday through Friday. These instances are idle the remainder of the time on a weekly basis.

Table 2: Cost estimate for AppStream 2.0 configuration

Dimensions	Cost based on the number of end users accessing the sandbox account		
	5	15	25
• Fleet instance type: stream.standard.medium • Number of hours used: 50 hours x instance fee per hour: $0.10 • Number of hours unused in the month: 190 hours x stopped instance fee per hour: $0.025 • User fee: $4.19 x number of users	~$30.70	~$72.60	~$114.50

Cost for testing AWS services in a sandbox account

Additional charges will apply if you test AWS services in your sandbox account. Data transfer charges may also apply depending on the data traffic that occurs between your sandbox account and your management account. Refer to the following example use cases and cost estimates to give you an idea of costs for using AWS services in the AWS Innovation Sandbox solution.

> **Note**
> The following use cases are examples only. Costs will vary based on the AWS services and third-party applications you run in the sandbox account.

Use case #1: AWS Innovation Sandbox running an Amazon Elastic Compute Cloud (Amazon EC2) instance in the sandbox account.

AWS service	Dimensions	Cost per month
Amazon AppStream 2.0	• Medium AppStream 2.0 configuration using a stream.standard.medium fleet instance type • AppStream 2.0 runs 50 hours per month • Image builder runs 4 hours per month	$72.60/ month
Amazon EC2	• Instance type: m5.large-linux • Instance runs 24x7	$46.80/month
	Total:	**~$119.40/month**

Use case #2: AWS Innovation Sandbox running an Amazon EMR big data processing workload in the sandbox account.

AWS service	Dimensions	Cost per month
Amazon AppStream 2.0	• Medium AppStream 2.0 configuration using a stream.standard.medium fleet instance type • AppStream 2.0 runs 50 hours per month • Image builder runs 4 hours per month	$72.60
Amazon S3	• Storage: ~750 GB per month	$17.25
Amazon EMR	• Instance type: c4.2xlarge • 1 primary node with two core nodes	$229.95
AWS Transit Gateway	• Data transfer: approximately 1 TB per month	$20.00
	Total:	**~$360.14/month**

Prices are subject to change. For full details, refer to the pricing webpage for each AWS service you will use in this solution.

Architecture overview

Deploying this solution with the default parameters builds the following environment in the AWS Cloud.
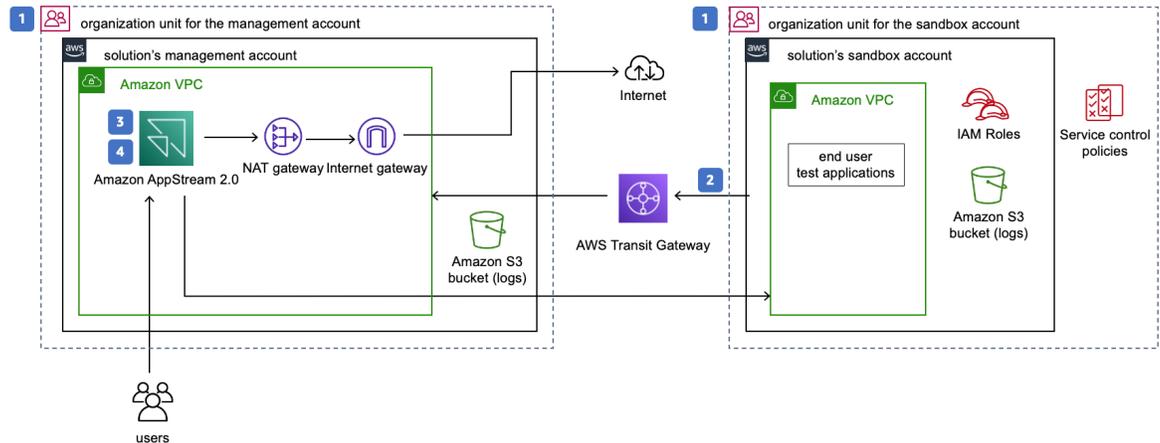
Figure 1: AWS Innovation Sandbox architecture on AWS

This solution deploys two AWS CloudFormation templates in your AWS Organizations account and sets up the following:

1. The first AWS CloudFormation template, `aws-innovation-sandbox`, creates two new AWS accounts and two new organizational units (OUs):
   - An organizational unit containing the management account, an Amazon Virtual Private Cloud (Amazon VPC) running a NAT gateway, an AWS Transit Gateway, and an internet gateway.
   - An organizational unit containing the sandbox account and an Amazon VPC.
2. The solution's sandbox account has no direct access to the Internet. Ingress and egress traffic to this sandbox account are routed through AWS Transit Gateway to the solution's management account. Access to the sandbox account is restricted via the AWS Identity and Access Management (IAM) condition key `aws:SourceIp`, to allow access only from the management account (allowing for a self-contained environment).
3. An AppStream 2.0 image is created by the customer with required applications and tools.
4. The second CloudFormation template, `aws-innovation-sandbox-appstream`, uses the image created in Step 3 to launch an Amazon AppStream 2.0 instance fleet, where end users connect to access the sandbox account.

For redundancy, the Amazon VPCs are created with subnets in two Availability Zones (AZs) for high availability. The NAT gateway and Amazon AppStream 2.0 fleet are deployed across these two AZs. The Transit Gateway are connected to both subnets.

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

Solution components

AWS accounts

This solution creates two AWS accounts in your AWS Organizations account:

- Management account: Access is restricted to the solution's administrator. This account hosts the core infrastructure and control access, including:
  - Hosting the Amazon AppStream 2.0 cluster and instances.
  - Serving as the egress and ingress entry points for traffic from the Internet and the sandbox account, allowing inspection and control.
  - Managing AWS infrastructure requirements such as monitoring and quotas.
- Sandbox account: This account is used by developers and other end users to launch their experiments using AWS services or third-party services.

Amazon AppStream 2.0

AppStream 2.0 instances are used to provide logical isolation between the sandbox account and other AWS accounts within your organization. You determine the user applications that are allowed in the AppStream 2.0 instances. You can create a standard AppStream 2.0 image from these user applications and deploy them across AppStream 2.0 instances. For more information about the AppStream 2.0 image, refer to Images in the *Amazon AppStream 2.0 Admin Guide*.

Security

When you build systems on AWS infrastructure, security responsibilities are shared between you and AWS. This shared 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.

Sandbox account security

By default, the solution's sandbox account has no direct access to the Internet. Ingress and egress traffic to this sandbox account are routed through AWS Transit Gateway to the solution's management account, and sandbox account is accessible only through the AppStream 2.0 instance on the management account.

Amazon AppStream 2.0 security

By default, the Amazon AppStream 2.0 fleet has open outbound access to the Internet. You have the option to restrict access from the fleet to a list of domains, or to only allow access to the AWS Management Console. For such restrictions we recommend the use of the AWS Network Firewall or a third-party firewall. For information on internet access in AppStream 2.0, refer to Internet Access in the *Amazon AppStream 2.0 Administration Guide*.

Copy and paste operations have been turned off by default. If you would like to change clipboard options, refer to Create an Appstream 2.0 fleet and stack in the *Amazon AppStream 2.0 Administration Guide*.

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 solution creates the following IAM roles:

- **SandboxLoginRole:** This role allows your development team to sign in to the sandbox account using the IAM condition key `aws:SourceIp`, which allows access only form the management account (providing for a self-contained environment). This role has a trust set with the management account so that only authenticated and authorized principals can assume this role and log in to the sandbox account.
- **SandboxAdminExecutionRole:** This role performs the administration activities in the sandbox account, including create/modify IAM Roles, manage identities, manage network configuration, etc. This role has an established trust with the root account, which is where this solution is launched. We recommend that access to this role be tracked and limited to only those persons with administrative responsibilities for this solution.
- **SandboxServiceRole:** Many AWS services require that you use roles to allow the service to access resources in other services on your behalf, called AWS service role. This role is pre-configured for such function and can be customized based on user requirements.

Service control policies

Service control policies (SCPs) are a type of organization policy that you can use to manage permissions in your organization. This solution creates two SCPs and attaches them to the sandbox organization unit. These policies guardrail the sandbox account set up.

- **Sandbox-guardrail:** This SCP contains a set of policies to ensure default setup in the Sandbox accounts are not altered by any administrators other than those with the **SandboxAdminExecutionRole** IAM role. This SCP establishes the following restrictions:
  - Denies edits to and deletion of the Amazon VPC flow log, AWS CloudTrail configurations, and associated logs in Amazon S3 buckets; ensuring that the original records are retained for audit purposes.
  - Denies the creation of public Amazon S3 buckets.
  - Denies the creation of new IAM roles.
  - Denies edits to the three IAM roles that are created during deployment.

- **Sandbox-vpc-guardrail:** This SCP is attached to the sandbox organizational unit. It contains a set of policies to restrict changes to the Amazon VPC that is created during the setup and ensures that egress traffic goes through that Amazon VPC only. The following restrictions apply to administrators that are not assigned the **SandboxAdminExecutionRole** IAM role.
  - Denies the creation of additional internet gateways and prevents the attachment of an internet gateway to an Amazon VPC.
  - Denies the creation of egress-only internet gateways.
  - Denies the creation of additional Amazon VPCs and refuses VPC peering connections.
  - Denies create and accept of Transit Gateway Peering attachment.
  - Denies create and update of global accelerator.
  - Denies changes to VPC and subnets Sandbox.

Design considerations

AWS Organizations setup

This solution requires AWS Organizations be activated and set up in your AWS account. If you do not have AWS Organizations set up then refer to the Tutorial: Creating and configuring an organization in the *AWS Organizations User Guide*.

If you have an active AWS Organizations account, you can deploy this solution in your existing account. This solution does not impact any of your existing OU's or AWS accounts. If you prefer to deploy this solution in a dedicated AWS Organizations account, you can set up a new account.

This solution creates two AWS accounts–a management account and a sandbox account–and two organizational units (OUs) in AWS Organizations. The management account contains the Amazon AppStream 2.0 instance. The sandbox account is for your team of developers and analysts to test and experiment with AWS services and other applications. You can have multiple sandbox accounts, which can be managed by a single management account. The two OUs provide security controls, with one OU assigned to each AWS account.

Service Control Policies (SCPs) must be activated at the AWS Organizations level. SCPs are automatically applied when you activate all AWS Organizations features. For information to set up AWS Organizations prior to deploying this solution's AWS CloudFormation template, refer to Prerequisites in the Automated deployment section of this guide.

Regional deployments

This solution uses Amazon AppStream 2.0, which is not currently available in all AWS Regions. You must launch this solution in an AWS Region where Amazon AppStream 2.0 is available. For the most current availability by Region, refer to the AWS Regional Services List.

AWS CloudFormation templates

This solution uses AWS CloudFormation to automate the deployment of the AWS Innovation Sandbox solution in the AWS Cloud. It includes the following CloudFormation templates, which you can download before deployment:

aws-innovation-sandbox: Use this template to launch AWS Innovation Sandbox accounts, networking infrastructure, Amazon Simple Storage Service buckets for logging, and AWS Organizations OUs. The default configuration deploys two Amazon Virtual Private Clouds (Amazon VPCs) with NAT Gateway, but you can also customize the template based on your specific needs.

aws-innovation-sandbox-appstream: Use this template to launch the Amazon AppStream 2.0 fleet. The default configuration deploys two streaming instances, but you can also customize the template based on your specific needs.

Automated deployment

Before you launch the automated deployment, review the architecture, network security, and other considerations discussed in this guide. Follow the step-by-step instructions in this section to configure and deploy the AWS Innovation Sandbox solution into your account.

Time to deploy: Approximately 20 minutes

Prerequisites

This solution requires AWS Organizations to be active in your AWS account. Additionally, all features in your AWS Organizations account must be activated to support the use of Service Control Policies.

- If you have not activated AWS Organizations, refer to the Tutorial: Creating and configuring an organization in the *AWS Organizations User Guide*.
- If you have not activated all the features in AWS Organizations, refer to Enabling all features in your organization in the *AWS Organizations User Guide*.

After you have verified that you have AWS Organizations and all features activated in your AWS account, proceed to the Deployment overview to deploy the CloudFormation templates into your AWS account where AWS Organizations is active.

Launch the stack

Important
This solution includes an option to send anonymous operational metrics to AWS. We use this data to better understand how customers use this solution and related services and products. AWS owns the data gathered though this survey. Data collection is subject to the AWS Privacy Policy.
To opt out of this feature, download the template, modify the AWS CloudFormation mapping section, and then use the AWS CloudFormation console to upload your template and deploy the solution. For more information, refer to the *Collection of operational metrics* (p. 27) section of this guide.

This automated AWS CloudFormation template deploys AWS Innovation Sandbox in the AWS Cloud. Review the Prerequisites before launching the stack.

Note
You are responsible for the cost of the AWS services used while running this solution. For more details, visit the Cost (p. 2) section in this guide, and refer to the pricing webpage for each AWS service you will be using in this solution.

1. Sign in to the AWS Management Console and select the button to launch the `aws-innovation-sandbox` AWS CloudFormation template.

   Alternatively, you can download the template as a starting point for your own implementation.

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

3. On the **Create stack** page, verify that the correct template URL shows in the **Amazon S3 URL** text box and choose **Next**.

4. On the **Specify stack details** page, assign a name to your solution stack. For information about naming character limitations, refer to IAM and STS Limits in the *AWS Identity and Access Management User Guide*.

5. Under **Parameters**, review the parameters for the template and modify them as necessary. This solution uses the following default values.

Parameter	Default	Description
AppStream Management Account Email	*<Requires input>*	Enter a valid email address to register the solution's management account.
AppStream Management Account Name	*<Requires input>*	Specify a name for the solution's management account. This field is limited to 50 alphanumeric characters and special characters are not supported.
AppStream Management OU Name	*<Requires input>*	Specify a name for the solution's management OU, which contains the management account. This field is limited to 128 alphanumeric characters and special characters are supported.
Sandbox Account Email	*<Requires input>*	Enter a valid email address to register the solution's sandbox account.
Sandbox Account Name	*<Requires input>*	Specify a name for the solution's sandbox account.
Sandbox OU Name	*<Requires input>*	Specify a name for the solution's sandbox OU, which contains the sandbox account.

6. Choose **Next.**

7. On the **Configure stack options** page, chose **Next**.

8. On the **Review** page, review and confirm the settings. Check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.

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

10. 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 20 minutes.

> **Note**
> In addition to the primary AWS Lambda functions, this solution includes the `solution-helper` Lambda function, which runs only during initial configuration or when resources are updated or deleted.

When running this solution, you will see multiple Lambda functions in the AWS console, but only the primary AWS Lambda functions are regularly active. However, do not delete the `solution-helper` Lambda function as it is necessary to manage associated resources.

Once the stack has successfully deployed, record the `ManagementAccountID` and `SandboxAccountID` for use in the proceeding steps. To obtain these IDs, navigate to the **Outputs** tab and record the **Value** for each of these IDs.

Post-deployment tasks

After the stack has successfully deployed, complete post-deployment tasks and post-configuration tasks and activities.

Post-deployment overview

Use the following steps to complete deployment of this solution on AWS. For detailed instructions, follow the links for each step.

Step 1. Create the image builder AppStream 2.0 cluster (p. 14)

- Launch the AppStream 2.0 image builder in the solution's management account.

Step 2. Create the AppStream 2.0 image (p. 15)

- Use the image builder application to create an image template that will be used by the Amazon AppStream 2.0 fleet.
- Select the applications to be installed in the template image (such as, a web browser, text editor, and developer integrated development environment (IDE)).

Step 3. Create the AppStream 2.0 fleet (p. 16)

- Launch the `aws-innovation-sandbox-appstream` AWS CloudFormation template into your AWS account.
- Enter values for the required parameters.
- Review the other template parameters, and adjust if necessary.

Step 4. Access the AppStream 2.0 instance (p. 18)

- Connect to the Amazon AppStream instance, and then connect to the Sandbox account.

Step 5. Post-configuration tasks and activities (p. 18)

Step 1. Create the image builder AppStream 2.0 cluster

The administrator for this solution's sandbox account must create the image builder AppStream 2.0 cluster from the management account. The designated administrator must complete the following steps to use the AWS Management Console to sign into the solution's management account.

If the designated administrator has a different role in your account, follow these steps to switch roles:

1. In the AWS Management Console, navigate to the top right corner, select your user name (*<username@account_ID_number>*) and choose **Switch Roles**.

2. On the **Switch Role** page, do the following:

- In the **Account** field, enter your `ManagementAccountID`.
- In the **Role** field, enter `SandboxAdminExecutionRole`.
- In the **Display Name** field, enter `Management Account`.

3. Choose **Switch Role**.

Next, the administrator can create the image builder AppStream 2.0 cluster.

1. Navigate to the Amazon AppStream 2.0 console.
2. Choose **Get Started**.
3. On the **Quick Links** page, under **Set up an AppStream 2.0 image with your own applications**, choose **Custom set up**,
4. On the **Choose Image** page, select the following filters:

- For operating system, select **Windows Server 2019 Base**.
- For the Instance family, select **General Purpose**.

5. From the filtered list of images, select the image with the most recent launch date, and then choose **Next**.
6. On the **Configure Image Builder** page, enter a **Name** and **Display Name** for the Image Builder.
7. Under **Instance Type**, select the following instance type: General Purpose, 2 vCPUs, and 4 GiB Memory.
8. Choose **Next**.
9. On the **Network Access** page, do the following:

- Verify that the **Default Internet Access** option is **not** selected.
- For the **VPC** field, select **ISMgmtStack VPC**.
- For the **Subnet 1** field, select **private_innovation_mgmtSubnet1** with the IP address 10.0.2.0/24.
- For **Security group,** select the default option.

10. Choose **Review** then choose **Launch**.

Note
Newly created accounts may have a quota limit, impeding the creation of the image builder. If you receive the following error message: `The ImageBuilder count limit was exceeded. Requested: 1, Limit: 0`, visit the AWS Support Center and choose **Create case**.

Step 2. Create the AppStream 2.0 image

Administrators create an AppStream 2.0 image to assign applications to developers and end users with access to the sandbox account. The AppStream 2.0 image contains the AWS services and third-party applications, along with the default Windows and application settings that helps end users to begin testing in the sandbox environment.

1. On the Amazon AppStream 2.0 console, left navigation menu pane, select **Images**.
2. On the **Images** page, select the **Image Builder** tab.
3. Select the Image Builder that you created in Step 2. The Image Builder may still be in progress creating your image build. Wait until the **Status** column changes from **Pending** to **Running** before proceeding to the next step.
4. Choose **Connect**. A new browser tab opens displaying the Image Builder application.

5. On the **Local User** tab, choose **Administrator**. A Windows desktop screen will display.

6. On the Windows desktop, select the **Image Assistant** application.

7. In the application, select the **Add Apps** tab and choose **Add App.**

8. Do the following to select the Firefox application:

   - Open Windows Explorer.
   - Navigate to the C: drive.
   - Open **Program Files (x86)**.
   - Open **Mozilla Firefox**.
   - Select `firefox.exe`.

9. Choose **Save**.

   > **Note**
   > To install additional applications on the AppStream 2.0 image, repeat Step 6. The application must be installed in the Image Builder first in order to be available, so use the Windows installation method to ensure the availability of the application.
   > For basic AWS operations such as using the AWS Management Console, accessing EC2 instances, or editing configuration files, we recommend using a secured browser (such as Firefox) with an SSH client (such as Putty) and a text editor (such as Notepad++).

10. Choose **Next** then choose **Next** again.

11. In the **Optimize** tab, choose **Launch.** Wait for each application to load, then choose **Continue** for each application.

12. Specify the name of the **AppStream Image** then choose **Next**.

13. Choose **Disconnect and Create the Image.** Once disconnected, you'll receive a confirmation message: `You have been disconnected from your session.`

14. Return to the Amazon AppStream console.

15. On the **Image Builder** tab, confirm that the Image Builder status shows **Snapshotting.** After the status updates to **Stopped** (approximately 30 minutes) you can continue to the next step.

Step 3. Create the AppStream 2.0 fleet

With AppStream 2.0, you create fleet instances and stacks as part of the process of streaming applications. A fleet consists of streaming instances that run the image that you specify. A stack consists of an associated fleet, user access policies, and storage configurations.

1. From the solution's management account, select the button to launch the `aws-innovation-sandbox-appstream` AWS CloudFormation template.

   You can also download the template as a starting point for your own implementation.

2. The template is launched in the US East (N. Virginia) by default. This template must be launched in the same Region as the `aws-innovation-sandbox` template. To launch the solution in a different AWS Region, use the Region selector in the console navigation bar.

   > **Note**
   > This solution uses the Amazon AppStream 2.0 service, which is currently available in specific AWS Regions only. Therefore, you must launch this solution in an AWS Region where

Amazon AppStream 2.0 is available. For the most current availability by Region, refer to the AWS Regional Services List.

3. On the **Create stack** page, verify that the correct template URL shows in the **Amazon S3 URL** text box and choose **Next**.

4. On the **Specify stack details** page, assign a name to your solution stack.

5. Under **Parameters**, review the parameters for the template and modify them as necessary. This solution uses the following default values.

Parameter	Default	Description
AppStreamFleetName	*<Requires input>*	Specify a name for the AppStream 2.0 fleet.
AppStreamImageName	*<Requires input>*	Enter the name of the image that you created in Step 2.9 (p. 15).
AppStreamInstanceType	`stream.standard.medium`	Identifies the instance type that is running on the AppStream 2.0 fleet.
AppStreamSecurityGroup	`default`	Determines the security group that is attached to the AppStream 2.0 instances.
AppStreamStackName	*<Requires input>*	Specify a name for the AppStream 2.0 stack.
AppStreamSubnet1	*<Requires input>*	Select the first private subnet to attach to the AppStream 2.0 instances.
AppStreamSubnet2	*<Requires input>*	Select the second private subnet to attach to the AppStream 2.0 instances.
First Name	*<Requires input>*	Enter the first name of the user that can access the AppStream 2.0 instance.
Last Name	*<Requires input>*	Enter the last name of the user that can access the AppStream 2.0 instance.
User Email	*<Requires input>*	Enter a valid email address for the user that will access the AppStream 2.0 instance. A temporary password is generated and emailed to the user. For information about the AppStream 2.0 user experience, refer to AppStream 2.0 User Pools in the *AppStream 2.0 Admin Guide*.

6. Choose **Next.**

7. On the **Configure stack options** page, chose **Next**.

8. On the **Review** page, review and confirm the settings. Check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.

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

10. 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 ten minutes.

11. After the stack is successfully deployed, navigate to the Amazon AppStream 2.0 console.

12. From the left navigation menu, select **Fleet**, select the fleet that was just created, choose **Action**, then choose **Start**.

Step 4. Access the AppStream 2.0 instance

After the AppStream 2.0 cluster and fleet are created, users can access the AppStream 2.0 instance that provides access to the sandbox account.

After the AppStream 2.0 fleet is successfully created, an email is generated and sent to the **User Email** address that you entered in Step 3 (p. 16). This email contains the link to access the AppStream application and a temporary password to sign in to the application.

1. Use the link from the email to access the AppStream 2.0 instance.

2. On the AppStream 2.0 instance log in page, enter the registered email address for the username and enter the temporary password.

3. Create a new password and choose **Next**.

4. From the list of applications installed on the AppStream 2.0 instance, select **Firefox.**

You can now use the Firefox browser to access the sandbox account

> **Note**
> If you get an error message `No applications available`, verify that the fleet is running. Review Steps 3.9 and 3.10 (p. 16).

Step 5. Post-configuration tasks and activities

The sandbox account is a secure environment, requiring an administrator management account to assign developers and other end users to this account using IAM authentication. Specifically, assignments must be made to the `SandboxLoginRole` IAM role, which was created when you deployed the `aws-innovation-sandbox` CloudFormation template. Assignments using IAM authentication can be accomplished in numerous ways, including:

- (Recommended option): Using the IAM identity providers (IdP) and federation to link users outside of AWS to the `SandboxLoginRole` IAM role.
- Manually assigning users to the `SandboxLoginRole` IAM role or the managed policy created by the solution as a permission policy.

If multiple developers will have access to the sandbox account, we recommend that you create individual IAM users for each developer to maintain access controls and monitoring.

For more information about managing user identities outside of AWS, refer to Identity providers and federation in the *IAM User Guide*.

For more information about creating end users locally, or running initial tests before implementing identity federation, refer to Creating an IAM user in your AWS account in the *IAM User Guide.*

For IAM best practices, refer to Security best practices in IAM in the *IAM User Guide*.

Additional resources

AWS services

• Amazon AppStream 2.0 • AWS CloudFormation • AWS Organizations • Amazon S3	• Amazon VPC • AWS Transit Gateway • AWS Identity and Access Management

User workflow

After this solution deploys in your AWS Organizations account, and you have added developers and other end users to the sandbox account, they can access the sandbox using Amazon AppStream 2.0. Diagram 2 shows the user workflow.
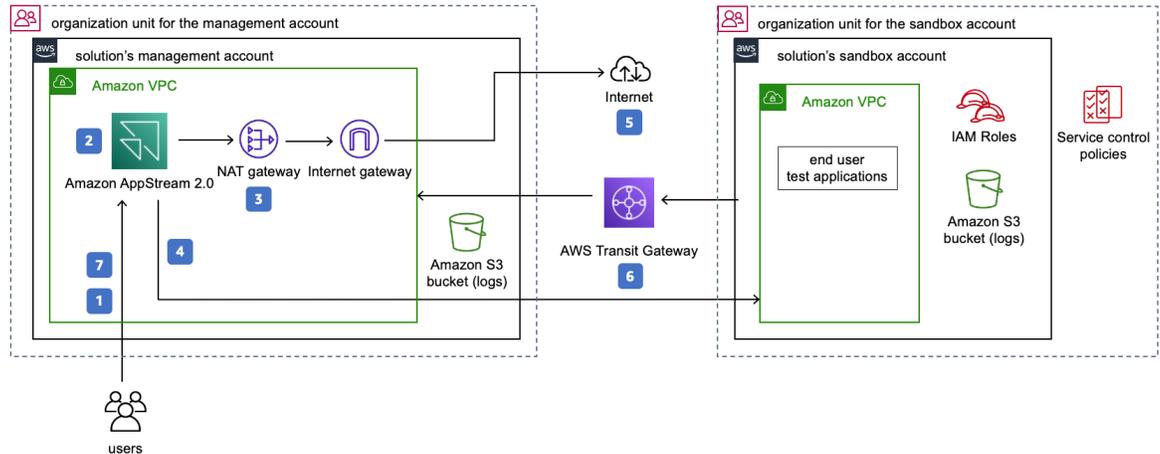
Figure 2: User workflow

1. Developers and other end users access the Amazon AppStream2.0 URL.
2. Users access the AWS Management Console using the AppStream 2.0 web browser.
3. Browser access is controlled using a firewall at the NAT gateway (the egress layer).
4. The user launches AWS services through the AWS Management Console.
5. (Optional) If the solution's administrator has set up Internet access, then resources can access the internet.
6. The Amazon Virtual Private Cloud (Amazon VPC) in the sandbox account connects to the AWS Transit Gateway containing AWS Identity and Access Management (IAM) controls which restricts this account's access to only the management account (allowing for a self-contained environment)
7. Users access the launched resources through the AppStream 2.0 web browser using a secure protocol (such as RDP, SSH, or HTTPS).

FAQ

Q: How do I ensure security for my Amazon AppStream 2.0 image?

Amazon AppStream 2.0 is a managed AWS service running within the constructs of your Amazon VPC. If your organization specifies infosec requirements, you can apply anti-virus protection on the image. For security guidance on your AppStream 2.0 image, refer to Infrastructure Security in Amazon AppStream 2.0 and Administer Your Amazon AppStream 2.0 Images in the *AppStream 2.0 Administration Guide*.

Q: How do I build a custom domain name for my Amazon AppStream 2.0 instance?

To build a custom domain for your Amazon AppStream 2.0 instance, refer to the Using custom domains with Amazon AppStream 2.0 blog post.

Q: Who should have access to the Amazon AppStream 2.0 management account?

While every organization is different, we recommend that your organization's security team manages the access and guardrails of the AppStream 2.0 management account.

Q: How do I protect my Amazon AppStream 2.0 instances from infrastructure security attacks?

Amazon AppStream 2.0 instances run inside an Amazon VPC that you designate. You can control the level of your instances isolation using security groups and other Amazon VPC security constructs. Additionally, you can limit the access to your AWS account containing the AppStream 2.0 instances using AWS Identity and Access Management policies and roles. For information about securing your AppStream 2.0 instances, refer to Infrastructure Security in Amazon AppStream 2.0 in the *Amazon AppStream 2.0 Administration Guide*.

Q: How do I configure idle timeout and session timeout ?

You can configure idle and session timeouts when you create the AppStream 2.0 fleet. For guidance, refer to Create an AppStream 2.0 Fleet and Stack in the *Amazon AppStream 2.0 Administration Guide*.

Q: How do I delete the AWS accounts that are created when the AWS CloudFormation templates are deployed?

AWS Innovation Sandbox creates two AWS accounts—a management account and a sandbox account—when this solution is deployed in your AWS Organizations account. These AWS accounts were created specifically to support this solution and serves no other purpose. You cannot delete these accounts and will need to contact AWS Support for assistance.

Q: How do I let developers create roles in the sandbox accounts ?

This solution sets an IAM permissive policy for role creation in the Sandbox account, but you can change this set up to fit your organization's needs. You have the following options:

- Pre-create standard sets of IAM roles and permissions to be used by end users
- Assign permission boundaries to end users
- Have a process which constantly checks roles and permissions created by end users so as to make sure they aren't overly permissive

For additional information to set up different policies and permissions, refer to Delegate permission management to developers by using IAM permissions boundaries in the AWS Security Blog and Unit testing IAM policies across multiple accounts in the AWS DevOps Blog.

Q: How do I restrict the sites that the AppStream 2.0 console can access?

Administrators with access to this solution's management account can use different AWS security services to restrict the AppStream 2.0 console's access to public websites. To enforce such controls, you can replace the existing pair of NAT Gateways by a pair of network firewalls or proxies that are able to restrict access based on website URL or IP addresses.

- Use AWS Network Firewall to specify stateless and stateful rules for network traffic restriction.
- Use a third-party firewall vendor and security product that can be deployed in the solution's management account.

Q: How do I adjust the permission of the policies associated with the AWS services?

You can adjust this solution's policies to lessen restrictions in regards to access to AWS services in the sandbox account. To edit this solution's policies, you can modify the `SandboxServiceRole` IAM role.

This IAM role uses the CalledVia condition and uses the permission of the Principal calling the role to make request calls to the resources. You could add stricter policies to the principal or the `SandboxServiceRole` to restrict calling of specific AWS services.

Q: How do I make secure or public data available to the sandbox account?

By default, this solution's sandbox account cannot access the resources in your network environment. To access your secured data, you can use `SandboxAdminExecutionRole` IAM role to create appropriate cross account access with AWS IAM role, AWS PrivateLink, etc., If you decide to share a Sandbox account to multiple developers, make sure the secured data access you are enabling in Sandbox account are allowed to be accessed by your developers.

To access public data, default configuration does not add any restriction.

Q: How should I organize the sandbox accounts (for example, by business unit, one-per team, etc.)?

If you deploy more than one sandbox account in your AWS Organizations account, you can organize these sandboxes that best fit your organizational needs. There is no recommended approach. Currently, customers have organized their sandboxes using the following approaches:

- One sandbox per business unit
- One-sandbox-per team

For more information, refer to the Organizing Your AWS Environment Using Multiple Accounts whitepaper.

Q: Does this solution integrate with existing Identity Providers?

Yes. You can set AppStream 2.0 to integrate with a SAML 2.0 based identity provider. For information about setting up single sign-on access, refer to Single Sign-on Access in the *Amazon AppStream 2.0 Administration Guide*.

To configure external identity providers for access to the sandbox account, configure the SandboxAdminExecutionRole IAM role. For information to edit this IAM role, refer to Creating IAM identity providers in the *IAM User Guide*.

Q: Can this solution integrate with AWS Control Tower?

No, as of May 2021, this solution cannot be deployed from AWS Control Tower, but this capability is being evaluated for a future release. You can deploy the CloudFormation template within your AWS Organizations account regardless of whether your account uses AWS Control Tower.

Q: Is there any cost management functionality built into this solution?

Cost management tools and resources are not integrated into this solution, but you can leverage various AWS Cost Management Services such as AWS Cost Explorer and AWS Budgets to help you manage your AWS costs. Additionally, you can leverage your existing cost management policies (for example, tagging, cost center allocation, account separation, etc.) to cascade into the sandbox account during deployment of the CloudFormation template.

Q: Can one management account support multiple sandbox accounts or should they always be 1 to 1?

Yes, one management account can support multiple sandbox accounts. By default, this solution creates one management and one sandbox account. You can customize the CloudFormation template to create multiple sandbox accounts.

Uninstall the solution

Delete the solution's accounts

Before you can uninstall the AWS Innovation Sandbox solution, you must delete the solution's management and sandbox accounts, then delete the resources provisioned in the management account. Use the following procedure to delete the solution's accounts.

Follow these steps to delete the sandbox accounts:

1. Navigate to the AWS Management Console. If you're already signed in, then you must sign out.
2. Select Root User and enter the email used to create the sandbox account.
3. Choose **Next**.
4. Choose **Forgot Password?**
5. Follow the password recovery process. Once you have your new password, return to the AWS Management Console.
6. From the top right corner, locate your username and access the drop-down menu, then choose **My Account**.
7. On your account homepage, scroll to the **Close Account** section.
8. Read the terms for closing your account.
9. Select the check boxes, and then choose **Close Account**.
10. In the confirmation box, choose **Close Account**. For more information, refer to closing AWS Accounts and Recovering lost passwords.

Delete the CloudFormation stacks

After the solution's management and sandbox accounts are deleted, you can uninstall the solution. You can uninstall this solution from either the AWS Management Console or by using the AWS Command Line Interface (AWS CLI). You must manually delete the Amazon Simple Storage Service (Amazon S3) buckets and CloudWatch Logs created by this solution. AWS Solutions Implementations do not automatically delete these resources in case you have stored data that you wish to retain.

Using the AWS Management Console

1. Sign in to the AWS CloudFormation console.
2. Select this solution'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, refer to What Is the AWS Command Line Interface in the *AWS CLI User Guide*. After confirming that the AWS CLI is available, run the following command.

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

Delete the Amazon S3 buckets

This solution is configured to retain the Amazon S3 buckets if you decide to delete the AWS CloudFormation stack to prevent accidental data loss. After uninstalling the solution, you can manually delete the S3 buckets if you do not need to retain the data. Follow these steps to delete the Amazon S3 buckets.

1. Sign in to the Amazon S3 console.
2. Choose **Buckets** from the left navigation pane.
3. Locate the *<stack-name>* S3 buckets.
4. Select one of the S3 buckets and choose **Delete**.

Repeat the steps until you have deleted all the *<stack-name>* S3 buckets.

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

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

Delete resources

During the creation of the stack, you have the option to retain or delete certain resources. However, even when selecting to delete all resources, the processes described above to uninstall the stack may leave behind a few resources in CloudWatch Logs. To complete the removal of these resources, follow these steps:

1. Sign in to the AWS CloudWatch console.
2. Navigate to **Logs**, then **Log groups** and search for Log groups that contain the name of your stack.
3. Select each of the Log groups and then under **Actions** select **Delete log group(s)**.

Collection of operational metrics

This solution includes an option to send anonymous operational metrics to AWS. We use this data to better understand how customers use this solution and related services and products. When active, the following information is collected and sent to AWS:

- **Solution ID:** The AWS solution identifier
- **Unique ID (UUID):** Randomly generated, unique identifier for each AWS Innovation Sandbox deployment
- **Timestamp:** Data-collection timestamp

Note that AWS will own the data gathered via this survey. Data collection will be subject to the AWS Privacy Policy. To opt out of this feature, complete the following steps before launching the AWS CloudFormation template.

1. Download the AWS CloudFormation template to your local hard drive.
2. Open the AWS CloudFormation template with a text editor.
3. Modify the AWS CloudFormation template mapping section as follows:

```
"Send" : {
    "AnonymousUsage" : { "Data" : "Yes" }
        },
```

to

```
"Send" : {
    "AnonymousUsage" : { "Data" : "No" }
        },
```

4. Sign in to the AWS CloudFormation console.
5. Select **Create stack**.
6. On the **Create stack** page, **Specify template** section, select **Upload a template file**.
7. Under **Upload a template file**, choose **Choose file** and select the edited template from your local drive.
8. Choose **Next** and follow the steps in Launch the stack (p. 11) in the Automated Deployment section of this guide.

Source code

Visit our GitHub repository to download the source files for this solution and to share your customizations with others. The AWS Innovation Sandbox templates are generated using the AWS Cloud Development Kit (AWS CDK). Refer to the README.md file for additional information.

Revisions

Date	Change
August 2021	Initial release

Contributors

- Rafael Suguiura
- Nickil Somanna
- Aravind Singirikonda
- Sujatha Kuppuraju
- Hunter Gillezeau
- Puneet Agarwal

Notices

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.

AWS Innovation Sandbox is licensed under the terms of the of the Apache License Version 2.0 available at The Apache Software Foundation.

AWS glossary

For the latest AWS terminology, see the AWS glossary in the *AWS General Reference*.