Amazon Connect

Administrator Guide

Amazon Connect: Administrator Guide

Table of Contents

What Is Amazon Connect?

Amazon Connect is a cloud-based contact center solution. Amazon Connect makes it easy to set up and manage a customer contact center and provide reliable customer engagement at any scale. You can set up a contact center in just a few steps, add agents from anywhere, and start to engage with your customers right away.

Amazon Connect provides rich metrics and real-time reporting that allow you to optimize contact routing to decrease wait times. You can also resolve customer issues more efficiently by putting customers in touch with the right agents. Amazon Connect integrates with your existing systems and business applications to provide visibility and insight into all of your customer interactions. Amazon Connect requires no long-term contracts, and you pay only for what you use.

Amazon Connect Instances

To create an Amazon Connect contact center, you create an Amazon Connect instance. Each instance contains all of the resources and settings related to your contact center. You can manage settings for your instance from the Amazon Connect console. You can manage settings for your contact center from within your contact center. You can create multiple instances, but each instance functions only within the AWS region in which you create it. Settings, users, metrics, and reporting are not shared between Amazon Connect instances.

Identity Management

When you create an Amazon Connect instance, you must choose how you want to manage your Amazon Connect users. Permissions to access Amazon Connect features and resources, such as opening the contact control panel (CCP), placing calls, or creating reports, are assigned to user accounts within Amazon Connect. You can choose from the following three options for identity management:

- Store users in Amazon Connect.
- Link to an existing directory using AWS Directory Service.
- Use SAML 2.0-based authentication to federate with your Amazon Connect instance and enable single sign-on.

To learn more about identity management in Amazon Connect, see Plan for User and Identity Management (p. 12).

Amazon Connect Administrator

Amazon Connect administrators set permissions, manage and generate metrics, add users, and configure all aspects of your contact center. You can grant or deny different types of permissions by assigning security profiles in Amazon Connect.

Secure Storage and Data Integrity

Secure storage and data integrity are an important part of managing recorded calls. Customer calls are recorded in real time and can contain sensitive information.

By default, AWS creates a new Amazon S3 bucket during the configuration process, with built-in encryption. You can also use existing S3 buckets. There are separate buckets for call recordings and exported reports, and they are configured independently. There is full access through Amazon Connect and control over recordings, allowing for custom retention policies. The customizable metrics reports published into Amazon S3 can be processed using the Amazon S3 API or AWS Lambda. Integrate the reports with external systems such as workforce management and business intelligence tools.

> **Note**
> We recommend that you keep the default settings for encryption.

The following security measures are supported:

- AWS Key Management Service—AWS KMS is a powerful, managed service that gives you complete control over your encryption keys. A default AWS KMS key is provided.
- ARN/ID—You can use an ARN/ID instead of an AWS KMS master key. This is an advanced option and should be attempted only if you are confident of the changes that you're going to make.

Supported Browsers

Before you start working with Amazon Connect, use the following table to verify that your browser is supported.

Browser	Version	Check your version
Google Chrome	Latest 3 versions	Open Chrome and type chrome://version in your address bar. The version is in the Google Chrome field at the top of the results.
Mozilla Firefox ESR	Latest 3 versions	Open Firefox. On the menu, choose the Help icon and then choose **About Firefox**. The version number is listed underneath the Firefox name.
Mozilla Firefox	Latest 3 versions	Open Firefox. On the menu, choose the Help icon and then choose **About Firefox**. The version number is listed underneath the Firefox name.

Service Limits

The following table provides the default limits for new Amazon Connect instances. Because the limits have been adjusted over time, the limits in place for your account may be different than the limits described here. There may even be differences between the instances created for your account. For example, if you created an instance during the period when the default limit for concurrent active calls was set to 10, your instance is limited to 10 concurrent active calls. If you create a new instance today, the limit for the instance is 100 concurrent active calls.

To start, you can create five instances per AWS account in each of AWS Regions where Amazon Connect is available. If you need more instances, or an increase (or decrease) to any of the service limits, it is easy to request an change using the Amazon Connect service limits increase form. You must be signed in to your AWS account to access the form.

Use the same form to submit a request to port your US phone number from your current carrier to Amazon Connect. For more information about porting phone numbers, see Port Your Current Phone Number (p. 19).

There is also a service limit for the countries to which you can place outbound calls from your instance. If you already have an instance, the countries that you are allowed to call may be different that those listed in the following table because we have changed the service limits over time. You can submit a service limit increase request to allow calling to additional countries, or to limit the countries that you can call from your instance.

> **Note**
> Amazon Connect is not available to customers in India using Amazon Web Services through Amazon Internet Services Pvt. Ltd (AISPL). You will receive an error message if you try to create an instance in Amazon Connect.

Item	Default limit
Amazon Connect instances per account	5
Users per instance	500
Phone numbers per instance	10
Queues per instance	50
Queues per routing profile	50
Routing profiles per instance	100
Hours of operation per instance	100
Quick connects per instance	100
Prompts per instance	500
Agent status per instance	50
Security profiles per instance	100
Contact flows per instance	100
Agent hierarchy groups per instance	50
Reports per instance	500
Scheduled reports per instance	50
Concurrent active calls per instance	100
Phone Number Porting	You can port your US phone numbers from your current carrier to Amazon Connect. For information about how to port your phone number, see Port Your Current Phone Number (p. 19).
Country Whitelisting for Outbound Calls	You can place calls to the following countries when you create a new instance: • Australia • Canada • China

Item	Default limit
	• Germany • Hong Kong • Israel • Japan • Mexico • Singapore • Sweden • United States • United Kingdom **Note** UK numbers with a 447 prefix are not allowed by default. If you need to dial these UK mobile numbers, please submit a service limit increase request.

Related Services

The following services are used with Amazon Connect:

- **AWS Directory Service**—AWS Directory Service for Microsoft Active Directory (Enterprise Edition), also known as Microsoft AD, enables your directory-aware workloads and AWS resources to use managed Active Directory in the AWS Cloud. Amazon Connect user and identity management is based on this service.
- **Amazon S3**—Amazon Simple Storage Service (Amazon S3) is object storage with a simple web service interface to store and retrieve any amount of data from anywhere on the web. Amazon Connect uses Amazon S3 as a primary data storage service/platform for call recordings and metrics reports delivered into your AWS account.
- **AWS Lambda**—Lambda allows you to build and run code quickly without provisioning or managing servers. Amazon Connect contact flows (IVR flows) are integrated with Lambda so you can build a highly personalized and dynamic IVR experience. You can build Lambda functions that communicate with CRM systems or custom services for data dips that influence customer IVR experience (such as customer segmentation and dynamic IVR menus, or account and last contact look ups). Lambda functions can also be used as notification mechanisms to external systems during specific points in the contact flow.
- **Amazon Lex**—Amazon Connect integrates with Amazon Lex to build conversational interfaces using voice and text. Amazon Lex provides the advanced deep learning functionalities of automatic speech recognition (ASR) for converting speech to text, and natural language understanding (NLU) to recognize the intent of the text, to enable you to build applications with highly engaging user experiences and lifelike conversational interactions. For more information, see the Amazon Lex Developer Guide.
- **Kinesis**—Amazon Connect integrates with Kinesis as the platform for streaming contact trace records (CTR) and agent event streams data. The data is published to Kinesis in JSON format, and include details about contacts and agent activities in your contact center. You can use this data stream to optionally process and publish them into Amazon Redshift (an AWS data warehouse service) or your custom data warehouse systems, enabling detailed analytics and reporting on your contact center data. You can leverage Amazon QuickSight (a cloud-powered business analytics service) or your own BI tools to build powerful visualizations on top of synthesized data. Additionally, this data can be streamed to Elasticsearch to query on this data using a convenient visual interface. For more information, see the Amazon Kinesis Data Streams Developer Guide.

> **Note**
> Amazon Connect does not support publishing data to streams for which server-side encryption is enabled.

- **Amazon CloudWatch**—Amazon Connect integrates with CloudWatch to provide you with real-time operational metrics for your contact center, such as total calls per second, calls rejected and throttled, percentage of concurrent calls, failed / missed calls count (errors, bad number/address, busy/line engaged), and contact flow errors. You can set up monitors on these metrics in order to stay on top of the health of your contact center. For more information, see Monitoring Amazon Connect in Amazon CloudWatch Metrics (p. 28).
- **AWS Identity and Access Management**—The AWS Management Console requires your user name and password so that any service you use can determine whether you have permission to access its resources. We recommend that you avoid using AWS account root user credentials to access AWS because root user credentials cannot be revoked or limited in any way. Instead, we recommend that you create an IAM user and add the user to an IAM group with administrative permissions. You can then access the console using the IAM user credentials. For more information, see the IAM User Guide.

  If you signed up for AWS but have not created an IAM user for yourself, you can create one using the IAM console. For more information, see Create Individual IAM Users in the *IAM User Guide*.
- **AWS Key Management Service**—Amazon Connect is integrated with AWS KMS to protect your customer data. Key management can be performed from the AWS KMS console. For more information, see What is the AWS Key Management Service in the *AWS Key Management Service Developer Guide*.

Release Notes

To help you keep track of the ongoing updates and improvements to Amazon Connect, we're now publishing monthly release notices that describe the changes we've released in the previous month.

Monthly Updates

September 2018 Updates

The following updates were released in September 2018:

Updates by category

General Updates

- Improved page loading times for the **User management** page.
- Resolved an issue that sometimes caused issues loading the **Queues** page when there were a large number of quick connects associated with a queue.

API Updates

Released the `UpdateContactAttributes` operation for the Amazon Connect API.

August 2018 Updates

The following updates were released in August 2018:

Updates by category

General Updates

- Added a restriction of 64 characters for the password length for the administrator account created during instance creation.
- Resolved an issue where the **Hours of operation** page would not load when no days were selected for a saved Hours of operation configuration.

Contact Routing

Increased the timeout for whispers to 2 minutes for outbound and queued callbacks so that agents have longer to prepare for the incoming call.

Metrics and Reporting

- Modified how the value for the Contacts abandoned metric so that calls that transfer to callbacks are not counted as abandoned contacts.

July 2018 Updates

The following updates were released in July 2018:

Updates by category

Feature Releases

- Dynamic Outbound Caller ID
- Add an Amazon Lex bot to Your Instance (p. 25)
- User Management APIs
- Queue to Queue Transfers

General Updates

- Added an error message when attempting to create an admin user during instance creation using “Administrator” as the user name. The user name Administrator is reserved for internal use, and cannot be used to create a user account in Amazon Connect.
- Added support for directory user names that include consecutive dashes.
- Added pagination when displaying security profiles in your instance so that more than 25 security profiles can be displayed.
- Performance optimizations to reduce latency when using the `StartOutboundVoiceContact` API.

Metrics and Reporting

- Resolved an issue in Real-time metrics reports where applied filters were not displayed in the settings page when an additional filter was applied. The settings page now displays the applied filters correctly.

Contact Flows

- Added drop-down menus for contact attributes to make it easier to reference attributes in a contact flows.

June 2018 Updates

The following updates were released in June 2018:

Updates by category

General Updates

- Changed the font in the UI to Amazon Ember for better readability.

Telephony and Voice

- Introduced support for using Amazon Lex bots with Amazon Connect in the US West (Oregon) Region.
- Fixed a bug that in some cases caused a call to drop when a Loop prompt occurred at the same as a call connecting to an agent.

Contact Flows

- Renamed the **Set queue** block to **Set working queue**.
- Added a **Copy to clipboard** button next to the ARN of a contact flow so you can easily copy the ARN. Choose **Show additional flow information** under the name of the contact flow in the designer to display the ARN.
- Added a new **Call phone number** block, which lets you choose the phone number from your instance to display as the caller ID in an outbound whisper flow. For more information, see Using a Call phone number block in a contact flow.
- Released contact attributes for system metrics, including a new **Get metrics** block in contact flows. For more information, see Using System Metric Attributes.

Metrics and Reporting

- Fixed an issue that caused incorrect rendering of the search field in the filters settings for some historical metrics reports.
- Fixed an issue in downloaded reports where the phone number would be blank instead of listing the phone number for calls that were callbacks.
- Login/Logout reports now support 20,000 rows per report generation, up from 10,000.

Contact Control Panel (CCP)

- Added a mute button to the CCP and a mute function to the Streams API so agents can mute and unmute active calls.

April and May 2018 Updates

The following updates were released in April and May 2018:

Updates by category

General Updates

- New Amazon Polly voices are now automatically made available in Amazon Connect as soon as they are launched. You can use new voices, such as Matthew and Léa, in your contact flows.
- Updated password enforcement for Amazon Connect user accounts to match requirements for the Amazon Connect admin account created during instance creation.
- Resolved an issue that sometimes resulted in the email addresses not being saved when updating an existing user account.

Telephony and Voice

- Service optimizations to reduce latency and improve caller ID for Japanese telephony.
- Customers can now place calls to Jersey and Guernsey in the Channel Islands.
- Added support for keypad numeric input to an Amazon Lex bots when used in an Amazon Connect contact flow. For more information, see Amazon Connect Now Supports Keypad Input with an Amazon Lex Chatbot.
- Reduced latency for the contact control panel, improving the agent user experience.

Contact Flows

- Resolved an issue with publishing a contact flow in the case where an **AWS Lambda function block** is used in a contact flow, and the input type for a parameter was changed from **Send attribute** with a **System** attribute is changed to **Send text**. These contact flows now publish successfully.
- Agent and customer whispers are now maintained with queued callbacks.
- Attributes now correctly persist with queue callbacks.
- Contact attributes are now maintained when using a **Loop prompt** block in a queue flow.

Metrics

- Data for scheduled reports is now delayed by 15 minutes to allow for most recent data to be incorporated in to reports. Previously, in some cases, report data for the final 15 minute period during the scheduled report interval did not get included in scheduled reports. This applies to all report types.
- In metric calculations, the time that an incoming call rings is attributed to idle time if the agent is in idle state before an incoming call.
- The metric **Agent on contact time** now includes time that an agent spent in an auxiliary busy state.

- Published new documentation on Amazon Connect metrics.

Contact Control Panel (CCP)

- Added a **Save** button to the settings menu for the CCP when an agent is using a desk phone. The **Save** button saves the deskphone configuration between sessions.
- Agent username is now available as part of agent configuration data in the Amazon Connect Streams API.
- Contact attributes are now available when using the streams.js (Streams API) for screenpops after queued callbacks.
- Fixed issue where for some auto-accept calls, the agent continued to hear ringing after accepting and joining the call.

Getting Started with Amazon Connect

An Amazon Connect instance is the starting point for your contact center. After you create an instance, you can edit the settings for it, which include telephony, data storage, data streaming, application integration, and contact flows. You can then launch your instance from the AWS Management Console and start using your contact center.

> **Note**
> Amazon Connect is not available to customers in India using Amazon Web Services through Amazon Internet Services Pvt. Ltd (AISPL). You will receive an error message if you try to create an instance in Amazon Connect.

After you create an Amazon Connect instance, you can claim a phone number to use for your contact center. After you claim a number, you can place a test call in to your contact center to confirm that it is working correctly. Calls are handled in the contact center using the Contact Control Panel (CCP). The CCP is built in to the Amazon Connect Contact Center Manager (CCM). For more information about how agents use the CCP, see Using the Contact Control Panel in the *Amazon Connect User Guide*.

You can edit the settings for your instance in the AWS Management Console. After you create your instance, you can access it by using the URL in the **Access URL** column. The access URL is the URL your agents, administrators, and managers use to log in to and access the CCM and the CCP. For more information, see Amazon Connect Instances (p. 1).

> **Note**
> If you use SAML-based authentication for identity management, your users must log in to your instance through your identity provider instead of using the access URL for your instance.

Before You Begin

When you sign up for Amazon Web Services (AWS), your AWS account is automatically signed up for all services in AWS, including Amazon Connect. You are charged only for the services that you use.

If you have an AWS account already, skip to the next task. If you don't have an AWS account, use the following procedure to create one.

To create an AWS account

1. Open https://aws.amazon.com/, and then choose **Create an AWS Account**.

   > **Note**
   > If you previously signed in to the AWS Management Console using AWS account root user credentials, choose **Sign in to a different account**. If you previously signed in to the console using IAM credentials, choose **Sign-in using root account credentials**. Then choose **Create a new AWS account**.

2. Follow the online instructions.

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

Plan for User and Identity Management

Before you set up your Amazon Connect instance, you should decide how you want to manage your Amazon Connect users. You cannot change the option you select for identity management after you create the instance. If you decide to change the option or directory you selected, you can delete the instance and create a new one. When you delete an instance, you lose all configuration settings and metrics data for it.

You can choose from one of the following supported identity management solutions supported in Amazon Connect:

- **Store users with Amazon Connect**—Choose this option if you want to create and manage user accounts within Amazon Connect. When you manage users in Amazon Connect, the user name and password for each user is specific to Amazon Connect. Users must remember a separate user name and password to log in to Amazon Connect.
- **Link to an existing directory**—Choose this option to use an existing directory. The directory must be associated with your account, set up in AWS Directory Service, and be active in the same Region in which you create your instance. If you plan to choose this option, you should prepare your directory before you create your Amazon Connect instance. For more information, see Use an Existing Directory for Amazon Connect Identity Management (p. 13).
- **SAML 2.0-based authentication**—Choose this option if you want to use your existing network identity provider to federate users with Amazon Connect. Users can only log in to Amazon Connect by using the link configured through your identity provider. If you plan to choose this option, you should configure your environment for SAML before you create your Amazon Connect instance. For more information, see Configure SAML for Identity Management in Amazon Connect (p. 14).

Create an Amazon Connect Instance

You can create or add an instance as follows.

To create an Amazon Connect instance

1. Open the Amazon Connect console at https://console.aws.amazon.com/connect/.
2. Do one of the following:
   - If you have not previously created an Amazon Connect instance, choose **Get started**.
   - If you have previously created an instance, choose **Add an instance**.
3. For **Step 1: Identity management** step, do one of the following:
   - To manage your users within Amazon Connect, choose **Store users within Amazon Connect**.
   - To use an existing directory where your users are managed, choose **Link to an existing directory**. For more information about using an existing directory, see Use an Existing Directory for Amazon Connect Identity Management (p. 13).
   - To use SAML-based authentication with your identity provider to federate users with Amazon Connect, choose **SAML 2.0-based authentication**. For more information about using SAML with Amazon Connect, see Configure SAML for Identity Management in Amazon Connect (p. 14).
4. For **Access URL**, enter an instance alias for your instance, and choose **Next step**.

   The name that you enter is displayed as the instance alias in the AWS Management Console, and is used as the domain in the access URL to access your contact center. The alias must be globally unique, meaning that an alias can be used only one time across all Amazon Connect instances and Regions. You cannot change the alias URL after your instance is created.
5. For **Step 2: Administrator**, do one of the following:

- If you chose **Store users with Amazon Connect** for identity management, enter the user details for an admin account, and choose **Next step**.
- If you chose **Link to an existing directory** for identity management, enter the user name for the account to use as the admin account for your instance, and choose **Next step**.

  If the user name that you enter does not exist in your directory, you can add it later.
- Choose **Skip this** to create an admin account later. To create an admin later, log in to your instance as an administrator from the Amazon Connect console.

6. For **Step 3: Telephony options**, indicate whether you'd like your contact center to accept calls, make calls, or both. You can set the user permissions within the Amazon Connect web application. The telephone number options are provided after setup.
7. For **Step 4: Data storage**, you can keep the default settings or choose **Customize settings**. For more information, see Data Storage (p. 22).
8. For **Step 5: Review and create**, review your settings and choose **Create instance**.

   Important
   This is the only time you can change the directory and domain name settings—you can edit any other setting later on.
9. After your instance is created, choose **Get started** to claim and test a phone number. Amazon Connect automatically configures your instance to use the phone number that you select.

   Note
   For information about how to keep your current phone number and use it with Amazon Connect, see Port Your Current Phone Number (p. 19).
10. (Optional) Continue to configure your instance. For more information, see Configuring Your Amazon Connect Instance (p. 22).

Use an Existing Directory for Amazon Connect Identity Management

If you are already using a in AWS Directory Service directory to manage users, you can use the same directory to manage user accounts in Amazon Connect. You can also create a new directory in AWS Directory Service to use for Amazon Connect. The directory you choose must be associated with your AWS account, and must be active in the AWS Region in which you create your instance. You can associate an AWS Directory Service directory with only one Amazon Connect instance at a time. To use the directory with a different instance, you must delete the instance with which it is already associated.

The following AWS Directory Service directories are supported in Amazon Connect:

- Microsoft Active Directory—AWS Directory Service lets you run Microsoft Active Directory as a managed service.
- Active Directory Connector—AD Connector is a directory gateway you can use to redirect directory requests to your on-premises Microsoft Active Directory.
- Simple Active Directory—Simple AD is a standalone managed directory that is powered by a Samba 4 Active Directory Compatible Server.

You cannot change the directory you select for identity management after you create the instance. If you decide to change the directory you selected, you can delete the instance and create a new one. When you delete an instance, you lose all configuration settings and metrics data for it.

There is no additional charge for using an existing or a proprietary directory in Amazon Connect. For information about the costs associated with using AWS Directory Service, see AWS Service Pricing Overview.

The following limitations apply to all new directories created using AWS Directory Service:

- Directories can only have alphanumeric names. Only the `.` character can be used.
- Directories cannot be unbound from an Amazon Connect instance after they have been associated.
- Only one directory can be added to an Amazon Connect instance.
- Directories cannot be shared across multiple Amazon Connect instances.

Configure SAML for Identity Management in Amazon Connect

Amazon Connect supports identity federation with Security Assertion Markup Language (SAML) 2.0 to enable web-based single sign-on (SSO) from your organization to your Amazon Connect instance. This allows your users to sign in to a portal in your organization hosted by a SAML 2.0–compatible identity provider (IdP), select an option to go to Amazon Connect, and be redirected to your Amazon Connect instance without having to provide separate credentials for Amazon Connect.

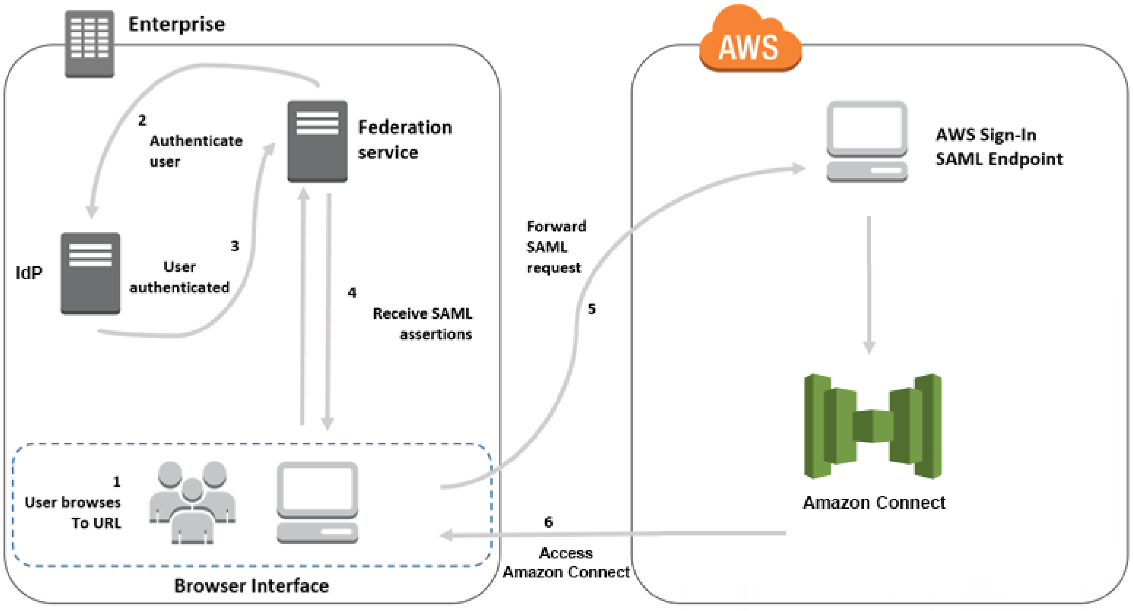
> **Important**
> To enable SAML authentication, you need to create an AWS Identity and Access Management (IAM) role that is used for federation between the identity provider on your existing network and Amazon Web Services. AWS Identity and Access Management is a web service that helps you securely control access to AWS resources. You use IAM to control who is authenticated (signed in) and authorized (has permissions) to use resources. In this case, the IAM role is used for federation between your identity provider and AWS. The permissions for the IAM role grant access to Amazon Connect.
> You cannot use the root credentials for your AWS as the account for SAML federation as it is not supported. Instead, follow the steps in the topic, and the topics linked to in the AWS Identity and Access Management documentation, to create an IAM role for federation. To learn more about IAM, see What is IAM?

Steps for configuring SAML include:

- Overview of Using SAML with Amazon Connect (p. 14)
- Enabling SAML-based Authentication for Amazon Connect (p. 15)
- Select SAML 2.0-based Authentication During Instance Creation (p. 16)
- Enable SAML Federation Between Your Identity Provider and AWS (p. 16)
- Use a Destination in Your Relay State URL (p. 18)
- Add users to Your Amazon Connect Instance (p. 18)
- SAML User Log in and Session Duration (p. 19)

Overview of Using SAML with Amazon Connect

The following diagram describes the flow for SAML requests to authenticate users and federate with Amazon Connect.



SAML requests go through the following steps:

1. The user browses to an internal portal that includes a link to log in to Amazon Connect. The link is defined in the identity provider.
2. The federation service requests authentication from the organization's identity store.
3. The identity store authenticates the user and returns the authentication response to the federation service.
4. When authentication is successful, the federation service posts the SAML assertion to the user's browser.
5. The user's browser posts the SAML assertion to the AWS Sign-In SAML endpoint (https://signin.aws.amazon.com/saml). AWS Sign-In receives the SAML request, processes the request, authenticates the user, and forwards the authentication token to the Amazon Connect service.
6. Using the authentication token from AWS, Amazon Connect authorizes the user and opens Amazon Connect in their browser.

Enabling SAML-based Authentication for Amazon Connect

The following steps are required to enable and configure SAML authentication for use with your Amazon Connect instance:

1. Create an Amazon Connect instance and select SAML 2.0-based authentication for identity management.
2. Enable SAML federation between your identity provider and AWS.
3. Add users to your Amazon Connect instance. Use the admin account created when you created your instance to log in and add users. The user names must exactly match the user name in your network directory and your identity provider.
4. Configure your identity provider for the SAML assertions, authentication response, and relay state. Users log in to your identity provider. When successful, they are redirected to your Amazon Connect

instance and then federated through an IAM role, which allows access to use the Amazon Connect console or CCP.

Select SAML 2.0-based Authentication During Instance Creation

When you are creating your Amazon Connect instance, select the SAML 2.0-based authentication option for identity management. On the second step, when you create an administrator user for the instance, the user name that you specify must exactly match a user name in your existing network directory. There is no option to specify a password for the admin user because passwords are managed through your existing directory. The admin user is created in Amazon Connect and assigned the **Admin** security profile.

You can log in to your Amazon Connect instance through your identity provider with the admin account specified to add additional users, assign security profiles, and manage configurations settings after you create your instance.

If you encounter an error and are unable to log in to your instance through your identity provider, you can log in as an administrator through the AWS Management Console to modify the admin user account.

Enable SAML Federation Between Your Identity Provider and AWS

To enable SAML-based authentication for Amazon Connect, you must create an identity provider in the IAM console. For more information, see Enabling SAML 2.0 Federated Users to Access the AWS Management Console.

The process to create an identity provider for AWS is the same for Amazon Connect, except that for step 7 in the flow diagram in the topic, the client is sent to your Amazon Connect instance instead of landing at the AWS Management Console.

The steps necessary to enable SAML federation with AWS include:

1. Create a SAML provider in AWS. For more information, see Creating SAML Identity Providers.
2. Create an IAM role for SAML 2.0 Federation with the AWS Management Console. You only need to create one IAM role for federation. The IAM role determines which permissions the users that log in through your identity provider have in AWS. In this case, the permissions are for accessing Amazon Connect. You can control the permissions to features of Amazon Connect by using security profiles in Amazon Connect. For more information, see Creating a Role for SAML 2.0 Federation (Console).

   In step 5, choose **Allow programmatic and AWS Management Console access**. In addition to the trust policy described in the topic in the procedure *To prepare to create a role for SAML 2.0 federation*, create a policy to assign permissions to your Amazon Connect instance. Permissions start on step 9 of the *To create a role for SAML-based federation* procedure .

   To create a policy for assigning permissions to the IAM role for SAML federation

   1. On the **Attach permissions policy** page, choose **Create policy**.
   2. On the **Create policy** page, choose **JSON**.
   3. Copy one of the following example policies and paste it into the JSON policy editor, replacing any existing text. You can use either policy to enable SAML federation, or customize them for your specific requirements.

      Use this policy to enable federation for all users in a specific Amazon Connect instance. For SAML-based authentication, replace the value for the `Resource` to the ARN for the instance that you created:

```
{
    "Version": "2012-10-17",
   "Statement": [
        {
            "Sid": "Statement1",
            "Effect": "Allow",
            "Action": "connect:GetFederationToken",
            "Resource": [
                "arn:aws:connect:us-east-1:361814831152:instance/2fb42df9-78a2-2e74-
d572-c8af67ed289b/user/${aws:userid}"
            ]
        }
    ]
}
```

Use this policy to enable federation to a specific Amazon Connect instances. Replace the value for the `connect:InstanceId` to the instance ID for your instance.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Statement2",
            "Effect": "Allow",
            "Action": "connect:GetFederationToken",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "connect:InstanceId": "2fb42df9-78a2-2e74-d572-c8af67ed289b"
                }
            }
        }
    ]
}
```

Use this policy to enable federation for multiple instances. Note the brackets around the listed instance IDs.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Statement2",
            "Effect": "Allow",
            "Action": "connect:GetFederationToken",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "connect:InstanceId": [
                    "2fb42df9-78a2-2e74-d572-c8af67ed289b",
                    "1234567-78a2-2e74-d572-c8af67ed289b"]
                }
            }
        }
    ]
}
```

4. After you create the policy, choose **Next:Review**, and then return to step 10 in the *To create a role for SAML-based federation* procedure in the Creating a Role for SAML 2.0 Federation (Console) topic.

3. Configure your network as a SAML provider for AWS. For more information, see Enabling SAML 2.0 Federated Users to Access the AWS Management Console.
4. Configure SAML Assertions for the Authentication Response. For more information, Configuring SAML Assertions for the Authentication Response.
5. Configure the relay state of your identity provider to point to your Amazon Connect instance. The URL to use for the relay state is comprised as follows:

   ```
   https://region-id.console.aws.amazon.com/connect/federate/instance-id
   ```

   Replace the *region-id* with the Region name where you created your Amazon Connect instance, such as us-east-1 for US East (N. Virginia). Replace the *instance-id* with the instance ID for your instance.

   > **Note**
   > You can find the instance ID for your instance by choosing the instance alias in the Amazon Connect console. The instance ID is the set of numbers and letters after '/instance' in the **Instance ARN** displayed on the **Overview** page. For example, the instance ID in the following Instance ARN is *178c75e4-b3de-4839-a6aa-e321ab3f3770*.
   > arn:aws:connect:us-east-1:450725743157:instance/*178c75e4-b3de-4839-a6aa-e321ab3f3770*

Use a Destination in Your Relay State URL

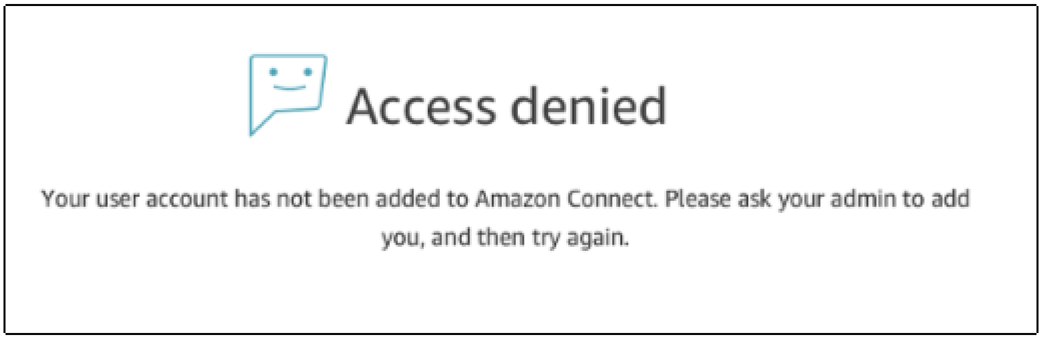
When you configure the relay state for your identity provider, you can use the ?destination argument in the URL to navigate users to a specific page in your Amazon Connect instance, such opening the CCP directly when an agent logs in, or displaying the real time metrics page when a call center manager logs in. The user must be assigned a security profile that grants access to that page in the instance. For example, if you want to send agents directly to the CCP when they log in, you can use a URL similar to the following for the relay state to go directly to the CCP when an agent logs in. You must use URL encoding for the destination value used in the URL:

```
https://us-east-1.console.aws.amazon.com/connect/federate/instance-id?
destination=%2Fconnect%2Fccp
```

Add users to Your Amazon Connect Instance

Add users to your connect instance, making sure that the user names exactly match the users names in your existing directory. If the names do not match, users can log in to the identity provider, but not to Amazon Connect because no user account with that user name exists in Amazon Connect. You can add users manually on the **User management** page, or you can bulk upload users with the CSV template. After you add the users to Amazon Connect, you can assign security profiles and other user settings.

When a user attempts to log in to Amazon Connect and successfully logs in to the identity provider, but no account with the same user name is found in Amazon Connect, the following **Access denied** message is displayed.



Access denied

Your user account has not been added to Amazon Connect. Please ask your admin to add you, and then try again.

Bulk upload users with the template.

You can also import all of your users by adding their information to a CSV file, and then using the import feature within Amazon Connect. If you plan to add users by uploading a CSV file, make sure that you use the template for SAML users, which you can find on the **User management** page in Amazon Connect. A different template is used for SAML-based authentication. If you previously downloaded the template, you should download the version available on the **User management** page after you set up your instance with SAML-based authentication. The template should not include a column for email or password.

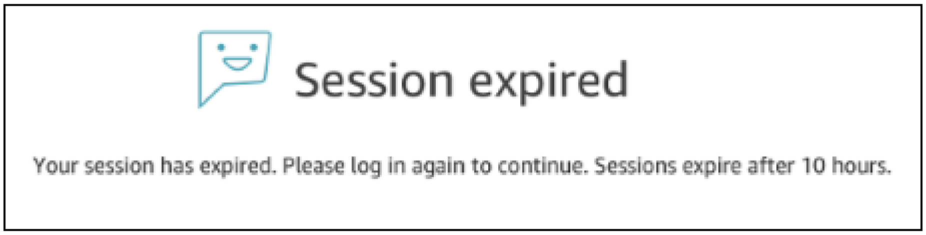
SAML User Log in and Session Duration

When you are using SAML for identity management in Amazon Connect, users must log in to Amazon Connect by first logging in to the identity provider you have configured in your network to use with Amazon Web Services. After authentication by the identity provider, a token for their session is created, and the user is redirected to your Amazon Connect instance and automatically logged in to Amazon Connect using single sign-on.

As a best practice, you should also define a process for your Amazon Connect users to log out when they are finished using Amazon Connect. They should log out from both Amazon Connect and your identity provider. If they do not, the next person that logs in to the same computer can log in to Amazon Connect without a password since the token for the previous sessions is still valid for the duration of the session, by default, 10 hours.

About Session Expiration

Amazon Connect sessions expire 10 hours after a user logs in. After 10 hours, users are automatically logged out, even if they are currently on a call. If you plan to have agents stay logged in for more than 10 hours, you should consider having agents log out of Amazon Connect and your identity provider, and then log in again through your identity provider before the session expires. This resets the session timer set on the token so that agents are not logged out during an active contact with a customer. When a session expires while a user is logged in, the following message is displayed. To use Amazon Connect again, the user needs to log in to your identity provider.



Session expired

Your session has expired. Please log in again to continue. Sessions expire after 10 hours.

Port Your Current Phone Number

To continue to use your current United States phone number with Amazon Connect, you can submit a support ticket to port the number to Amazon Connect. The Amazon Connect team processes your request and assists you with the number porting process.

Porting phone numbers typically takes between two to four weeks after you submit the required information. The amount of time depends on the complexity of the request and your current carrier. Porting toll-free numbers, or requests to port a large quantity of numbers at one time, usually take longer than porting local, direct dial numbers.

We recommend that you select a phone number for Amazon Connect so that you can become familiar with the service while waiting for your number to be ported.

To port your current phone number to Amazon Connect

1. Open the Amazon Connect console at https://console.aws.amazon.com/connect/.
2. Log in with the account used to create the Amazon Connect instance to which to port your current number.
3. Choose **Support**, **Support Center**.
4. On the **Support Center** page, choose **Create Case**.
5. Fill in values for the following fields:
   - For **Regarding**, choose **Service Limit Increase**.
   - For **Limit Type**, choose **Connect**.
   - For **Region**, select the Region in which you created your Amazon Connect instance.
   - For **Limit**, choose **Phone Number Porting**.
   - For **New limit value**, enter the number of phone numbers to port.
   - For **Use Case Description**, include as much information as possible about your request, including whether the numbers are Direct Inward Dial or toll-free, your current carrier, and the contact information for the person authorized to make changes to your current phone service. If you do not know all of these details, you may leave information out.
6. Fill in the rest of the form, and choose **Submit**.

About Porting Phone Numbers

When you port your current phone number into Amazon Connect, we provide any possible assistance. However, many of the steps are performed by telecommunications carriers.

We collect the information necessary to verify that you are authorized to port the numbers that you request. We pass that information on to your existing carrier, and coordinate with the new carrier to get your number ported. Each carrier has their own process and requirements for number porting. Your number cannot be ported until your current carrier verifies that you own and are authorized to port the numbers requested. Your current carrier must approve the request to port your number before the new carrier can provision the number. After that is complete, the Amazon Connect team can start configuring your Amazon Connect instance to use the ported numbers.

The steps in the porting process are as follows:

1. Submit a support ticket to port your number.
2. Confirm number portability. The Amazon Connect team confirms whether the numbers that you request can be ported from your current carrier. We then contact you with next steps, or notify you that the requested numbers cannot be ported.
3. Complete the Letter of Authorization/Agency (LOA). When you complete the LOA form, the information you provide must match the information on file with your current carrier. If the information does not match, it may delay the porting of your number. The LOA form authorizes your current carrier to release your number and allow it to be ported. If your number can be ported, we provide you with an LOA form appropriate for the type of number to port. There are different forms for local, Direct Inward Dial (DID), and toll-free numbers. If you are porting multiple numbers from different carriers, fill out a separate form for each carrier.

   On the LOA form, include the following:

   - The numbers to port
   - Information about your current carrier, such as a phone bill
   - Contact information for the person authorized to make changes to your phone service

4. To get the port started, the Amazon Connect team submits the LOA to the carrier for Amazon Connect on your behalf. The new carrier works with your current carrier to move your current number over to their service. This step typically takes 3–5 business days.

   If your current carrier is able to validate and approve your request, they provide a date for the number to be ported to Amazon Connect.

   If your current carrier rejects the request to port your number due to the LOA not having correct or complete information, the Amazon Connect team contacts you and requests a new LOA to submit to the carrier.

When we receive a date from your current carrier, we start adding the numbers to your Amazon Connect instance about a day before the scheduled date.

Integrate with Your CRM

You can integrate Amazon Connect with the Salesforce and Zendesk CRMs. Integration allows you to launch your contact center in your CRM of choice, maintain your existing user base, and use the Amazon Connect cloud-based infrastructure.

To integrate the Contact Control Panel (CCP) into your CRM, see Amazon Connect Contact Streams. When completed, add the origin URLs to your instance settings. This enables communication between Amazon Connect and your CRM. For more information, see Application Integration (p. 24).

Remove Your Amazon Connect Instance

If you no longer want to use an Amazon Connect instance, you can delete it. If you delete an instance, the phone number claimed for the instance is released. You lose all settings, data, metrics, and reports associated with the instance.

> **Important**
> You cannot undo the deletion of an instance or restore settings or data from the instance after it is deleted.

To delete an Amazon Connect instance

1. Open the Amazon Connect console at https://console.aws.amazon.com/connect/.
2. Select the check box for the instance and choose **Remove**.
3. When prompted, type the name of the instance and choose **Remove**.

Configuring Your Amazon Connect Instance

You can configure your Amazon Connect instance using the AWS Management Console. To access instance settings, choose the name of the instance in the **Instance Alias** column.

Settings

Overview

The **Overview** section displays the following information about your Amazon Connect instance.

- **Instance ARN**—the ARN for the instance. The instance ID for the instance is included in the ARN, and is the value after the instance/. For example, the instance ID in the following instance ARN is df9e742b-310b-4eb2-a062-31bc99177ed4.

  ```
  arn:aws:connect:us-east-1:361814831152:instance/df9e742b-310b-4eb2-
  a062-31bc99177ed4
  ```
- **Directory**—The instance alias for the instance.
- **Login URL**—The URL to use in a browser to log in directly to the contact center for your instance.

  If your agents (users that are assigned only the Agent security profile) try to use this URL to log in to Amazon Connect, "Error 403! (Forbidden) is displayed on the page. The agent can still open the Contact Control Panel (CCP) by selecting the phone icon in the top-right corner of the page.

You can use the **Login as administrator** button to log in to the instance using your AWS account with full admin permissions. This can be helpful if you ever forgot the password for the admin account, or need to update Amazon Connect settings.

Telephony

Select whether to accept incoming calls to, or allow outbound calls from, your Amazon Connect instance. You can use security profiles to set permissions to enable or disable outbound calling.

Data Storage

Data, such as call recordings and reports, is stored securely in an Amazon S3 bucket. During setup, a default Amazon S3 bucket is created and encrypted using AWS Key Management Service. This bucket and key are used for both calling recordings and reports. Alternatively, you can use separate buckets and keys for call recordings and reports.

Call recordings in Amazon Connect are stored as .wav files, and played back in 8 Khz pulse-code modulation (PCM) format.

Before updating the data storage settings, ensure that you are familiar with Amazon S3 and AWS KMS.

To update data storage settings

1. Open the Amazon Connect console at https://console.aws.amazon.com/connect/.
2. Choose the name of the instance from **Instance Alias**.
3. In the navigation pane, choose **Data storage**.
4. To update the settings for call recordings, do the following:

    a. For **Call recordings**, choose **Edit**.

    b. (Optional) To disable call recordings, clear **Enable call recording**.

    c. (Optional) If call recordings are enabled, you can create a new S3 bucket or select an S3 bucket that you've already created.

    d. (Optional) If call recordings are enabled, you can update the encryption settings as needed. To disable encryption, clear **Enable encryption**. To update the KMS key, specify a key from the same region as your S3 bucket.

    e. To save your changes, choose **Save**.
5. To update the settings for exported reports, do the following:

    a. For **Exported reports**, choose **Edit**.

    b. (Optional) To disable exported reports, clear **Enable exported reports**.

    c. (Optional) If exported reports are enabled, you can create a new S3 bucket or select an S3 bucket that you've already created.

    d. (Optional) If exported reports are enabled, you can update the encryption settings as needed. To disable encryption, clear **Enable encryption**. To update the KMS key, specify a key from the same region as your S3 bucket.

    e. To save your changes, choose **Save**.

Data Streaming

You can export contact trace records (CTRs) and agent events from Amazon Connect and perform real-time analysis on contacts. Data streaming uses the Amazon Kinesis platform to support data streaming.

To set up data streaming

1. Open the Amazon Connect console at https://console.aws.amazon.com/connect/.
2. Choose the name of the instance from **Instance Alias**.
3. In the navigation pane, choose **Data streaming**.
4. Choose **Enable data streaming**.
5. Select **Kinesis** or **Kinesis Data Firehose**, and then do one of the following:
    - To use an existing Amazon Kinesis stream or Kinesis Data Firehose, select the resource in the drop-down list.
    - To create a new resource, choose **Create a new Amazon Kinesis stream** (or Kinesis Data Firehose).

      This opens the Amazon Kinesis console where you can create the stream or firehose to use with Amazon Connect. Wait until the stream or firehose is created, then return to the Amazon Connect console.

Reload the page so that the stream or firehose you created is displayed in the resource selection, then select the stream or firehose.

Note
If you enable server-side encryption for the Kinesis stream you select, Amazon Connect cannot publish to the stream because it does not have permission to Kinesis kms:GenerateDataKey. To work-around this, enable encryption for call recordings or scheduled reports, create a customer master key (CMK) using KMS to use for encryption, and then choose the same CMK for your Kinesis data stream that you use for call recording or scheduled reports encryption so that Amazon Connect has appropriate permissions to encrypt data sent to Kinesis. To learn more about creating a customer master key (CMK) KMS key, see Creating Keys.

6. Choose **Save**.

Application Integration

All domains that embed the CCP for a particular instance must be explicitly whitelisted for cross-domain access to the instance. For example, to integrate with Salesforce, you must whitelist your Salesforce Visualforce domain.

To whitelist a domain URL

1. Open the Amazon Connect console at https://console.aws.amazon.com/connect/.
2. Choose the name of the instance from **Instance Alias**.
3. In the navigation pane, choose **Application integration**.
4. Choose **Add origin**.
5. Type the URL and choose **Add**.

Contact Flows

A contact flow defines the customer experience with the contact center from start to end. You can configure your contact flow using the AWS Management Console as follows.

Security Keys

Amazon Connect can encrypt sensitive data collected by contact flows using public-key cryptography. Provide an X.509 certificate within your contact flow to encrypt data captured using the stored customer input system attribute. You must upload a signing key in `.pem` format in order to use this feature. The signing key is used to verify the signature of the certificate used within the contact flow.

Note
You can have up to two signing keys active at one time to facilitate rotation.

Data that is encrypted within a contact flow is made available through the stored customer input system attribute. The AWS Encryption SDK can be used to decrypt this data within your system. For more information, see the AWS Encryption SDK Developer Guide.

To add a security key

1. Open the Amazon Connect console at https://console.aws.amazon.com/connect/.
2. Choose the name of the instance from the **Instance Alias** column.
3. In the navigation pane, choose **Contact flows**.

4. Choose **Add key**.
5. Paste the contents of your public key in **Public key contents** and choose **Add**.

Add an Amazon Lex bot to Your Instance

With Amazon Lex, you can build conversational interactions (bots) that feel natural to your customers, giving you access to the same speech recognition and natural language understanding technology that powers Alexa. After you create an Amazon Lex bot, you can add it to your instance and then integrate it into your contact flows. You can add bots from the same region as your Amazon Connect instance, or from a different region.

To add an Amazon Lex bot

1. Open the Amazon Connect console at https://console.aws.amazon.com/connect/.
2. Choose the name of the instance from the **Instance Alias** column.
3. In the navigation pane, choose **Contact flows**.
4. In the **Region** drop-down list, choose the Region in which you created your Amazon Lex bot.

   If there are bots associated with your AWS account in the chosen region, the bots are displayed in the **Bot** drop-down list. If no bots are found in the Region, or when there are no additional bots to add from that Region, the drop-down menu is disabled. A message indicates that there are no bots available to choose in that Region.
5. In the **Bots** drop-down menu, choose your bot, then choose **Add bot**.

To create a new bot, **Create a new Lex bot** to open the Amazon Lex console. You may need to select a region where Amazon Lex is available.

To remove a bot from your instance, choose **Remove** next to the bot to remove.

Contact flow logs

Select the **Enable Contact flow logs** check box to start sending your contact flow logs to Amazon CloudWatch. To learn more about Contact flow logs, see Contact Flow Logs.

Using Service-Linked Roles for Amazon Connect

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

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

For information about other services that support service-linked roles, see AWS Services That Work with IAM and look for the services that have **Yes** in the **Service-Linked Role** column. Choose a **Yes** with a link to view the service-linked role documentation for that service.

Service-Linked Role Permissions for Amazon Connect

Amazon Connect uses the service-linked role named **AWSServiceRoleForAmazonConnect_** – Grants Amazon Connect permission to access AWS resources on your behalf.

The AWSServiceRoleForAmazonConnect_ service-linked role trusts the following services to assume the role:

- `connect.amazonaws.com`

The role permissions policy allows Amazon Connect to complete the following actions on the specified resources. As you enable additional features in Amazon Connect, additional permissions are added for the service-link role to access the resources associated with those features:

- Action: all Amazon Connect actions, `connect:*`, on all Amazon Connect resources.
- Action: Amazon S3 `s3:GetObject`, `s3:GetObjectAcl`, `s3:PutObject`, `s3:PutObjectAcl`, `s3:DeleteObject`, `s3:GetBucketLocation`, and `GetBucketAcl` for the S3 bucket specified for call recordings.

  It also grants `s3:PutObject`, `s3:PutObjectAcl`, and `s3:GetObjectAcl` to the bucket specified for exported reports.
- Action: Amazon Kinesis Data Firehose `firehose:DescribeDeliveryStream` and `firehose:PutRecord`, and `firehose:PutRecordBatch` for the delivery stream defined for Agent event streams and CTRs.
- Action: Amazon Kinesis Data Streams `kinesis:PutRecord`, `kinesis:PutRecords`, and `kinesis:DescribeStream` for the stream specified for Agent event streams and CTRs.
- Action: Amazon Lex `lex:PostContent` for the bots added to your instance.
- Action: Amazon CloudWatch Logs `logs:CreateLogStream`, `logs:DescribeLogStreams`, and `logs:PutLogEvents` to the CloudWatch Logs group specified for contact flow logging.

You must configure permissions to allow an IAM entity (such as a user, group, or role) to create, edit, or delete a service-linked role. For more information, see Service-Linked Role Permissions in the *IAM User Guide*.

Creating a Service-Linked Role for Amazon Connect

You don't need to manually create a service-linked role. When you create a new instance in Amazon Connect in the AWS Management Console, Amazon Connect creates the service-linked role for you.

If you delete this service-linke role, and then need to create it again, you can use the same process to recreate the role in your account. When you create a new instance in Amazon Connect, Amazon Connect creates the service-linked role for you again.

You can also use the IAM console to create a service-linked role with the **Amazon Connect - Full access** use case. In the IAM CLI or the IAM API, create a service-linked role with the `connect.amazonaws.com` service name. For more information, see Creating a Service-Linked Role in the *IAM User Guide*. If you delete this service-linked role, you can use this same process to create the role again.

Editing a Service-Linked Role for Amazon Connect

Amazon Connect does not allow you to edit the AWSServiceRoleForAmazonConnect_ service-linked role. After you create a service-linked role, you cannot change the name of the role because various entities might reference the role. However, you can edit the description of the role using IAM. For more information, see Editing a Service-Linked Role in the *IAM User Guide*.

Deleting a Service-Linked Role for Amazon Connect

You don't need to manually delete the AWSServiceRoleForAmazonConnect_ role. When you delete your Amazon Connect instance in the AWS Management Console, Amazon Connect cleans up the resources and deletes the service-linked role for you.

Supported Regions for Amazon Connect Service-Linked Roles

Amazon Connect supports using service-linked roles in all of the regions where the service is available. For more information, see AWS Regions and Endpoints.

Monitoring Amazon Connect in Amazon CloudWatch Metrics

Amazon Connect sends data about your instance to CloudWatch metrics so that you can collect, view, and analyze CloudWatch metrics for your Amazon Connect virtual contact center. You can use this data to monitor key operational metrics and set up alarms. Data about your contact center is sent to CloudWatch every 1 minute.

When you view the CloudWatch metrics dashboard, you can specify the refresh interval for the data displayed. The values displayed in the dashboard reflect the values for the refresh interval you define. For example, if you set the refresh interval to 1 minute, the values displayed are for a minute period. You can select a refresh interval of 10 seconds, but Amazon Connect does not send data more often than every 1 minute. Metrics that are sent to CloudWatch are available for two weeks, and then discarded. To learn more about metrics in CloudWatch, see What is Amazon CloudWatch?

Amazon Connect Metrics Sent to CloudWatch

The following Amazon Connect metrics are sent to CloudWatch:

CallsBreachingConcurrencyQuota

The number of voice calls that exceeded the concurrent active calls limit for the instance. This is a count of the number of calls that exceeded the limit, not the number of concurrent calls in excess of the limit.

Unit: Count

CallBackNotDialableNumber

The number of times a queued call back to a customer could not be dialed because the customer's number is in a country for which outbound calls are not allowed for the instance. The countries allowed for an instance are defined by the service limits.

Unit: Count

CallRecordingUploadError

The number of call recordings that failed to upload to the Amazon S3 bucket configured for your instance. This is the bucket specified in **Data Storage** > **Call Recordings** settings for the instance.

Unit: Count

CallsPerInterval

The number of voice calls, both inbound and outbound, received or placed per second in the instance.

Unit: Count

ConcurrentCalls

The number of concurrent active voice calls in the instance at the time the data is displayed in the dashboard. The value displayed for this metric is the number of concurrent active calls at the time the dashboard is displayed, and not a sum for the entire interval of the refresh interval set. All active voice calls are included, not only active calls that are connected to agents.

Unit: Count

ConcurrentCallsPercentage

The percentage of the concurrent active voice calls service limit used in the instance. This is calculated by `ConcurrentCalls/ConfiguredConcurrentCallsLimit * 100`.

Unit: Percent

ContactFlowErrors

The number of times the error branch for a contact flow was executed.

Unit: Count

ContactFlowFatalErrors

The number of times a contact flow failed to execute due to a system error.

Unit: Count

LongestQueueWaitTime

The longest amount of time, in seconds, that a contact waited in a queue. This is the length of time a contact waited in a queue during the refresh interval selected in the CloudWatch dashboard, such as 1 minute or 5 minutes.

Unit: Seconds

MissedCalls

The number of voice calls that were missed by agents during the refresh interval selected, such as 1 minute or 5 minutes. A missed call is one that is not answered by an agent within 20 seconds.

Unit: Seconds

MisconfiguredPhoneNumbers

The number of calls that failed because the phone number is not associated with a contact flow.

Unit: Count

PublicSigningKeyUsage

The number of times a contact flow security key (public signing key) was used to encrypt customer input in a contact flow.

Unit: Count

QueueCapacityExceededError

The number of calls that were rejected because the queue was full.

Unit: Count

QueueSize>

The number of contacts in the queue. The value reflects the number of contacts in the queue at the time the dashboard is accessed, not for the duration of the reporting interval.

Unit: Count

ThrottledCalls

The number of voice calls that were throttled by the Amazon Connect service because the rate of calls per second (Callrate) exceeded the configured limit for the instance.

Unit: Count

ToInstancePacketLossRate

The ratio of packet loss for calls in the instance, reported every 10 seconds. Each data point is between 0 and 1, which represents the ratio of packets lost for the instance.

Unit: Percent

Amazon Connect CloudWatch Metrics Dimensions

In CloudWatch, a dimension is a name/value pair that uniquely identifies a metric. In the dashboard, metrics are grouped under dimensions. The following dimensions are used in the CloudWatch dashboard for Amazon Connect metrics. When you view metrics, only metrics for which there is data are displayed in the dashboard. If there is no activity during the refresh interval for which there is a metric, then no data from your instance is displayed in the dashboard. The following dimensions are used for Amazon Connect metrics in CloudWatch.

Instance ID, Participant, Stream Type, Type of Connection

This dimension contains metrics about connections to your instance, and includes:

- ToInstancePacketLossRate

Contact Flow Metrics Dimension

This dimension contains metrics about contact flows in your instance, and includes:

- CallRecordingUploadError
- ContactFlowErrors
- ContactFlowFatalErrors
- MisconfiguredPhoneNumbers
- PublicSigningKeyUsage

Queue Metrics Dimension

This dimension contains metrics about queues in your instance, and includes:

- CallBackNotDialableNumber
- LongestQueueWaitTime
- QueueCapacityExceededError
- QueueSize

Instance metrics Dimension

This dimension contains metrics about voice calls and call recordings in your instance, and includes:

- CallsBreachingConcurrencyQuota
- CallsPerInterval

- ConcurrentCalls
- ConcurrentCallsPercentage
- MissedCalls
- ThrottledCalls

Using AWS Lambda Functions with Amazon Connect

Amazon Connect can interact with your own systems and take different paths in contact flows dynamically. To achieve this, invoke Lambda functions, fetch results in an contact flow, and call your own services or interact with other AWS data stores or services.

To learn more about AWS Lambda, see the AWS Lambda Developer Guide.

Invoking a Lambda Function from a Contact Flow

The steps required to invoke a Lambda function from Amazon Connect include the following:

1. Create a Lambda function and define its trigger policy to allow Amazon Connect to invoke the function.
2. Use the ARN of the Lambda function in an **Invoke AWS Lambda function** block in your contact flow.
3. Configure the Lambda function code to parse the JSON event sent from the contact flow, and define the business logic to execute.
4. Test the configuration to confirm that the Lambda function returns the correct JSON response.
5. Consume the attribute values returned from Lambda to use in your contact flow.

Create a Lambda Function and Configure a Trigger Policy

Amazon Connect can successfully invoke a Lambda function in an AWS account when a resource policy has been set on the Lambda function. For more information, see Using Resource-Based Policies for AWS Lambda in the *AWS Lambda Developer Guide*.

To begin, create a Lambda function, and then note down the function name. For more information about creating a Lambda function, see Create a Simple Lambda Function.

Use the following add-permission command to create a resource policy using this information:

```
aws lambda add-permission --function-name function:my-lambda-function --statement-id 1 \
     --principal connect.amazonaws.com --action lambda:InvokeFunction --source-
account 123456789012 \
     --source-arn arn:aws:connect:us-east-1:123456789012:instance/def1a4fc-ac9d-11e6-
b582-06a0be38cccf
```

This command uses the following input:

- The name of the Lambda function (for example, `my-lambda-function`)
- The ARN of a Amazon Connect instance (for example, `arn:aws:connect:us-east-1:123456789012:instance/def1a4fc-ac9d-11e6-b582-example`)

To find the ARN for your instance, open the Amazon Connect console, and then choose the **Instance Alias** to open the **Overview** page.

- The AWS account ID for the Amazon Connect instance (for example, **123456789012**).

Invoke the Lambda Function in Your Contact Flow

To invoke a Lambda function from your contact flow, add an **Invoke AWS Lambda function** block to the flow, and then add the ARN for the function you created as the value for the **Function ARN** in the contact flow properties. You can view the ARN for the function in the AWS Lambda console at https://console.aws.amazon.com/lambda/.

You can also run the following command in the AWS Command Line Interface to view the function ARN:

```
aws lambda get-function --function-name my-lambda-function
```

In the **Invoke AWS Lambda function** block, you can add **Function input parameters**, which are key-value pairs that are sent to the Lambda function when invoked. You can also specify a **Timeout** value for the function.

On every Lambda function invocation from a contact flow, you pass a default set of information related to ongoing contact, as well as any additional attributes defined in the **Function input parameters** for the **Invoke AWS Lambda function** block added to your contact flow.

The following is an example JSON request to a Lambda function:

```
{
    "Details": {
        "ContactData": {
            "Attributes": {},
            "Channel": "VOICE",
            "ContactId": "4a573372-1f28-4e26-b97b-XXXXXXXXXXX",
            "CustomerEndpoint": {
                "Address": "+1234567890",
                "Type": "TELEPHONE_NUMBER"
            },
            "InitialContactId": "4a573372-1f28-4e26-b97b-XXXXXXXXXXX",
            "InitiationMethod": "INBOUND | OUTBOUND | TRANSFER | CALLBACK",
            "InstanceARN": "arn:aws:connect:aws-region:1234567890:instance/
c8c0e68d-2200-4265-82c0-XXXXXXXXXX",
            "PreviousContactId": "4a573372-1f28-4e26-b97b-XXXXXXXXXX",
            "Queue": "QueueName",
            "SystemEndpoint": {
                "Address": "+1234567890",
                "Type": "TELEPHONE_NUMBER"
            }
        },
        "Parameters": {
            "sentAttributeKey": "sentAttributeValue"
        }
    },
    "Name": "ContactFlowEvent"
}
```

The request is divided into three parts:

- Contact data—This is always passed by Amazon Connect for every contact. Some parameters are optional.

- User attributes—These are attributes that have been previously associated with a contact, such as when using a **Set contact attributes** block in a contact flow. This map may be empty if there aren't any saved attributes.
- Parameters—These are parameters specific to this call that were defined when you created the Lambda function.

The Lambda function response should be a simple Map *`String String`*. This map can be up to 32k. If you fail to reach Lambda, the function throws an exception, the response is not understood, or the Lambda function takes more time than the limit, the contact flow jumps to the `Error` label. The following code is an example Python Lambda function:

Configure Your Lambda Function

To successfully pass attributes between your Lambda function and Amazon Connect, configure your function to correctly parse the JSON request sent from the **Invoke AWS Lambda function** block, and define any business logic that should be applied. How the JSON is parsed depends on the runtime you use for your function. For example, the following example shows how to access the sentAttributeKey using sing Node.JS:

```
var receivedAttribute = event['Details']['Parameters']['sentAttributeKey'];
```

Verify the Function Response

Test the output returned from your Lambda function to confirm that it will be correctly consumed when returned to Amazon Connect. The following example shows a sample response in Node.JS:

```
exports.handler = function(event, context, callback) {

var resultMap = {
    Name:'CustomerName',
    Address:'1234 Main Road',
    CallerType:'Patient'
}

callback(null, resultMap);
}
```

And this example shows an example response using Python:

```
def lambda_handler(event, context):
resultMap = {"Name":"CustomerName","Address":"1234 Main Road","CallerType":"Patient"};
return resultMap;
```

The output returned from the function must be a flat object of key/value pairs, with values that include only alphanumeric, dash, and underscore characters. Nested and complex objects are not supported. The size of the returned data must be less than 32 Kb of UTF-8 data.

The following example shows the JSON output from these Lambda functions:

```
{
    "Name": "CustomerName",
    "Address": "1234 Main Road",
    "CallerType": "Patient"
}
```

Using the Lambda Function Response

There are two ways to use the function response in your contact flow. You can either directly reference the variables returned from Lambda, or store the values returned from the function as contact attributes and then reference the stored attributes. When you use an external reference to a response from a Lambda function, the reference will always receive the response form the most recently invoked function. To use the response from a function before a subsequent function is invoked, the response must be saved as a contact attribute, or passed as a parameter to the next function.

Access Lambda attributes directly

If you access the variables directly, you can use them in contact flow blocks, but they are not included in contact trace records (CTR). To access these variables directly in a contact flow block, add the block after the **Invoke AWS Lambda function** block, and then reference the attributes as shown in the following example:

```
Name - $.External.Name
Address - $.External.Address
CallerType - $.External.CallerType
```

Make sure that the name specified for the source attribute matches the key name returned from Lambda.

Store Lambda variables as contact attributes

If you store the variables as contact attributes, you can use them throughout your contact flow, and they are included in CTRs.

To store the values returned as contact attributes and then reference them, use a **Set contact attributes** block in your contact flow after the **Invoke AWS Lambda function** block. Choose **External** for the **Type**. Following the example we're using, set **Destination key** to `returnedContactName`, and set the **Source attribute** to `Name`

Add Address as a **Source attribute** and use `returnedContactAddress` as the **Destination key**. Then add callerType as a **Source attribute** and use `returnedContactType` for the **Destination key**.

Make sure that the name specified for the source attribute matches the key name returned from Lambda.

Amazon Connect and Salesforce Integration

The core functionality of the Amazon Connect CTI Adapter provides a WebRTC browser-based Contact Control Panel (CCP) within Salesforce. The Amazon Connect CTI integration consists of two components, a managed Salesforce package and a AWS Serverless application (need link) deployed to your AWS environment.

With those components, customers can build a deep integration between the Amazon Connect contact center platform and Salesforce, the leading customer relationship management (CRM) platform. The collection of pre-build utilities enables a rapid integration between these two platforms. The AWS Serverless application package contains a set of common Lambda functions to be used by Amazon Connect to interact with Salesforce.

About the Adapter

The key benefits of the adapter include:

- Agent state synchronization between Salesforce Omni and Amazon Connect
- Provide valuable information to the agent through configurable view of call attributes
- Utilize the Amazon Connect Call Campaign Object for automated outbound dialling
- Automatically create phone call tasks and relate it to the right Salesforce object
- Embed Amazon Connect Call Recordings in the Salesforce record
- Automatically clean-up open tabs to improve agent efficiency
- Easily enable lookup, create and update operations for different Salesforce objects, like Contacts and Cases, within Amazon Connect contact flows.
- Support Salesforce Sales and Service Console in Classic and Lightning.

We recommend that you initially install the package into your Salesforce sandbox. After the package is installed, you can configure your Salesforce Call Center configuration within Salesforce.

The next step is to whitelist your Salesforce Visualforce domain within your Amazon Connect Application integration. This allows cross-domain access to your Amazon Connect instance.

This page provides a quick setup guide. Please review the Amazon Connect CTI Adapter v2 for Salesforce installation guide for a more detailed walk-through and setup of the full CTI Adapter capabilities. We also have a trailhead available at https://sfdc.co/Amazon-Connect. Note, it's still in process of being updated to support latest CTI Adapter features.

Prerequisites

Before the Amazon Connect CTI package can be installed, the following prerequisites need to be fulfilled:

- Salesforce Classic, Salesforce Console, or Lightning Experience
- Create an Amazon Connect instance (https://aws.amazon.com/connect/).

- Salesforce Omni-Channel must be activated in the Salesforce org. For more information, see Enable Omni-Channel.

Browser Compatibility

Amazon Connect requires WebRTC to enable soft-phone voice media stream and Websockets to enable soft-phone signalling. Consequently, users are required to use the latest version of either Google Chrome or Mozilla Firefox. For more details, please see the Amazon Connect FAQ page.

To integrate with Salesforce

1. In your Salesforce sandbox, install the following managed package: Amazon Connect CTI Adapter.
2. Edit one of appropriate call center configuration (Amazon Connect CCP Adapter Classic, Console, or Lightning).

   - For Amazon Connect CCP URL, type the CCP URL for your instance (for example, https://instance.awsapps.com/connect/ccp).
   - For Phone Number Formatting, Country, specify the appropriate 2-digit ISO country code.
   - To provide Salesforce users with access to the Amazon Connect CCP, on the Setup Call Centers page, choose Manage Call Center Users. Add the Salesforce users you want to enable for using these call features. Be sure to add your own Salesforce user account if you plan to these features.
3. Whitelist your Salesforce Visualforce domain URL using the directions in Application Integration. To verify the URL, open the Visualforce page in setup. This URL usually has the following format:

   https://amazonconnect.`your-instance-name`.visual.force.com
4. Log in to your Amazon Connect instance.
5. Launch Salesforce. You should see the integrated CCP in the side panel (Salesforce Classic) or the phone toolbar (Salesforce Classic and Lightning Experience).

Troubleshooting Common Issues

If you encounter errors with your configuration, check the following common issues:

- Confirm that Salesforce is not blocking your iFrame. For more information, see Enable Clickjack Protection for Visualforce Pages Even When Headers Are Disabled.
- Confirm that the Amazon Connect user is assigned only the Agent security profile.
- Confirm that your Salesforce Call Center **Phone Number Formatting** is configured with the following parameters:

  {"OPF":"0","NPF”:"*2 digit dialing code*","Country”:"*2 digit country code*","NF":"International_plaintext","TNF":"(555) 123-4567"}
- Confirm that the Salesforce user can access the call center. To check a user's status, choose **Manage Call Center Users**.
- Under **Softphone Layout**, **Screen Pop**, confirm that **Single-matching record** is set to **Pop detail page** and **Multiple-matching record** is set to **Pop to search page**.
- If you are using Salesforce Lightning Experience and do not see a phone toolbar icon, confirm that you have enabled console navigation. To enable console navigation, in the **Salesforce Setup Console**, choose **App Manager**, **Service Console (Lightning)**, **Edit**. On the **Edit** page, choose **App Options**, **App Navigation**, **Console Navigation**.

Amazon Connect Troubleshooting and Best Practices

Use this guide to identify best practices for using Amazon Connect. Also, to troubleshoot information when something isn't working quite how it should.

Sections in this guide:

Best Practice for Using the Contact Control Panel

This guide provides information about the CCP soft phone, including best practices and troubleshooting. For workstations unable to meet soft phone connectivity requirements or experiencing soft phone issues, the CCP also features the ability to redirect to an external device.

Topics in this section:

Agent Workstation Requirements

Agent workstations in the contact center vary widely. While the Amazon Connect CCP is built to handle high levels of jitter and high latency environments, the architecture of the workstations that agents use, and the location and environment in which they take calls, can impact the quality of experience.

Under-powered workstations can make it difficult for agents to access the tools and resources they need to service callers. Also, keep in mind the resource requirements when scoping workstations to ensure that they can perform under load while appropriately multitasking for the use case. For the best agent and customer audio experience, a USB headset is recommended. Alternatively, you can redirect the call to an external number, in E.164 format, using an agent's existing telephony.

The following values are the minimum system requirements for the workstations using the CCP only. Additional memory, bandwidth, and CPU should be scoped for the operating system and anything else running on the workstation to avoid resource contention.

- **Browser**—The latest three versions of Google Chrome or Mozilla Firefox
- **Network**—100 Kbps bandwidth per connected workstation
- **Memory**—2 GB RAM
- **Processor (CPU)**—2 GHz

Monitoring Workstations

There are many factors that can affect CCP functionality at the workstation level. Access to various levels of logging information is essential in determining steps towards remediation. Adding additional logging and monitoring to workstations that are experiencing resource contention may further reduce available resources and invalidate test results. We recommended that your workstation meet the minimum requirements outlined in the Agent Workstation Requirements (p. 38) section of this guide, leaving additional resources available for logging, monitoring, malware scanning, operating system functions, and any other running processes.

Collect additional historical logging and data sources for correlation. If you see a correlation between the time of the event and the time the issue was reported, you may be able to determine the root cause with the following information:

- Round trip time (RTT) and packet loss to endpoints located within your Amazon Connect Region from your agent workstation, or an identical workstation on the same network segment. If no Region endpoints are available because of security policies, any public WAN endpoint suffices, for example, www.Amazon.com. Ideally, use your instance alias address (https://yourInstanceName.awsapps.com), and also your signaling address for endpoints.
- Regular monitoring of workstations that show processes running, and the current resource usage of each process.
- Workstation performance/utilization in these areas:
  - Processor (CPU)
  - Disk / drive
  - RAM / memory
  - Network throughput and performance
- Monitor all of the preceding for your VDI desktop environment, including RTT/packet monitoring between the agent workstation and the VDI environment.

Network Ports and Protocols

The CCP soft phone requires three connections to AWS resources. You must open the address and port to these resources with the appropriate protocol for the Region in which you created your Amazon Connect instance to allow bi-directional communication for full functionality of the CCP. The CCP needs access to the IP ranges for Amazon Elastic Compute Cloud (Amazon EC2) , Amazon CloudFront, and Amazon Connect, which are listed in the https://ip-ranges.amazonaws.com/ip-ranges.json file under Amazon EC2 (EC2), CloudFront (CLOUDFRONT), and Amazon Connect (AMAZON_CONNECT) respectively. The address ranges in the file are updated as new resources are added. This means that you need to monitor the included ranges and update your environment accordingly to ensure that agents can use the CCP successfully. 30 days after new IP ranges are added to this file, they start being used by Amazon Connect.

Service	Port	Protocol	Comments
Amazon Connect	3478	UDP in/out	Used for media endpoints within the Region, and for call audio for the softphone client.
Amazon Connect	443	TCP in/out	
Amazon EC2	443	TCP in/out	CCP signaling endpoint
CloudFront	443	TCP in/out	Used for hosting web content associated with your instance. Endpoints are determined by the

Service	Port	Protocol	Comments
			location of the end-user client.

Alternatively, for Amazon EC2 endpoints, you can allow access for the following URL and port to allow all Amazon EC2 endpoints rather than all of the IP Address ranges listed in the AWS ipranges.json file:

```
rtc.connect-telecom.{region}.amazonaws.com:443
```

Replace `{region}` with the Region in which you created your Amazon Connect instance, such as `us-east-1`. In certain proxy applications, web socket handling may impact functionality when using this address. You should perform testing to validate before deploying to a production environment.

For CloudFront, you can use the following URL with port 443 to allow traffic for all CloudFront endpoints. Do this instead of including all ranges listed in the AWS ipranges.json file: `https://`**`myInstanceName`**`.awsapps.com`. Replace **`myInstanceName`** with the name of the instance for which to allow traffic. In certain proxy applications, web socket handling may impact functionality when using this address, so you should perform testing to validate before deploying to a production environment.

Port and Protocol Considerations

Consider the following when implementing your network configuration changes for Amazon Connect:

- You need to allow traffic for all addresses and ranges for the Region in which you created your Amazon Connect instance.
- If you are using a proxy or firewall between the CCP and Amazon Connect, increase the SSL certificate cache timeout to cover the duration of an entire shift for your agents, Do this to avoid connectivity issues with certificate renewals during their scheduled working time. For example, if your agents are scheduled to work 8 hour shifts that include breaks, increase the interval to 8 hours plus time for breaks and lunch.
- When opening ports, Amazon EC2 and Amazon Connect require only the ports for endpoints in the same Region as your instance. CloudFront, however, requires the endpoints in the Region closest to where your agents are located. If you have agents in multiple Regions, you need to allow traffic for the endpoints in each Region where agents are using the Amazon Connect CCP. For example, if your instance is US East, and you have an agent physically located in another country, you would need to open the ports for AWS CloudFormation using the IP address ranges for the Region where the agent is located.
- Update the ranges for which traffic is allowed within the 30 days after the ranges are updated in the AWS ipranges.json file. If you don't, you may experience intermittent connectivity issues when using the CCP with a softphone when traffic is routed to the new ranges, but not allowed to connect to your agents using the CCP.
- If you are using a custom CCP with the Amazon Connect Streams API, you can create a media-less CCP that does not require opening ports for communication with Amazon Connect, but still requires ports opened for communication with Amazon EC2 and CloudFront.

Region Selection Considerations

Amazon Connect Region selection is contingent upon data governance requirements, use case, services available in each Region, and latency in relation to your agents, callers, and external transfer endpoint geography.

- **Agent location/network**—CCP connectivity traverses the public WAN, so it is important that the workstation has the lowest latency and fewest hops possible, specifically to the AWS Region where your resources and Amazon Connect instance are hosted. For example, hub and spoke networks that need to make several hops to reach an edge router can add latency and reduce the quality of experience.

When you set up your instance and agents, make sure to create your instance in the Region that is geographically closest to the Region where you create your instance. If you need to set up an instance in a specific Region to comply with company policies or other regulations, choose the configuration that results in the fewest network hops between your agent computers and your Amazon Connect instance.

- **Location of your callers**—Because calls are anchored to your Amazon Connect Region endpoint, they are subject to PSTN latency. Ideally your callers and transfer endpoints are geographically located as closely as possible to the AWS Region where your Amazon Connect instance is hosted for lowest latency.

  For optimal performance, and to limit the latency for your customers when they call in to your contact center, create your Amazon Connect instance in the Region that is geographically closest to where your customers call from. You might consider creating multiple Amazon Connect instances, and providing contact information to customers for the number that is closest to where they call from.
- **External transfers**—from Amazon Connect remain anchored to your Amazon Connect Region endpoint for the duration of the call. Per-minute usage continues to accrue until the call is disconnected by the recipient of the transferred call. The call is not recorded after the agent drops or the transfer completes. The CTR data and associated call recording of a transferred call are generated after the call is terminated. Whenever possible, don't transfer calls that could be transferred back into Amazon Connect, known as circular transfers, to avoid compounding PSTN latency.

Agents Using Amazon Connect Remotely

Remote agents, those that use Amazon Connect from a location other than those connected to your organization's main network, may experience issues relating to their local network if they have an unstable connection, packet loss, or high latency. This is compounded if a VPN is required to access resources. Ideally, the agents are located close to the AWS Region where your AWS resources and Amazon Connect instance are hosted, and have a stable connection to the public WAN.

Rerouting Audio

When rerouting audio to an existing device, consider the location of the device in relation to your Amazon Connect Region. This is so you can account for potential additional latency. If you reroute your audio, whenever there is a call intended for the agent, an outbound call is placed to the configured device. When the agent answers the device, that agent is connected with the caller. If the agent does not answer their device, they are moved into a missed call state until they or a supervisor changes their state back to available.

Using AWS Direct Connect

AWS Direct Connect can help solve for latency and poor call quality between your edge router and AWS resources. It also allows you to configure your edge router to redirect AWS traffic across dedicated fiber rather than traversing the public WAN. This allows for a durable, consistent connection rather than relying on your ISP to dynamically route requests to AWS resources. Keep in mind that this does not solve issues with the private LAN/WAN traversal to your edge router like hub-and-spoke network architecture.

Using Amazon Connect in a VDI Environment

Virtual Desktop Infrastructure (VDI) environments add another layer of complexity to your solution that warrants separate POC efforts and performance testing to optimize. The Amazon Connect Contact

Control Panel (CCP) can operate in thick, thin, and zero client VDI environments as any other WebRTC based browser application does, and the configuration/support/optimization is best handled by your VDI support team. That being said, the following is a collection of considerations and best practices that have been helpful for our VDI-based customers.

- **Location of your agents**—Ideally, there are as few hops as possible with the lowest round trip time between the location from which your agents use the CCP and the VDI host location.
- **Host location of your VDI solution**—Ideally, your VDI host location is on the same network segment as your agents, with as few hops as possible from both internal resources as well as an edge router. You also want the lowest round-trip time possible to both WebRTC and Amazon EC2 range endpoints.
- **Network**—Each hop that traffic goes through between endpoints increases the possibility of failure and adds opportunity to introduce latency. VDI environments are particularly susceptible to call quality issues if the underlying route is not optimized or the pipe isn't either fast or wide enough. While AWS Direct Connect can improve call quality from the edge router to AWS, it will not address internal routing issues. You may need to upgrade or optimize your private LAN/WAN, or redirect to an external device to circumvent call audio issues. In most scenarios, if this is required, the CCP is not the only application that is having issues.
- **Dedicated resources**—at the Network and desktop level are recommended to prevent an impact to available agent resources from activities, such as backups and large file transfers. One way to prevent resource contention is by restricting the desktop access to Amazon Connect users who will be using their environment similarly, instead of sharing resources with other business units who may use those resources differently.
- **Using a soft phone with remote connections**—in VDI environments can cause impact to audio quality. If your agents connect to a remote endpoint and operates in that environment, we recommend either rerouting audio to an external E.164 endpoint or connecting the media through the local device and then signaling through the remote connection. You can build a custom CCP with the Amazon Connect Streams API by creating a CCP with no media for call signaling. This way, the media is handled on the local desktop using standard CCP, and the signaling and call controls are handled on the remote connection with the CCP with no media. For more information about the streams API, see the GitHub repository at https://github.com/aws/amazon-connect-streams.

CCP Connectivity

When an agent logs in, the CCP attempts to connect to the Amazon EC2 signaling endpoints listed in the AWS ipranges.json file, Amazon Connect for media, and CloudFront for web artifacts such as images. When the agent logs out or the browser is closed, endpoints are reselected when the agent next logs in. If a connection to Amazon EC2 or Amazon Connect fails, errors display on the CCP. If a connection to CloudFront fails, web elements such as buttons and icons, or even the page itself fails to load correctly.

Outbound calls:

- When an outbound call is placed, the event signal is sent to the Amazon EC2 endpoint, which then communicates with Amazon Connect to place the call. Upon a successful dial attempt, the agent is bridged in, which anchors the call to the agent's Amazon Connect endpoint. Any external transfers or conferences also uses the anchor until the call is disconnected. Anchoring can help reduce PSTN latency.

Inbound calls:

- When an inbound call is received, the call is anchored to an Amazon Connect endpoint. Any external transfers or conferences also use this anchor until the call is disconnected.
- When an agent is available, the call is pushed through via a new Amazon EC2 connection to their browser and offered to the agent.

- When the agent accepts the call and either the external device has been answered or the CCP determines it can receive a call, a connection to Amazon Connect is established for call media to the agent.

Transferred calls:

- When a call is transferred, the transfer event that signals to place an outbound call to the specified transfer destination is sent to Amazon EC2, which then communicates with Amazon Connect to place the call.
- When the call is connected, the agent is bridged in, anchoring the call to the agent's existing Amazon Connect endpoint. Any external transfers or conferences also use this anchor until the call is disconnected.
- If the agent hangs up after the call is bridged, the agent's connection to the call is terminated, but Amazon Connect hangs on to the call at the Amazon Connect anchor point until there is a far side disconnect. When the call is disconnected, CTRs and associated recordings are generated and made available for the call.

Missed calls:

- If the call is waiting on an agent, customer queue flow logic is used until an agent is available and the call has been successfully routed to that agent.
- If the agent does not accept the call, the agent moves into a Missed Call state and is unable to take calls until the agent, or a call center manager, changes their status to Available again. The caller does not hear ringing while the call is waiting for the agent, and continues to hold until connected with an agent as defined in the customer queue flow logic.

Panic logout:

- If the browser window where the CCP is running is closed, the call remains connected, but opening the browser and logging back in will not allow you to re-establish the media connection. You are still able to transfer or end the call, but no audio path is established between the agent and caller.

Troubleshooting Issues with the CCP

Troubleshooting CCP issues requires support from your network operations, system administrator, and VDI solution teams to collect the appropriate level of information to identify root cause and drive resolution. To help determine the appropriate resources to engage, it's important to break issues down into those with similar symptoms. The following guidance has been helpful in assisting Amazon Connect customers in resolving CCP issues with their operations support teams.

Topics in this section:

Common CCP Issues

The following are common issues encountered when using the Amazon Connect CCP.

- **CCP does not initialize/connect**—The most common causes are missing port/IP whitelist entries, not allowing browser microphone access, or not answering your external device. Be sure that you have whitelisted all IPs covered in the Network Ports and Protocols (p. 39) section of this guide, and that you have allowed microphone access to your browser when prompted.
- **Periodic connection errors**—The most common cause is network contention, or there may have been an ipranges.json update and the new entries have not been whitelisted. For more information, see the Network Ports and Protocols (p. 39) section of this guide.
- **Missed calls, state change delays, and CCP unresponsive**—In most cases, this is intermittent and directly correlated with resource contention in the agent's workstation, network, or both. This can be made worse, or caused directly, by a poor, unstable, or strained connection to AWS resources at the private WAN/LAN, public WAN levels, or local workstation resource contention.

The following are common issues with call quality when using the CCP. Call quality encompasses a large range of potential causes and is best approached by first identifying the types of issues that you're having.

- **Latency/cross-talk**—in a voice connection manifests as a delay between when something is said and when the person on the other end hears it. In some use cases that require a lot of conversation, high latency can create situations in which both parties are talking over each other. The PSTN and agent latency need to be calculated in this scenario to identify contributing factors and take action to reduce PSTN latency, agent latency, or both. For more information, see the PSTN and agent connection latency section of this documentation.
- **One way audio**—is when the agent can't hear the caller or the caller can't hear them. This is normally indicative of an issue with the agent's workstation at the hardware, network, resource levels, or all three. It and can also be related to browser microphone permissions or headset issues. For more information, see the Monitoring Workstations (p. 38) section of this guide.
- **Volume increase or decrease**— can happen at the beginning or intermittently during the call, and it's important to differentiate the two for troubleshooting purposes. Typically, this relates to forwarding calls to or from Amazon Connect that inherit this from an issue with the third party transfer.
- **Audio choppy, cutting out, echo, reverb, or other signal noise**—could also manifest as a robotic sound or other distortion making it difficult for either the agent, caller, or both parties to understand what’s being said. This is normally indicative of an issue with the agent's workstation at the hardware, network, resource levels, or all three. For more information, see the Monitoring Workstations (p. 38) section of this guide.
- **Wobble**—is the effect that media codecs can have on audio that manifests as the slowing down and speeding up of audio to combat high jitter and latency. This is normally indicative of an issue with the agent's workstation at the hardware, network, resource levels, or all three. For more information, see the Monitoring Workstations (p. 38) section of this guide.
- **Disconnects**—can happen at any point in the call. It is important to note when during the call that the disconnections occur to identify a pattern. For example, disconnects on call transfers to a specific external number typically relate to forwarding calls to or from Amazon Connect that inherit this from an issue with the third party transfer. They can also be related to circular transfers, which means transferring calls out of Amazon Connect and back in the same call.

Useful Troubleshooting Tools and Information

The following tools and information can be helpful with troubleshooting issues with Amazon Connect.

- **Instance ARN**—Provide your instance ARN when you contact AWS support so that they can see the activity in your Amazon Connect instance. You can find the ARN for your instance on the Overview page that you access by choosing the alias of the instance from the Amazon Connect console.
- **Call recordings**—are very useful, not only to illustrate and determine reported behavior, but also to rule out audio issues from the agent's side. Recordings in Amazon Connect are done at the instance

side of the interaction, before the audio traverses the agent connection. This allows you to determine if the audio issue was isolated to the agent's side of the interaction or if it existed in the audio received by the agent. You can find call recordings associated with a contact in the Contact Search report.

- **Contact IDs from the CTR**—Provide when you contact AWS support.
- **Agent desktop performance/process logs**—can help rule out local resource/network contention.
- **Contact Control Panel logs**—to track agent actions and timing. To download CCP logs, choose the settings cogwheel in the CCP, and then choose **Download logs**. The logs are saved to your browser's default download directory.
- **Network utilization logging/monitoring**—specifically for latency and dropped packets on the same network segment as your agents.
- **Private WAN/LAN network diagram**—outlining connection paths to the edge router to AWS to explain network traversal.
- **Firewall whitelist access**—to verify that IP/port ranges are whitelisted as described in Network Ports and Protocols (p. 39).
- **Audio capturing and analytic tools**—for latency calculations from the agent's workstation.
- **AWS region latency test tools**—such as the Amazon Connect Call Control Panel Connectivity Tool.

Gathering Helpful Information using the Streams API

For tracking and troubleshooting issues at scale, collecting data surrounding overall call quality is recommended. Anytime poor call quality is experienced, agents can note the current time and corresponding disposition code by using the disposition key chart, as shown in the following chart. Alternatively, you can use the Streams API to incorporate your own report and issue feature in the custom CCP to write these dispositions with corresponding call information to a database, like Amazon DynamoDB. For more information about the Amazon Connect Streams API, see the GitHub repository at https://github.com/aws/amazon-connect-streams.

Example Agent Issue Report Disposition

The following example disposition keys are listed by symptom, scenario, and severity.

Symptom

- **S**—Softphone error
- **M**—Missed calls
- **L**—Latency causes poor quality
- **P**—Starts off OK, gets progressively worse over time
- **D**—Disconnected calls
- **W**—One way audio; for example, the agent can hear the customer, but the customer cannot hear the agent
- **V**—Volume too quiet or too loud
- **C**—Choppy/cuts in and out intermittently

Senario

- **O**—Outbound call
- **I**—Inbound call
- **T**—Three-way call

Severity

- **1**—Small impact, but can use the CCP effectively
- **2**—Medium impact, communication is difficult, but can still service calls
- **3**—Large impact, cannot use the CCP to take calls

For example:

- 5:45PM agentName LT2 (latency on a three-way call with medium impact).
- 6:05PM agentName DO3 (disconnected three-way call with large impact).
- 6:34PM agentName MI3 (missed inbound call with large impact).

Analyzing the Data

The following guidelines can assist you in analyzing the data to identify issues in your environment.

- Use the CTR / Contact search report to identify the contact IDs for the contacts during which call quality issues occurred. The CTR includes a link to the associated call recording, and additional details that you can use for symptom verification and to provide to your AWS support representative.
- Use the agent name and timestamp in the CTR to get a sense of the types of issues you're experiencing and their prevalence by agent, symptom, scenario, and severity over time. This will allow you to see if issues are happening around the same time, surround a specific event, or are isolated to specific agents or agent actions. You can also easily identify and access associated call recordings and associated contact IDs available if you need to engage support.
- Correlate data sources, such as local network logs, CPU/disk/memory utilization and process monitor logs from the operating system on the client workstation. This lets you correlate events by agent over time to rule out local resource contention as a cause or contributor.
- Analyze data by symptom and scenario reported per minute or per hour to create heat maps of an issue by type and severity by agent over time. Doing this is especially helpful in environmental troubleshooting as you may find clustered impacts associated with scheduled activity like backups or large file transfers.
- If you can't find any evidence of local resource contention or derive any noteworthy correlations, you can use the contact IDs collected to open a support case. If issues experienced are intermittent in nature, they most likely relate to issues with the agent's workstation, network connectivity, or both.

Validation Testing

Voice quality issues can have many contributing sources. It's important to run controlled tests and monitor the same environment or workstation as those reporting the issue, and be able to reproduce the same use cases. Consider the following general testing recommendations for measuring and gathering data to investigate voice quality issues.

PSTN and Agent Connection Latency

For troubleshooting cross-talk issues, you need to differentiate and measure agent and raw PSTN latency contributions, as they require different remediation efforts.

- [overall_latency] is the total latency experienced between caller and agent. This latency can be calculated as [overall_latency] = [agent_latency] + [pstn_latency].
- [pstn_latency] is the latency between Amazon Connect endpoint and the caller. This latency can be calculated as [pstn_latency] = [overall_latency] - [agent_connection_latency]. This latency can be improved by using a different Amazon Connect Region location or avoiding external and circular transfers to geographically distant endpoint locations.

- [agent_latency] is the latency between Amazon Connect endpoint and the agent. This latency can be calculated as [agent_latency] = [overall_latency] - [recording_latency]. This latency can be improved by using AWS Direct Connect for agents on-premises, avoiding the use of VPN connections, improving private WAN/LAN performance/durability, or using an Amazon Connect Region location closer to your agents. Depending on your use case, selecting a different Region selection may also increase [pstn_latency].
- [redirect_latency] is the latency resulting in redirecting audio to an external device. This latency can be calculated by measuring [overall_latency] once with redirect and once without and take the difference between the two.
- [forward_latency] is the latency resulting in forward calls to or from Amazon Connect. This latency can be calculated by measuring [overall_latency], once with forward and once without, and take the difference between the two.

Measuring Latency

- Reproduce your use case. Any deviations need to be measured and accounted for, because they skew test results.
- Match production controls and environment as much as possible. Use the same flows, phone numbers, and endpoint locations.
- Note the geographical locations of your callers, agents, and external transfer destinations, where applicable. If you are servicing multiple countries, each country should be tested individually to provide the same test coverage that your agents experience in production.
- Note mobile and land line use in your tests. Mobile networks can add latency and need to be measured and considered for customer, agent, and transfer endpoints, where applicable.
- Reproduce the business use case. If the agents use conference and transfer, be sure to test those scenarios. If circular transfers occur, which are not recommended, be sure to test those as well.
- Reproduce the agent environment by including the workstation environment, located on the same network segment, and using equipment your agents would use.

Requirements for Testing Latency

To perform effective testing for latency, the following are required:

- Call recording enabled to capture [agent_latency]. Without call recording, you can calculate only [overall_latency].
- A customer phone source. For testing, confirm call quality on an actual call from a customer.
- An agent phone, if redirecting audio to an external device. You must be able to record the input and output of this device.
- A third-party transfer endpoint, if applicable. Testing is best when performed on actual calls or transfers from a third party.
- An agent workstation with sound recording or analysis software.
- Reproducible use cases. Troubleshooting can be difficult for issues that cannot be reproduced.
- NTP or other method to sync timestamps to facilitate identifying specific contacts and when they occurred, especially when activity is occurring across multiple time zones.

Testing Inbound Calls Using a Soft Phone

This process allows you to complete a latency test scenario in ~15 seconds. Analyzing the results and marking timestamps takes approximately 1-2 minutes per recording.

1. Go to a quiet location.

2. Configure agent workstation to play audio from external speakers and make sure they are turned up.
3. Use the agent workstation to log in to the CCP.
4. Start recording using an audio capturing tool on the agent workstation.
5. From the customer's phone source, use a speaker phone to call the incoming number for your Amazon Connect instance. This could really just be any external phone source to simulate a customer call.
6. Answer the incoming call using the soft phone on the agent workstation.
7. Make sure that the customer phone is not muted.
8. On the customer side, use an object or your hand, tap loudly on the desk or table, and then immediately mute the customer phone.
9. Wait 3 or more seconds. Repeat steps 7-8 at least 3 times.
10. Stop recording on the agent workstation.
11. Open the recording in your audio analysis tool. You should be able to see both the initial tapping sound that you made on the desk, and the tapping sound on the agent line on the other end. Take the three deltas and average for your [overall_latency].
12. Optionally, to calculate [agent_latency], open the associated Amazon Connect call recording in your audio analysis tool. You should be able to see both the initial tapping sound and the sound when it arrives to the agent at the other end. Take the three deltas and average for your [recording_latency]. [agent_latency] = [overall_latency] - [recording_latency]. Repeat as needed.

Modify the test plan as necessary to fit your use case. As the steps change, the process of recording and analyzing the audio is the same. If you need to test conferences and transfers, take measurements as normal, and then take another measurement when the conference is active with the third party transfer endpoint.

Interpreting the Test Results

The impact of increasing [overall_latency] begins to be noticeable at approximately 300ms and can result in crosstalk above 500ms. The impact, and what level of latency is considered acceptable, depends on your use case. For recommended remediation efforts for decreasing latency, see the PSTN and Agent Connection Latency (p. 46).

Document History

The following table describes the update history for Amazon Connect documentation.

Change	Description	Date
Added a Troubleshooting and Best Practices topic	Added a new topic, Amazon Connect Troubleshooting and Best Practices (p. 38), that covers best practices for agent connectivity using the CCP, and troubleshooting connectivity and call quality issues in Amazon Connect.	October 18, 2018
Added a topic about using service-linked roles in Amazon Connect.	Added the Using Service-Linked Roles for Amazon Connect (p. 26) topic, which described using service-linked roles in Amazon Connect.	October 17, 2018
Updated the steps for adding an Amazon Lex bot to your instance.	Updated the section Add an Amazon Lex bot to Your Instance (p. 25) to include steps for selecting a bot from a different region.	July 30, 2018
Added a Release Notes topic.	Published a Release Notes topic that lists the changes and updates to Amazon Connect during the previous month. For more information, see Release Notes (p. 6).	June 11, 2018
Updated topic about metrics sent to Amazon CloudWatch Logs.	Updated the topic Monitoring Amazon Connect in Amazon CloudWatch Metrics (p. 28) to include additional metrics and update the descriptions for all metrics.	April 19, 2018
Added content for using SAML for identity management.	New content added that describes how to configure your instance to use SAML for identity management to enable single sign-on. For more information, see Configure SAML for Identity Management in Amazon Connect (p. 14).	March 30, 2018
Updated topic on using AWS Lambda functions with Amazon Connect	Replaced existing content with new information and examples to make the topic current with the technology. For more	January 05, 2018

Change	Description	Date
	information, see Using AWS Lambda Functions with Amazon Connect (p. 32).	
Added Port Your Current Phone Number	Added information about how to port your current telephone number to Amazon Connect. For more information, see Port Your Current Phone Number (p. 19).	November 10, 2017
Updated Salesforce integration information	Updated the steps to integrate Amazon Connect with Salesforce to clarify settings. For more information, see Amazon Connect and Salesforce Integration (p. 36).	October 27, 2017
Initial release	Initial release of the *Amazon Connect Administrator Guide*.	March 28, 2017