

Setup and Installation Guide



February, 2024

© Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved. SPDX-License-Identifier: CC-BY-SA-4.0

Notices

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

Abstract

This guide details the integration between Amazon Connect and Salesforce Lightning. It covers the installation, configuration, and operation of the two primary components of the integration: the Amazon Connect CTI Adapter for Salesforce and the AWS Serverless Application Repository for Amazon Connect Salesforce integration.

Salesforce Lambda Versions

The Amazon Connect CTI Integration consists of two components - A salesforce package we refer to as the CTI Adapter, and an AWS Serverless application, which contain a set of lambdas to be deployed to your AWS environment. For more information on the lambdas, [visit here](#).

The version of the Lambdas and the version of the CTI Adapter may differ as they are two separate packages. If a specific version of the lambdas package is needed to run with the CTI Adapter and vice versa, it will be stated [here](#).

License Summary

The documentation is made available under the Creative Commons Attribution-ShareAlike 4.0 International License. See the [LICENSE file](#).

The sample code within this documentation is made available under the MIT-0 license. See the [LICENSE-SAMPLECODE file](#).

Table of contents:

- [Release Notes](#)
 - [Important Notes](#)
 - [Google Chrome third-party cookies support](#)
 - [Summer '23 Release](#)
 - [Salesforce Enhanced Domains](#)
 - [Spring '22 Release](#)
 - [WebRTC Plan-B Deprecation](#)
 - [Installing as Admin](#)
 - [5.22 February 2024](#)
 - [5.21.1 November 2023](#)
 - [5.21 October 2023](#)
 - [5.20.1 July 2023](#)
 - [5.19 April 2022](#)
 - [5.18 January 2022](#)
 - [5.17 November 2021](#)
 - [5.16 August 2021](#)
 - [5.15 July 2021](#)
 - [5.14 June 2021](#)
 - [5.13 April 2021](#)
 - [5.12 March 2021](#)
 - [5.11 March 2021](#)
 - [5.10 February 2021](#)
 - [5.9 December 2020](#)
 - [5.7 November 2020](#)
 - [5.5 October 2020](#)
 - [5.4 Late September 2020](#)

- 5.3 September 2020
- 5.1 Late August 2020
- 5.0 August 2020
- 4.5 April 2020
- 4.4 March 2020
- 4.2 December 2019
- 4.1 November 2019
- Key Benefits and Requirements
 - Requirements
 - Prerequisites - Amazon Connect CTI Adapter
 - Prerequisites - AWS Serverless Application Repository for Salesforce
 - Browser Compatibility
 - Salesforce Lightning Support
- Installing the CTI Adapter and Salesforce Lambdas
 - Amazon Connect Salesforce CTI Adapter Managed Package
 - Amazon Connect Salesforce Lambda package
 - Setting up the ExecuteAwsService Named Credential
- Setting Up The CTI Adapter Using Guided Setup
 - Guided Setup Prerequisites
 - Create Named Credential
 - Create Connected App
 - Guided Setup Additional Instructions
 - Retrieve Amazon Connect Instance Url
 - Add users to the Call Center
 - Add users to a Permission Set
 - AC_Administrator
 - AC_Manager
 - AC_Agent
 - Configure the Toolkit settings
 - Create the Softphone Layout
 - Retrieve the Salesforce API Version
 - Setting up the Salesforce API User
 - Allowing the API user to authenticate using password
 - Setting up the SecretsManager Secret
 - Test the Salesforce Lambda Core Functionality

- Validate the core functionality
- Allow Amazon Connect to Access the sfInvokeAPI Lambda Function
 - Add the Lambda function to your Amazon Connect instance
- Setting Up The CTI Adapter Managed Package Manually
 - Set Access Permissions
 - AC_Administrator
 - AC_Manager
 - AC_Agent
 - Configure the Lightning Experience
 - Configure Service Console
 - Allowlist Your Salesforce Org with Amazon Connect
 - Modify the Call Center
 - Configure the Toolkit settings
 - Create the Softphone Layout
 - Initial CTI Adapter Configuration
 - Add the CTI Adapter Console App
 - Enhanced Agent Logout
 - Validate Basic Screenpop
- Setting Up The Salesforce Lambdas Manually
 - Prerequisite Configuration and Data Collection
 - Check your Salesforce API Version
 - Create a New Connected App
 - Create a new API user
 - Allowing the API user to authenticate using password
 - Gather Amazon Connect information
 - Store Salesforce Credentials in AWS Secrets Manager
 - Install the Amazon Connect Salesforce Lambda package
 - Compatibility Table
 - Instructions
 - Test the Core Functionality
 - Validate the core functionality
 - Allow Amazon Connect to Access the sfInvokeAPI Lambda Function
 - Add the Lambda function to your Amazon Connect instance
- Upgrading from an Earlier Version
 - Upgrading the Salesforce Lambdas

- Deploying the New Lambda Package
- Removing the Previous Lambda Package
 - Unlink the sfInvokeAPI Function
 - Remove triggers from the Lambda functions of the previously-deployed stack
 - Delete the Previously-Deployed Stack
- Upgrading the CTI Adapter
 - AC CTI Adapter
 - AC CTI Script
- CTI Adapter Installation Troubleshooting and Common Issues
 - I upgraded my adapter to v5.10, but I cannot see the CCP Config changes
 - Error "refused to run the JavaScript URL because it violates the following Content Security Policy directive..."
 - Error "refused to frame" Visualforce page
 - Browser refreshing when trying to open lightning components
 - How do you fix it?
 - Why does this happen?
 - What are the Disable X Trigger options in the Custom Settings?
 - I upgraded my adapter to v5, but I don't see the CTI Flows feature.
 - I upgraded my adapter from v3 to v5 and we lost some screenpop functionality.
 - The CCP doesn't show up in service console and I instead see the following image:
 - Certain picklists are missing picklist items.
 - How to remove permissions to Visualforce pages, Apex classes for a desired profile
- CTI Adapter Configuration
 - CTI Adapter Details
 - Single Sign On Settings
 - Identify the SSO URL components
 - Configure the CTI Lightning Adapter in Salesforce
- CTI Attributes
 - Attribute Properties
 - CTI Attributes Example Walkthrough
 - Adding a Text-based CTI Attribute
 - Adding a Hyperlink-based CTI Attribute
 - CTI Attribute Additional Features
 - Enabling CTI Attribute Additional Features
- CTI Flow

- Create CTI Flow
- Accessing CTI Flow Block Values
 - JSON Paths
 - Accessing Object Properties
 - Accessing CTI Flow Object Properties
 - Why Would I Use This?
- Presence Sync Rules
 - Configuring Statuses
 - Amazon Connect System Statuses
 - Create Presence Statuses in Amazon Connect
 - Create an Amazon Connect status
 - Create Presence Statuses in Salesforce
 - Create a Salesforce presence status
 - Configure Enabled Service Presences Status Access in Salesforce
 - Configure Presence Sync Rules
 - Create a Presence Sync Rule
- Localization
 - Prerequisites
 - Setting your preferred language
 - Additional Notes
- Set Agent Status on Session End
- Contact Lens
 - Prerequisites
- CTI Actions
 - CCP Overlay
 - Example
 - Receiving Data from CTI Flows
 - Upgrading from an earlier version
- Recording Controls
 - Setup
 - Using Recording Controls with Contact Flows
 - Recording Named Credential
 - Synchronizing Recording State with Contact Attributes
- Voicemail Drops
- Chat Widget Integration

- Setup Experience Cloud Site
- Setup Chat Widget in Amazon Connect
- Create Required Visualforce Pages
- Setup Chat Widget for your Experience Cloud Sites
- MultiChat Management
 - Recommendations
 - Example Use
- Amazon Q Integration
- Voice Id
- Setting up Medialess
 - Medialess
 - Prerequisites
 - Set Up for CITRIX VDI Platform
 - Set Up for Other VDI Platforms
- Accessing the Salesforce API from Amazon Connect Contact Flows Using AWS Lambda
 - Salesforce Lookup
 - Salesforce Create
 - Salesforce Update
 - Salesforce Phone Lookup
 - Salesforce Delete
 - Salesforce query
 - Salesforce queryOne
 - Salesforce createChatterPost
 - Salesforce createChatterComment
 - Salesforce search
 - Salesforce searchOne
- Amazon Connect Historical Metrics in Salesforce
 - Configuring the AWS Services
 - Configuring the Historical Reports in Amazon Connect
 - Creating the AWS Lambda Trigger for the Queue Data
 - Creating the AWS Lambda Trigger for the Agent Data
 - Verifying the Data Import in Salesforce
 - Viewing Amazon Connect Reports in Salesforce
- Amazon Connect Real-Time Metrics in Salesforce
 - Deployment and Configuration

- Adding Real-Time Reports to the Service Console
- Contact Channel Analytics
 - Call Recording Streaming
 - Cloudformation Template
 - Enabling call recording streaming
 - Adding users to the AC_CallRecording permission set
 - Adding Contact Channel Analytics to the Service Console
 - Recording Transcripts
 - Enabling recording transcription
 - Accessing transcriptions
 - AI Driven Contact Analysis
 - Enabling AI Driven Contact Analysis
 - Accessing the AI Driven Contact Analysis
- Contact Trace Record Import
 - Contact Trace Record Import
 - Enabling Contact Trace Record Import
 - Adding Contact Trace Records to the Service Console
 - Display Additional Contact Trace Record Data
 - Customizing the AC Contact Trace Record Layout
- Postcall Contact Lens Import
 - Contact Lens Import
 - Creating the AWS Lambda Trigger for the Contact Lens Data
 - Enabling Contact Lens Import
- Configuring My Domain in Salesforce
 - Register Your Domain
 - Deploy the Domain to Your Users
- Configure Salesforce Omnichannel for Testing
 - Enable Omnichannel
 - Enable Omnichannel in Your Salesforce Org
 - Configure Presence Statuses
 - Add a Presence Status
 - Configure Profiles to Use the New Statuses
 - Modify Profiles to Use New Statuses
 - Add Omni-Channel to the Utility Bar
 - Add the Omni-Channel Utility Item

- Appendix B: Configuring Salesforce as Your Identity Provider
 - Prerequisites
 - Configuring Salesforce as an Identity Provider
 - Setup Identity Provider & Download Metadata
 - Configure the Identity Provider, Policy, and Role in the AWS Console
 - Configure the Identity Provider
 - Create the IAM Role and Policy
 - Complete the Base Salesforce Configuration
 - Create the Connected App in Salesforce
 - Complete the Amazon Connect Configuration
 - Add Users to Amazon Connect
 - Final Configuration for the Lightning Experience
 - Create the Amazon Connect SSO URL
 - Configure the CTI Lightning Adapter in Salesforce For SSO
- Appendix C: CTI Flow Sources and Events
- Appendix D: CTI Flow Examples
 - Voice Contact Screenpop (Legacy Adapter Support)
 - Chat Contact Screenpop
 - Click-to-Dial
 - Screen Pop on Customer Phone Number
 - Screen Pop a Case on Contact Attribute Data (if it exists) or Pop a New Case (if it does not)
 - Create a Task (Call Activity) and Pop That Task
 - Screenpop on Customer Email Address (in contact attribute data)
 - Create a Task (Call Activity) and Pop That Task
 - Create a Task (Call Activity) and Pop That Task using CTI Actions
 - Create a Record on Chat Connected and Screenpop
 - Screenpop Chat Contact on View
 - Default CTI Flows
- Appendix E: Integration with Salesforce High Velocity Sales
 - What is High Velocity Sales?
 - Enabling the Integration with High Velocity Sales
 - Enable High Velocity Sales
 - Call Outcomes for Branching
 - Define Call Outcomes for Branching

- Assign HVS permission sets to Connect Users
 - Assign the permission set
 - Create Sales Cadence
 - Assigning Prospects
 - Create and Map Dispositions
 - Create and map disposition fields
 - Setup CTI Flows for High Volume Sales
 - Configuring the CTI Flow
 - Expected Behavior
- Appendix F: CTI Flow Blocks
 - If-else
 - HTTP Request
 - Get Property
 - Get All Properties
 - Format Phone Number
 - Format Phone Number (E164)
 - Format a Date object
 - Is Truthy?
 - Set Property
 - Log to Console
 - Show Modal
 - Enable Click To Dial?
 - Enable Click To Dial
 - Disable Click To Dial
 - Get App View Info
 - Get Softphone Layout
 - Get Agent Workload on Salesforce
 - Complete High Velocity Sales Work With Task Saved
 - Refresh View
 - Show Softphone Panel
 - Hide Softphone Panel
 - Set Softphone Panel Height
 - Set Softphone Panel Width
 - Screenpop Object
 - Screenpop Url

- Screenpop Object Home
- Screenpop List
- Screenpop Search
- Screenpop New Record
- Search And Screenpop
- Run Apex
- Get Agent State from Salesforce
- Set Agent State on Salesforce
- Login Agent on Salesforce
- Logout Agent on Salesforce
- Save (or Create) a Record
- Create a Task
- Is Contact "Do Not Call"?
- Dial Number
- Mute Agent
- Unmute Agent
- Get Agent Status from Connect
- Set Agent Status on Connect
- Set Agent Status By Name on Connect
- Set Agent as Available on Connect
- Get Quick Connection List
- Get Transfer Connection List
- Get Endpoint by Phone Number
- Get Available Agent States
- Get Agent Name
- Get Agent Extension
- Get Agent Deskphone Number
- Is Agent Softphone Enabled?
- Change Agent to Softphone
- Change Agent to Deskphone
- Get Agent Configuration
- Get Agent Dialable Countries
- Create Task Contact
- Get Contact Attribute
- Is Voice Contact?

- Is Chat Contact?
- Is Task Contact?
- Is Contact Inbound?
- Is Contact Transfer?
- Is Callback?
- Get Contact Properties
- Get Customer Phone Number
- Get Contact Interaction Metadata
- Pop Task Contact's Reference Urls
- Query value
- Get Salesforce Lead Id
- Get Salesforce Contact Id
- Retrieve Salesforce Record
- Extract Value
- Open Salesforce Primary Tab
- Open Salesforce Sub Tab
- Get Focused Primary Tab Object Id
- Get Focused Subtab Object Id
- Call jQuery Method
- Replace String
- Text Starts With Value
- Text Ends With Value
- Join Strings
- Multiply
- Divide
- Get Tab Object Map
- Close Salesforce Tab
- Delay
- Get Primary Tab Ids
- Get Tabs With Matching Url
- Length
- Slice
- Cast a Value to a Type
- Get CCP Logs
- Clear All Properties

- [Unset Property](#)
- [Is Task Contact?](#)
- [Create Task Contact](#)
- [Start Recording](#)
- [Stop Recording](#)
- [Update Contact Attributes](#)
- [Get Payload](#)
- [Send Data to CCP Overlay](#)
- [Leave a Voicemail](#)
- [Destroy Agent Connection to Live Contact](#)
- [Clear Contact](#)

Release Notes

Important Notes

Google Chrome third-party cookies support

The CTI Adapter v5.21 now provides support for third party cookies (see [Amazon Connect third party cookie documentation](#)). After you upgrade to the latest version of the CTI Adapter (v5.21+), agents will be prompted to allow third-party cookies from Amazon Connect:

1. When agents open the CCP within the CTI Adapter, a new **Allow access to cookies** banner appears. It has one action button: **Grant access**.
2. When agents choose **Grant access**, the browser displays a prompt to authorize the use of third-party cookies.
3. Agents must select **Allow** on this second pop-up, and then proceed to log in.

Note: If the agent does not follow steps above, please see [our documentation](#) on how to resolve.

Summer '23 Release

The Salesforce summer release '23 blocks Username-Password Flow by default (see more details [here](#)). If your org uses this version of Salesforce, please unblock the flow by following [these](#) instructions.

Salesforce Enhanced Domains

Salesforce is making changes to the instance domains on account of the [enhanced domains](#) feature in the Spring 23 release. Once this feature is enabled, you must migrate the CTI adapter to using these new domains. See [here](#) for migration instructions.

Spring '22 Release

The Salesforce Spring '22 release introduces a change that will likely cause an install or update to any version of the adapter before 5.18 to fail. In addition if you are using the `ac_PhoneCallListView` component in any version of the adapter, the loading of your component may fail. This component has been deprecated in v5.18.

WebRTC Plan-B Deprecation

The Plan-B deprecation should not affect any current users of the CTI Adapter, as we utilize the embedded CCP and do not build in connect-rtc-js separately.

Installing as Admin

Please **confirm that the application was installed for admins only** (see [installation](#) for more details). If you did this by accident, then you will have to [manually edit the profiles](#) to remove the permissions to the objects and pages created by the app. If you are updating the package, please verify that all users have the proper AC permission set.

Important: when upgrading the CTI Adapter, please make sure that the Salesforce Lambdas are also updated to the newest version. Also review the [CTI Adapter Installation Troubleshooting and Common Issues](#) section for known issues and troubleshooting.

5.22 February 2024

- **Note:** If you wish to use the v5.22 lambdas, you will need to upgrade your CTI Adapter to v5.22. Consult the [compatibility chart](#).
- **Feature:** Citrix Support: Enabled native VDI support for Citrix. [See Documentation](#).
- **Feature:** Early Get User Media(GUM): Enabled support for the CCP feature EarlyGUM. [See Documentation](#)
- **Feature:** MultiChat Management: Provided support for Multi-Chat Management. [See Documentation](#).
- **Enhancement:** Amazon Q: Amazon Q has undergone a change and goes by a new name. As such, it has been reflected in our documentation. Here is the documentation for [Amazon Q](#)
- **Enhancement:** Recording Controls: Updated the Recording Controls feature to allow users to specify the Named Credential they want to use per CTI Adapter in the "Recording Named

"Credential" field. This field will not be used if this feature is enabled. If the feature is enabled but no value is provided, a default value of "AmazonConnectAPI" will be assumed.

- **Enhancement:** Salesforce Lambdas:
 - Provided support for queue names with special characters.
 - Updated the Salesforce Lambdas to support new fields for Agent Performance, Historic Queue Metrics, and Contact Lens.
- **Enhancement:** Troubleshooting: Added new section with additional troubleshooting for known problems
- **Enhancement:** Triggers:
 - Fixed typo in CCA Case Trigger + CCA Contact Trigger
- **Enhancement:** Presence Sync:
 - Clarified in our documentation that Presence Sync is not supported in Salesforce Classic adapters. It's listed under the section for Salesforce classic, so this was done to prevent confusion
- **Bug Fix:** AC Contact Channels: `InteractionDuration` value will be updated only once after the call ends.
- **Bug Fix:** Phone numbers: Upgraded the library responsible for formatting numbers to latest version to support calls to more regions.
- **Bug Fix:** Guided Setup: Improved the process to allowlist user URLs.
- **Bug Fix:** Recording Controls: Recording Controls tab will now be visible on the first session load in the user's salesforce instance.
- **Documentation Change:** Medialess: Created new documentation page for setting up medialess ([Link to page](#)).
- **Documentation Change:** Historical Metrics: Added clarifying information to setup historical metrics.
- **Documentation Change:** Upgrading from an earlier version: Added new documentation on upgrading [Salesforce Lambdas](#)
- **Documentation Change:** CTI Flow Blocks: Updated with latest CTI Flow blocks. Added new section about accessing CTI flow block values ([Link to section](#)).

5.21.1 November 2023

- **Bug fix:** Google Chrome third party cookie support for GovCloud instances: The v5.21.1 patch includes updated third party cookie support for GovCloud instances.
- **Enhancement:** [Python 3.10 runtime](#) now available in Amazon Connect Salesforce Lambda package v5.19.7 to address AWS ending support for Python 3.7 in AWS Lambda.

5.21 October 2023

- **Enhancement:** Google Chrome third party cookie support : Salesforce CTI Adapter v5.21 enables requests for third party cookies within Salesforce domains to support Amazon Connect. See [Amazon Connect third party cookie documentation](#) for further information.

5.20.1 July 2023

- **Enhancement:** Amazon Connect Streams API Upgrade : The Amazon Connect Streams API has been upgraded to version 2.2.0 for improved performance and functionality.
- **Enhancement:** CCP Element Editor Permission Change : For CCP Element Editor, editing features was previously available to users assigned to permission sets Agent (AC_Agent), Manager (AC_Manager), and Administrator (AC_Administrator). Starting from this version, only users with the Administrator permission set (AC_Administrator) will be able to view and edit feature. This change is designed to restrict modification access of CCP Overlay Elements.
- **Enhancement:** Chat Widget Integration Setup Process Changes: The setup process for Chat Widget Integration has been updated to enhance the integration experience and security.
- **Backward Incompatibility Notice::** Chat Widget Integration Update : Customers who have previously set up Chat Widget Integration will need to redo the setup process due to changes introduced in this version. This ensures compatibility with the latest enhancements.*Note: To avoid any downtime of feature, set up should be completed before upgrading the version*
- **Security:** Improved Amazon Connect Instance Security : Throttling mechanisms have been introduced to enhance the security of Amazon Connect Instances, ensuring a safer environment for users and their data.

5.19 April 2022

- **Enhancement:** replace call recording audio streaming via cloudfront distribution with the connect native get-recording endpoint. This change makes it so that the cloudfront infrastructure and associated setup process is no longer necessary. Please note that this change will remove audio recording infrastructure from your AWS account, please make sure to test this change before fully deploying.
- **Enhancement:** add IgnorePermissionSet setting to FEATURE_WISDOM_PANEL feature. The setting determines whether the AC_CallRecording/AC_Administrator permission set is checked before showing Wisdom to the logged in user.
- **Bug fix:** CTI Flows on contact events will fire after the page was reloaded during a contact's life cycle

- **Bug fix:** Fixed an issue where we would create a CCACase or CCAContact batch job even if there were no updates to any related fields.

5.18 January 2022

- **Bug Fix:** Updated the **Get Salesforce Contact ID** block to accept E.164 numbers.
- **Bug Fix:** Fixed **onMessage** event name and label which was causing CTI flows to not trigger.
- **Bug Fix:** Fixed stray template tag in `ac_contactChannelListView` causing Spring '22 package installation failure.
- **Bug Fix:** Deprecated `ac_PhoneCallListView` LWC, as it is an artifact of an old version of the adapter and was causing Spring '22 package installation failure.
- **Bug Fix:** Fixed issue where switching contact tabs didn't update the CCP overlay attributes.
- **Bug Fix:** Fixed issue where some `sfInvoke` operations were returning complex JSON objects that don't work with Connect Contact Flows

5.17 November 2021

- **Feature:** Added the integration with Wisdom, which delivers articles and article recommendations to agents. See [here](#) for more details.
- **Feature:** Added the integration with Voice id, which provides real-time caller authentication. See [here](#) for more details.
- **Bug Fix:** Fixed a bug where CTI Actions would only load if you switched overlay tabs. Now they will load immediately.
- **Bug Fix:** Fixed a few bugs with Contact Attributes Overlay.
 - Where you needed to set they would not populate in the overlay unless the CTI Attribute Name value was the same as the contact attribute key.
 - Selecting DisplayValue of `Key` did not show just the Key value.
 - When using the ShowAllAttributes feature, the already configured CTI Attributes did not maintain the same HTML formatting as before.
- **Bug Fix:** Fixed a bug where DialedNumber__c was not filled on outbound calls.
- **Bug Fix:** Fixed a bug where Update Contact Attributes didn't work for Chat or Task contacts.
- **Bug Fix:** Fixed a bug where the CTI Flow payload would only contain the CTI Action Additional Data when both CTI Action Payload and Additional Data are configured. Now the CTI Flow payload will have both the CTI Action Payload and Additional Data
- **Enhancement:** Added two new CTI Flow Blocks - Destroy Live Contact and Clear Contact.

5.16 August 2021

- **Feature:** Added a `callIncomingDuration` field to the `Contact Interaction Metadata` CTI Flow block, which captures the time between the call coming into an agent and it being accepted/missed/declined.
- **Feature:** Moved the medialess popout page to be an optional feature. Learn how to enable it [here](#)
- **Bug Fix:** Fixed an issue where the `callInteractionDuration` would be too large if the call is missed. It is now defaulted to 0 if the call is not picked up.
- **Bug Fix:** Fixed an issue with the medialess adapter where media was still coming through the adapter and causing audio quality issues. Now, when the medialess option is checked, this will disable the `allowFramedSoftphone` option in CCP config, and media will not be sent through the CCP embedded on Salesforce.
- **Bug Fix:** Fixed an issue where Agents couldn't see some CTI Actions if more than 20 CTI Actions are set up. Now, a scroll bar should appear to navigate to all of them.
- **Bug Fix:** Fixed an issue with the `isInbound` CTI Flow block, which would return false if the Customer hangs up the error before the Agent could answer the call, even if it was inbound.
- **Bug Fix:** Fixed an issue with the `InitialAgentStatus` sub-feature of `SetAgentStatusOnSessionEnd`, which would not follow the `IfProfileNameIncludes` condition.
- **Bug Fix:** Fixed an issue with CCP overlay where if no additional data is added, including Title, Instructions and Fields, the right pointing caret icon will be displayed for detailed form view. Now the execute button will be displayed in this case.
- **Bug Fix:** Fixed an issue with CCP overlay where the order parameter was not affecting the sorting of the CTI Actions in the overlay.
- **Bug Fix:** Fixed an issue with the CCP Element Editor where typing the CTI Action name first caused the cursor to move out of the input box.
- **Bug Fix:** Fixed an issue with the Set Agent Salesforce State CTI Flow block.

5.15 July 2021

When installing v5.15, please **confirm that the application was installed for admins only** (see [installation](#) for more details). If you did this by accident, then you will have to [manually edit the profiles](#) to remove the permissions to the objects and pages created by the app.

- **Feature: Guided Setup** The Guided Setup feature helps make the setup process easier. See [Guided Setup](#) for more details.
- **Feature: Chat Widget Integration for SalesForce Experience Cloud(formerly Community Cloud)** Added VisualForce Page component that allows you to add Amazon Connect Chat

Widget in your Salesforce Experience Cloud Site.

- **Enhancement:** Changed the default audio recording component in the Contact Channel Analytics for easier setup. See [Contact Channel Analytics](#) for more details.
- **Enhancement:** Created the ExecuteAwsService service for simpler communication between Salesforce and AWS. **WARNING:** If you are using Contact Lens for audio recording you *must* replace your existing AwsGenerateAudioRecordingUrl named credential with the ExecuteAwsService named credential. See [here](#) for more details.
- **Bug Fix:** Fixed an issue with the lambda package that caused Contact Lens Call Recording Streaming to be broken for redacted calls.
- **Bug Fix:** Fixed an issue that caused the "Clear All Properties" CTI Flow Block to clear properties important to the CTI adapter working.
- **Bug Fix:** Added the `DISCONNECT` field to the `Initiation Method` field in Contact Trace Records.

5.14 June 2021

- **BugFix:** Added batch processing to CCA Case Trigger and CCA Contact Trigger.
- **Bugfix:** The issue that caused an Attribute label to not display properly in the attributes panel has been fixed.
- **Bugfix:** The issue that caused AC Queue Metrics tab's name showing blank has been fixed.
- **Bugfix:** The issue that caused the Recording Panel button to fail when a url is used for connect instance alias has been fixed.
- **Enhancement:** We now make it possible for voicemail drops to work with queue callbacks.
- **Enhancement:** You can now configure the CT Action Recording Panel's initial state using contact attributes. If you're recording your call, make sure to add an attribute named `RECORDING_STARTED` whose value is `true` in your Contact Flow.
- **Enhancement:** We have added `IfCurrentAgentState` tag to `SetAgentStatusOnSessionEnd` feature, which allows customers to condition this feature on the Agent's current state.

5.13 April 2021

- **Feature: CTI Actions - programmable buttons within the CCP overlay**

In this release, we have added a feature called CTI Action which are programmable buttons for your CTI Flows. Each CTI Action is a button that can be programmed to trigger a CTI Flows whose source value is "CTI Action." In addition, CTI Actions can be programmed to ask the agent for additional

information via a data entry form. You can use the agent's entry in your CTI Flow with the help of "Get Payload" block. This is a great way to ask your agents to enter ad-hoc data prior to running the CTI Flow to provide additional information as part of a workflow to automate case creation, or start a customer refund process. **If you are upgrading from a previous version of the CTI Adapter, please be sure to review the [additional setup steps required for CTI Actions](#).**

- **Feature: CTI Actions: recording API integration within the CCP overlay**

The CTI Adapter now includes integration with Connect's recording API. This feature allows the agent to control when to start and stop recording a call. Once the recording has started, they can also pause and resume it. For example, agents can pause a recording before asking for sensitive information from your customers. Once the agent stops a recording, you cannot start it again. Use pause/resume buttons after you've started recording a call to control the recording.

- **Enhancement: Voicemail Drops (beta)**

The **beta Voicemail Drops** feature now integrates with CTI Actions. In the beta, voicemail drops were loaded directly into the CCP Overlay. As of 5.13, you will need to create a CTI Action, and use the newly added "Leave a Voicemail" block in the CTI Flow where you can configure the specific voicemail drop and the quick connect name to use for the voicemail.

- **Feature: CCP Overlay: Data panel to receive data from CTI Flows.**

You can now send data from a CTI Flow to the CCP Overlay. The Data panel on CCP Overlay will display any object you pass it from "Send Data to CCP Overlay" block.

- **Feature: CTI Flow Blocks: "Start Recording" and "Stop Recording"**

With "Start Recording" and "Stop Recording" blocks, you can control the voice recording of the call within your CTI Flows.

- **Feature: CTI Flow Block: "Update Contact Attributes"**

You can now update contract attributes using CTI Flows. This block accepts a list of key-value pairs and assigns them to the currently active contact. It may come handy for passing Case id and other important information to the next agent when transferring a call.

- **Feature: CTI Flow Block: "Get Payload"**

The **payload** object contains the arguments passed to the CTI Flow. Now you will be able to use "Get Payload" block to reference a payload key as an input in other blocks on your CTI Flow.

- **Feature: CTI Flow Block: "Send Data to CCP Overlay"**

This block allows you to send data to your agent from a CTI flow. The agent will see this information in the CCP Overlay in a panel entitled "Data."

- **Feature: CTI Flow Block: "Leave a Voicemail"**

This block works with the beta Voicemail Drops feature. When you configure the voicemailDropName and quickConnectName, it will pass the contact to an IVR to leave a voicemail on the agent's behalf.

- **Feature: CTI Flow Block: "Get Salesforce Lead ID":** This block allows you to get a Salesforce lead by using a phone number.
- **Enhancement:** "Get Salesforce Contact Id" block now uses FIND syntax to search across multiple fields.
- **BugFix:** For the `SetAgentStatusOnSessionEnd` feature, it would occasionally fail if the agent hadn't interacted with the webpage. We solve this by creating a popout to monitor the agent session.
- **Enhancement:** For the `SetAgentStatusOnSessionEnd` attribute, you can now specify multiple values.
- **Enhancement:** When `SetAgentStatusOnSessionEnd` feature is enabled, you can now configure which state the agent should be shown as when they login with the InitialAgentState setting.
- **Enhancement:** When `SetAgentStatusOnSessionEnd` feature is enabled, you can now configure which agent to logout when all tabs are closed by setting the Status to Logout.
- **Bugfix:** Addressed issue that caused CTI Flows to be run on every open Salesforce tab.
- **Bugfix:** Addressed an issue in "Get Salesforce Contact Id" block that caused the query to fail if the phone number was in E164 format.
- **Enhancement:** Added the onDestroy Event to certain CTI Flow Sources

5.12 March 2021

- **Feature:** Added custom setting which will allow customers to enable and disable non-essential triggers (They are disabled by default now). [More details in the troubleshooting section](#)
- **Bugfix:** Addressed additional trigger issue that prevented orgs with 200k+ CCA records from updating Case and Contact records.

- **Bugfix:** Addressed issue where AC Permission sets did not include the CustomerEndpointAddress field for the ContactChannelAnalytics object.
- **Bugfix:** Addressed issue where AC Permission sets did not include the MedialessPopout page.

5.11 March 2021

- **Bugfix:** Addressed trigger issue that prevented community and partner users from updating Contact and Case records.

5.10 February 2021

- **Feature:** *Contact Control Panel (CCP) Audio Device settings option.* Admins can toggle Phone type settings and the new [Audio Devices settings](#) for agents to see on their CCP. [Audio Device settings](#) allow the agents to choose audio devices for their speaker, microphone, and ringer.
- **Feature:** *Custom Ringtone for chat.* Admins can configure a custom ringtone for chat (separate from CCP) from the CTI Adapter configuration page.
- **Enhancement:** The Salesforce built-in Cross Site Request Forgery (CSRF) protection is enabled for Visualforce pages in the CTI Adapter package which improves organizational security to protect against cross site request forgeries.
- **Bugfix:** Decision blocks no longer requires both sockets to be connected.
- **Bugfix:** Click to Dial stopped working after first use until the agent refreshed the page.
- **Bugfix:** Error that prevented Contact Lens app resources from being hosted on a different domain than the Salesforce instance.
- **Bugfix:** Error that prevented Contact Lens app from displaying intermittently when Transcribe was enabled.
- **Bugfix:** Changed the logic for the IsContactTransfer CTI Flow Block which always returned true.
- **Bugfix:** Medialess popout not closing after Salesforce tabs are closed.
- **Bugfix:** Login window did not close automatically after logging into Connect.
- **Bugfix:** Unable to upgrade the package if the Case or Contact object contained encrypted fields.

5.9 December 2020

- **Feature:** Contact Lens Integration
- **Feature:** Tasks Integration - Added the Amazon Connect Task Contact as a source to CTI Flow in addition to Task specific events

- **Feature:** CTI Block - Is Task Contact? - Check if the contact is a task
- **Feature:** CTI Block - Create Task Contact - Creating a new task contact with certain inputs.
- **Feature:** CTI Block - Pop Task Contact's Reference Urls - Pop any reference urls that are related to the task contact
- Upgraded Salesforce API to v50.0.
- **Feature update:** If you have CCP open on multiple tabs, CTI Flows will be executed only on one of them. The execution will be performed on the current tab, by default. If the agent is currently looking at a different site, a random tab will be selected to perform the execution.
- **Enhancement:** \$User.ProfileId is now available through "userProfile" property.
- **Enhancement:** CTI Flow execution timeout window has been increased to 60 seconds.
- **Feature update:** When the CCP popout is opened, we now ask for a confirmation before refreshing or closing the tab that opened it. Note that if you do close the original tab, the pop out might also be closed.
- **Bugfix:** Voicemail Drops feature has been fixed.
- **Bugfix:** CTI Flow "Open Subtab" block has been fixed.

5.7 November 2020

- **Feature update:** Change audio recording feature in the Contact Channel Analytics page to use an audio streaming approach. Please review the updated [Contact Channel Analytics](#) section for the setup details.
- **Feature:** Add permission set specifically for the audio recording feature
- **Feature:** Localization into 9 languages.
- **Feature:** Add callType to return fields of "Get Contact Properties" block
- **Feature:** Add formatted phone number to return fields of "Get Contact Properties" block
- **Feature:** Add script name to CTI flow definition file.
- **Feature:** Remove context from log outputs
- **Bugfix:** Return field of "Open Primary Tab" was value, not id, as specified. We now provide it in both `value` and `id` fields for backward compatibility.
- **Feature:** Make the error message shown when the execution runs too long more informative.
- **Feature:** Make sure the attributes overlay doesn't open automatically when CCP is opened.
Documentation: "Create and pop that task" default flow is fixed.
- **Bugfix:** update return value of "Get Agent Configuration" block to match the documentation.
- **Feature:** Increase CTI Flow timeout to 10 seconds.
- **Bugfix:** remove the leading wildcard matcher in "Get Salesforce Contact Id" block query. The wildcard matcher caused performance issues with the query. Going forward make sure the

phone number is an exact match to the one in file.

- **Bugfix:** Ensure "Join Strings" block does not ignore boolean false values.
- **Bugfix:** Ensure "Log to Console" block does not ignore boolean false values.
- **Feature:** Add uid field on top of the block on the canvas.
- **Bugfix:** Remove the loginWindow object from log output because it errors with "Cannot convert object to primitive value."
- **Bugfix:** ContactChannel object updates to new agent if previous agent rejected or missed a contact
- **Bugfix:** Changing status to logout now correctly logs agent out
- **Feature:** Rename "Enable Click to Dial?" to "Can Make Outbound Calls?".
- **Feature:** CTI Flow Block - math function - "Multiply"
- **Feature:** CTI Flow Block - math function - "Divide"
- **Feature:** CTI Flow Block - "Get Tab Object Map"
- **Feature:** CTI Flow Block - "Close Salesforce Tab"
- **Feature:** CTI Flow Block - "Delay"
- **Feature:** CTI Flow Block - "Get Primary Tab Ids"
- **Feature:** Improve browser log formatting.
- **Feature:** CTI Flow Block - "Get Tabs With Matching Url"
- **Feature:** *Update Connect agent status when all Salesforce tabs are closed:* You can set the agent status to a specific state if the SetAgentStatusOnSessionEnd feature is turned on and the agent's routing profile name includes the value of IfProfileNameIncludes setting, such as "On-Call." By default, the agent status is set to "Offline" if the feature is enabled and nothing is specified for IfProfileNameIncludes. If this feature is enabled, the agent will be automatically shown as available when they login to Salesforce and the CCP.
- **Feature:** CTI Flow Block - Length"
- **Feature:** CTI Flow Block - "Slice"
- **Feature:** CTI Flow Block - "Cast a Value to a Type"
- **Bugfix:** Agent is able to accept calls when Medialess is turned on.
- **Feature:** CTI Flow Block - "Get CCP Logs" Remove "Initialization" and "Browser" sources
- **Feature:** Allow users to specify Amazon Connect Instance url in CTI Adapter details in addition to Amazon Connect Instance Alias

5.5 October 2020

- **Feature:** CTI Flow Block - "Clear All Properties"
- **Feature:** CTI Flow Block - "Unset Property"

- **Feature:** CTI Flow Block - "Show All Attributes"
- **Bugfix:** Attributes panel can now display attributes of transferred contacts.

5.4 Late September 2020

- **Feature:** You can now provide additional ad-hoc fields to "Create a Task" block. (Note: the values of these fields don't have a lookup dropdown yet.)
- **Feature:** New CTI Block! - You can now create "counters" with the "Update Counter" and read the value of your counters using "Get Counter" block.
- **Feature:** You can now get the number of open tabs from `openAgentTabs` counter.
- **Feature:** You can now compare multiple things using "Is One Of?" block in CTI Flows.
- **Feature:** New CTI Block! - You can now extract a value from a complex value, such as an array or an object, using the "Extract Value" block. (This comes handy when you retrieve a Salesforce object.)
- **Feature:** New CTI Block! - You can use the Salesforce retrieve API to fetch a record from the server by id using "Retrieve Salesforce Record" block.
- **Feature:** New CTI Block! - You can use the "Get Salesforce Contact Id" to fetch the id of a Salesforce contact by its phone number.
- **Feature:** New CTI Block! - You can now show a window alert using "Alert" block.
- **Feature:** New CTI Block! - You can now use create a complex string using string templates and multiple variables with the help of "String Template" block.
- **Bugfix:** When a screenpop is "deferred," the CTI Block used to return an inexact match and the Id field in the return value of the block would be blank. This issue has been fixed in this release.
- **Bugfix:** Presence sync is working again. The current release also reduces the wait threshold between each presence sync update from 1 second to 100ms, i.e. co-occurring events won't get lost anymore (as much).
- **Bugfix:** The encoding issue affecting "SOQL Block" has been fixed. The single quotes in the SOQL query are no longer encoded as HTML entities.
- **Bugfix:** To access the return value of another block, power users use "magic strings," e.g. `\$.actions.<blockId>.results.<fieldName>`, but these strings used to be cleared in the UI when the block is selected on the canvas. This issue is now fixed.
- **Bugfix:** The spelling of `TaskSubtype` field in "Create a Task" block has been fixed. Your TaskSubtype won't get lost anymore.
- **Bugfix:** Call recording view for a Case has been fixed.
- **Bugfix:** "Is Contact Inbound?" block is working again.
- **Bugfix:** "Is Truthy?" block now works with boolean input values.
- **Bugfix:** Salesforce UI onNavigationChange event listener is working again.

- **Bugfix:** We now alert you to change your instance alias if you try to sign in with instance alias set to "default."

5.3 September 2020

- **Bugfix:** Fix the issue that caused ACSFCCP_CallRecordingTask component to not work.

5.1 Late August 2020

- **Bugfix:** Ensure "Get App View" CTI Flow block doesn't break the sidebar
- **Enhancement:** Add "queueARN" field to "Dial Number" CTI Flow block
- **Bugfix:** Ensure some required CTI Flow block fields are not shown as "optional"
- **Bugfix:** Ensure "Save (or Create) a Record" block works as expected
- **Bugfix:** Fix the validation error on "CallDurationInSeconds" field in "Create a Task" block
- **Bugfix:** Fix phantom scrollbar on Windows machines
- **Bugfix:** Fix issue where copying contact attributes to clipboard doesn't work
- **Bugfix:** Fix issue where "saveLog" CTI Flow block throws an error
- **Bugfix:** Fix issue with onOffline Flow event not firing
- **Bugfix:** Fix various omnichannel presence sync bugs
- **Bugfix:** Ensure the CCP default dimensions are adjusted to CCPv2 defaults
- **Feature:** Add block "Set Agent Status By Name on Connect."

5.0 August 2020

- **This release has new features and updates:** Please test and validate version 5.0 in your Salesforce sandbox before upgrading this in production.
- **CTI Flows:** CTI Flows replace Lightning CTI Extensions in allowing customers to build their agent for Lightning and Classic via a drag drop UI. Many of the CTI blocks are similar to the Lightning CTI Extension script API calls and can be mapped similarly. Lightning CTI Extension scripts are NOT automatically migrated to CTI Flows. When upgrading the with existing scripts, it will give you the option to download the existing script for reference before building your CTI Flows. We strongly recommend you validate this install/upgrade in a test environment and fully test the CTI Flows against your previous scripts functionality. Please open a support ticket if there is additional functionality you require from your current scripting implementation.
- **Security Profile improvements:** AC Administrator, AC Agent, and AC Manager permission sets to enforces objects access and fields level (FLS) as per Salesforce security guideline for

managed package. To Amazon Connect Objects and fields, user should either one of Amazon Connect permission sets AC Administrator, AC Agent, and AC Manager.

- **Attributes:** Amazon Connect CCP (Contact Control Panel) in Lightning Classic now display an overlay for showing attributes consistently.
- **AWS Secrets Manager** support for storing Salesforce credentials.
- **VPC Support:** ability to place Lambdas in VPC
- **New Salesforce API integration:** Exposed new operations in sfinvokeapi read or create Salesforce records(query queryOne, createChatterPost, createChatterComment, lookup_all, delete)
- **Upgrade:** Amazon Connect Streams API bumped up to version 1.5.
- **Bugfix:** Task creation issue for non connect users - Fixed task trigger apex code, added a validation before security access check for Amazon managed package objects
- **Bugfix:** Contact interaction fixed.
- **Other minor bugfixes and improvements**

4.5 April 2020

- **This release has new features and updates:** Please test and validate version 4.5 in your Salesforce sandbox before upgrading this in production.
- **Installation / Configuration:** AC_Administrator permission set has been added to manage CTI Configuration in addition to AC_Manager and AC_Agent. See documentation for further information.
- **API:** Updated support for CCPv2 in Classic/Console. See documentation for Call Center settings.
- **Bugfix:** Updated attribute display to resolve duplicated attributes.
- **Security:** Improved control access at the object-level, the record-level, and at the field level.

4.4 March 2020

- **This release has significant new features and updates:** Please test and validate version 4.4 in your Salesforce sandbox before upgrading this in production.
- **Documentation:** Guide has been rewritten and restructured based on feedback.
- **Installation / Configuration:** Improved installation and configuration guide
- **Installation / Configuration:** Added Enhanced Agent Logout functionality to Lightning.
- **API:** Updated to the latest Amazon Connect Streams and Chat libraries
- **API:** Additional extensibility methods provided

- **Setup:** Improved Presence Sync Rule editor
- **Setup:** CTI Adapter validation is performed upon initialization and will inform the user of common misconfigurations.
- **Setup:** Additional CTI Script examples are provided.
- **Setup:** The ability to place the lightning transcript view on Task, Contact Channel, and Contact Channel Analytics object has been added.
- **Bugfix:** Updated allowlisting steps to address login popup issue.
- **Bugfix:** OmniChannel workload data not being usable has been resolved
- **Bugfix:** CTI Attribute issue when processing multiple pieces of contact attribute data has been resolved.
- **Bugfix:** The call transcript now scrolls within a fixed region rather than consuming vertical space.
- **Bugfix:** Finding Task Record in Classic/Console fixed.
- **Security:** The ability to create, update, and delete AC_CtiAdapter, AC_CtiScript, AC_CtiAttribute and AC_PresenceSyncRule records has been removed from the AC_Agent permission set.

4.2 December 2019

- **This release has significant new features and updates:** Please test and validate version 4.2 in your Salesforce sandbox before upgrading this in production.
- **Installation / Configuration:** Improved installation and configuration guide
- **API:** Lightning CCP Extension scripts and reference guide
- **Setup:** A default CTI adapter and scripts for click-to-dial, voice contact pop, and chat contact pop are not included in the base installation.
- **Editor:** A more robust script editor is included for use in CTI adapter / script configuration.
- **Bugfix:** SSO issue has been resolved

4.1 November 2019

- **This release has significant new features and updates:** Please test and validate version 4.1 in your Salesforce sandbox before upgrading this in production. As we look to simplify documentation, this release introduces a new [Amazon Connect CTI Adapter v4 for Salesforce Lightning](#) setup and installation guide. Please review this setup guide in detail to see all the latest changes for Lightning CTI Adapter installations.
- **Classic and Console CTI setup guide:** Please use the [Amazon Connect CTI Adapter v4 for Salesforce Classic](#) setup and installation guide for Classic and Console CTI Adapter

installations.

- **Amazon Connect Chat and Contact Control Panel (CCP) v2:** support for Amazon Connect chat and integration of CCP v2. CCP v2 is required for Lightning CTI Adapter installations. CCP v1 is still supported for Classic / Console CTI Adapter installations.
- **Historical and Real-Time Reporting:** updated historical metric functionality with additional metrics and dashboards. Added real-time metrics and dashboards. This functionality requires an update of AWS Serverless Lambda functions for Salesforce.
- **Lightning CCP Extensions and configuration:** We have revamped the approach for the Call Center config and have added a new AC CTI Adapters Lighting config page.
- **High Velocity Sales:** CTI Adapter integration supported for Salesforce High Velocity Sales product.

Key Benefits and Requirements

The key benefits of the Amazon Connect CTI Adapter are:

- **Amazon Connect Voice and Chat:** ability to take voice and chat calls in the salesforce agent experience and advanced screen pop on the incoming phone number, case, account or contact. Agents can also click to dial a number within their contacts.
- **Single Sign-On support:** seamless login with Connect and Salesforce with any standard SAML 2.0 provider.
- **Call disposition and activity management:** configure post call workflows to support your Agent's after call work.
- **Call logging and recording:** Voice and chat interactions can be logged as Salesforce activities and Amazon Connect call recordings can be played within the Salesforce.
- **Omnichannel Presence Sync:** enable Salesforce chat, sms and email to share presence with Amazon Connect. Amazon Connect will know when an agent is handling a Salesforce chat and make them unavailable for a voice call, and vice versa.
- **CTI Flows:** easily customize and extend behaviors within the CTI Adapter such as screenpop and activity management. Default flows along with the API guide provide key examples.
- **High-velocity sales (HVS):** using Salesforce HVS, enable your inside sales team to follow a repeatable pre-define sales cadence for your business. It enables sales managers and reps to work on prioritize list of prospects and follow best sequence of sales outreach activities defined by your sales process.

The key benefits of the AWS Serverless Application Repository for Salesforce are:

- **Access Salesforce Data:** easily inject salesforce data into the customer experience. Businesses can offer personalized greetings and dynamic routing based on customer information, create new objects, update existing records, and delete items based on customer choices in the IVR.
- **Contact center real-time reports:** display real-time contact center metrics within Salesforce from Amazon Connect.
- **Contact center historical reports:** display historical contact center metrics within Salesforce from Amazon Connect.
- **Contact analytics:** transcribe voice calls and perform analysis of the conversations using AI to surface sentiment, keywords, syntax, entities, etc.

We recommend that you initially install and configure the package into your Salesforce sandbox. This will allow you to test the integration, become more familiar with it, and modify it to your needs prior to deploying it to your production org.

If you are using Lightning, you can get a head start by working through the [Build an Amazon Connect Integration Salesforce Trailhead](#).

Requirements

To successfully deploy, configure, and implement the Amazon Connect integration with Salesforce, you must ensure that the following requirements and prerequisites are in place before.

Prerequisites - Amazon Connect CTI Adapter

In order to successfully install and configure the Amazon Connect CTI Adapter from the AppExchange you will need:

1. Salesforce
 - a. Salesforce org with Lightning experience
 - b. My Domain configured and deployed to users
2. An Amazon Connect instance
3. SAML Details (If using SAML)

Prerequisites - AWS Serverless Application Repository for Salesforce

In order to successfully install and configure the Salesforce functions from the Serverless Application Repository, you will also need:

1. A Kinesis stream configured for your Amazon Connect contact trace records (CTRs)
2. Salesforce:
 - a. An API user account
 - b. A new Connected App

Browser Compatibility

Amazon Connect requires WebRTC to enable soft-phone voice media stream and Websockets to enable soft-phone signaling. Consequently, users are required to use the latest version of either Google Chrome or Mozilla Firefox. For more information, please see the [Amazon Connect documentation](#).

Salesforce Lightning Support

Please note that following features are currently not supported in Salesforce Lightning:

- Outbound Campaign Calls using Salesforce Omni can be routed to the agent, but the automated screen pops and the dialing of the phone number will not work. The agent will have to click on the record links to open the records and use Salesforce's Click-to-Dial feature to make the phone call.
- Lightning Standard Navigation is not currently supported in App Options for the Amazon Connect CTI Adapter.

Installing the CTI Adapter and Salesforce Lambdas

Amazon Connect Salesforce CTI Adapter Managed Package

The Amazon Connect CTI Adapter for Salesforce provides the core integration between the two platforms. It embeds the Amazon Connect Contact Control Panel into Salesforce which provides telephony control as well as access to event data coming from Amazon Connect. Using this adapter, you can configure screen pops based on customer data, automate contact center telephony functions like click-to-dial, and establish presence syncing rules for integration with Salesforce Omni-Channel. This is the base of the integration.

The first step in the deployment of the integration is to install the Amazon Connect CTI Adapter managed package from the AppExchange Marketplace.

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find**, type **AppExchange** (the results will populate without hitting enter)
3. Select **AppExchange Marketplace** from the links provided
4. In the AppExchange window, enter **Amazon Connect** into the **Search AppExchange** field and press enter
5. In the **Search Results**, select **Amazon Connect CTI Adapter**

[<> BACK](#)

Search Results for "amazon connect"

40 Apps · Sorted by Relevance



Amazon Connect

Easy to use omnichannel cloud contact center

(37) FREE



CTI CON...

FOR SA...

ON-PREMIS...

MS-TEAMS • A...

CTI Data Con...

(44)

6. On the **Amazon Connect CTI Adapter** detail page, select **Get It Now**

< BACK

Amazon Connect CTI Adapter: CTI | Contact Center | IVR | ACD | Call Recording

by Amazon Web Services

Bring the Power of Intelligent CTI to Salesforce Service Cloud



★★★☆☆

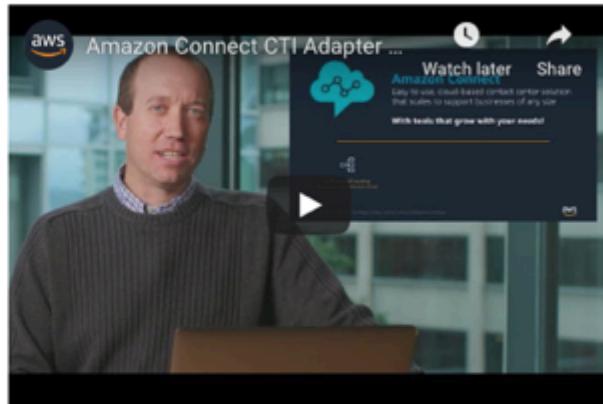
Free



DETAILS

REVIEWS

PROVIDER



Amazon Connect CTI Adapter for Salesforce Overview and Demo



Highlights

Setting up Amazon Connect is easy. With only a few clicks in the AWS Management Console, agents can take calls within minutes. The adapter...

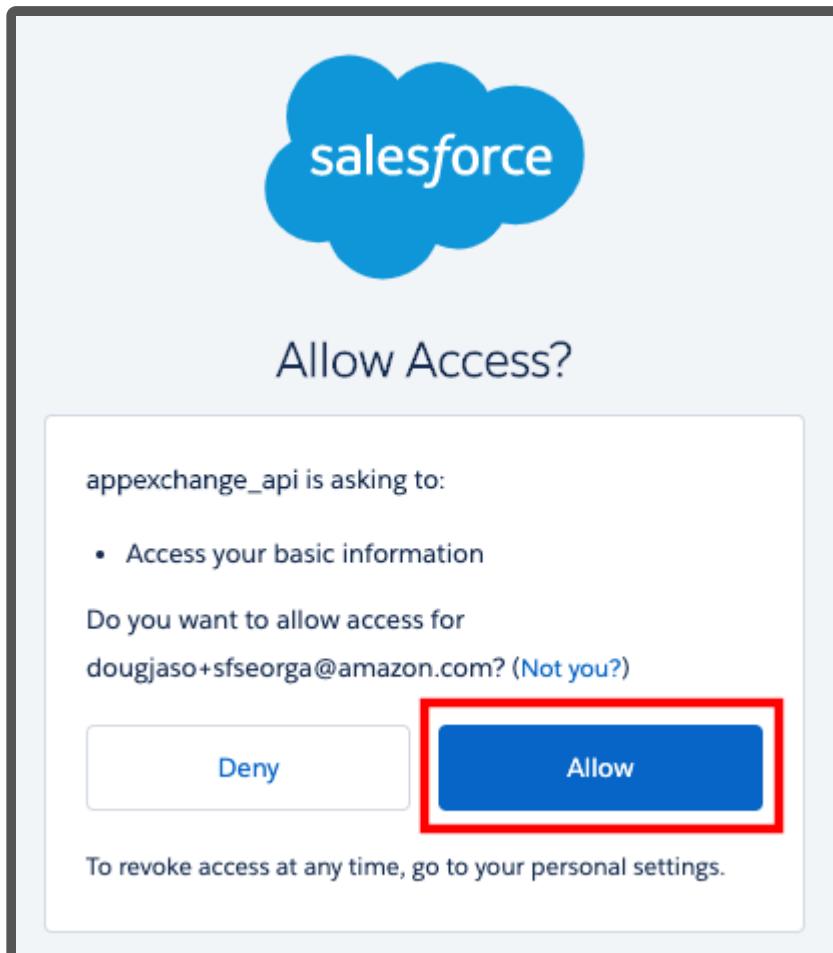
Contact Information

<https://aws.amazon.com/contact-us/>

Watch Demo

↓ Get It Now

7. If you are presented with the Log In to AppExchange screen, select **Open Login Screen**. You should then be presented with an Allow Access Screen. Choose **Allow**



8. On the **Where do you want to install Amazon Connect CTI Adapter** page, choose the **Install Here** button in the Install in This Org section

Where do you want to install Amazon Connect CTI Adapter: CTI | Contact Center | IVR | ACD | Call Recording?

Before you install in a production org, we recommend testing in a sandbox first.

Install in This Org

Get going in the org where you're logged in right now.

Install Here

Install in a Sandbox Org

Test in a copy of a production org.

Install in Sandbox

Cancel

9. On the **Confirm installation details** screen, fill out the **Tell us about yourself** form, check the box to **agree with the terms and conditions**, and optionally select the box to **allow the provider to contact you**. Then select **Confirm and Install**

I have read and agree to the [terms and conditions](#).

Salesforce.com Inc. is not the provider of this application but has conducted a limited security review. Please [click here](#) for detailed information on what is and is not included in this review.

Allow the provider to contact me by email, phone, or SMS about other products or services I might like

Cancel

Confirm and Install

10. Select **Install for Admins Only**, then choose **Install**. **THIS SELECTION IS VERY IMPORTANT** - if you select the wrong option, then standard users may have access to objects and pages that they shouldn't have access to.



Install Amazon Connect - Universal Package

By

Install for Admins Only

Install for All Users

Install for Specific Profiles...

Install

Cancel

11. The CTI Adapter will take some time to install. While it installs, you will be presented with the **This app is taking a long time to install screen.**

12. Choose **Done**.

aws Install Amazon Connect CTI Adapter: CTI | Contact Center | IVR | ACD | Call Recording

By Amazon Web Services



This app is taking a long time to install.

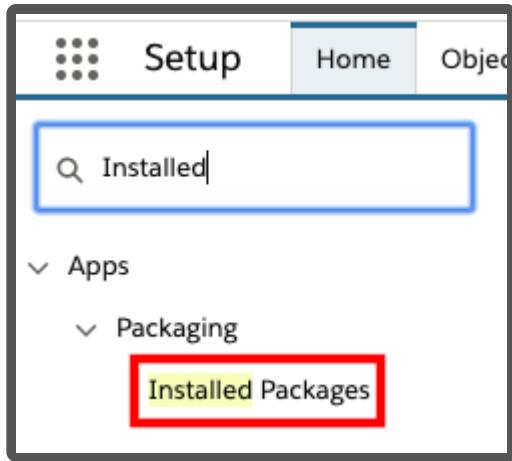
You will receive an email after the installation has completed.

Done

13. Once you receive confirmation that the **installation has completed** via email, return to the browser

14. Close the **Amazon Connect CTI Adapter** detail page (if still open)

15. In Quick Find, enter **Installed**, then select Installed Packages from the result



16. Once the **Installed Packages** page opens, validate that the **Amazon Connect -- Universal Package** is installed

A screenshot of the 'Installed Packages' page in Salesforce. The page title is 'Installed Packages'. It includes a help link 'Help for this Page' and a 'Visit AppExchange' button. The main content area shows a table of installed packages. One row is selected, showing the package name 'Amazon Connect - Universal Package', publisher 'Amazon AWS', version '4.2', namespace prefix 'amazonconnect', install date '1/21/2020, 10:42 PM', and status 'Passed'. The table has columns for Action, Package Name, Publisher, Version Number, Namespace Prefix, Install Date, Limits, Apps, Tabs, Objects, and AppExchange Ready.

Amazon Connect Salesforce Lambda package

The Amazon Connect Salesforce Lambda package adds considerable capability to the integration. It includes data connectivity between Amazon Connect and Salesforce for typical tasks like lookups, case creation, and updates. Additionally, it adds new features like real-time and historical data imports, contact trace record imports, recording import, transcription, and contact analytics functions. These capabilities are configurable and can be activated or deactivated on a call-by-call basis.

The Amazon Connect Salesforce Lambda package is delivered via the AWS Serverless Application Repository. The AWS Serverless Application Repository enables you to quickly deploy code samples, components, and complete applications. Each application is packaged with an AWS Serverless Application Model (SAM) template that defines the AWS resources used. There is no additional charge to use the Serverless Application Repository - you only pay for the AWS resources used in the applications you deploy.

1. In a new browser tab, login to the [AWS console](#)
2. Make sure you are in the same region as your Amazon Connect instance
3. Once you have selected the region, navigate to the [Amazon Connect Console](#)

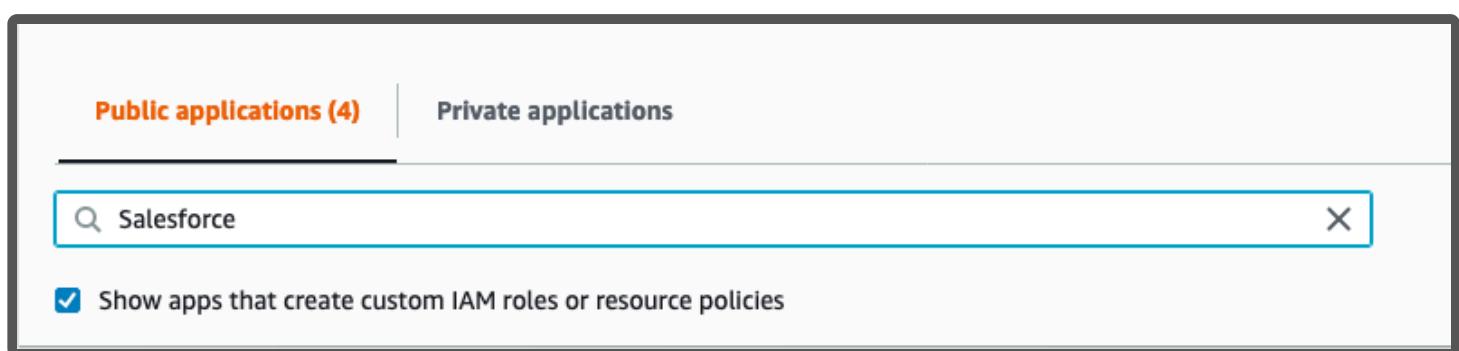
4. Verify that the Amazon Connect instance that you wish to configure is listed

5. Once you have verified your Amazon Connect instance, Open the [Serverless Application Repository Console](#)

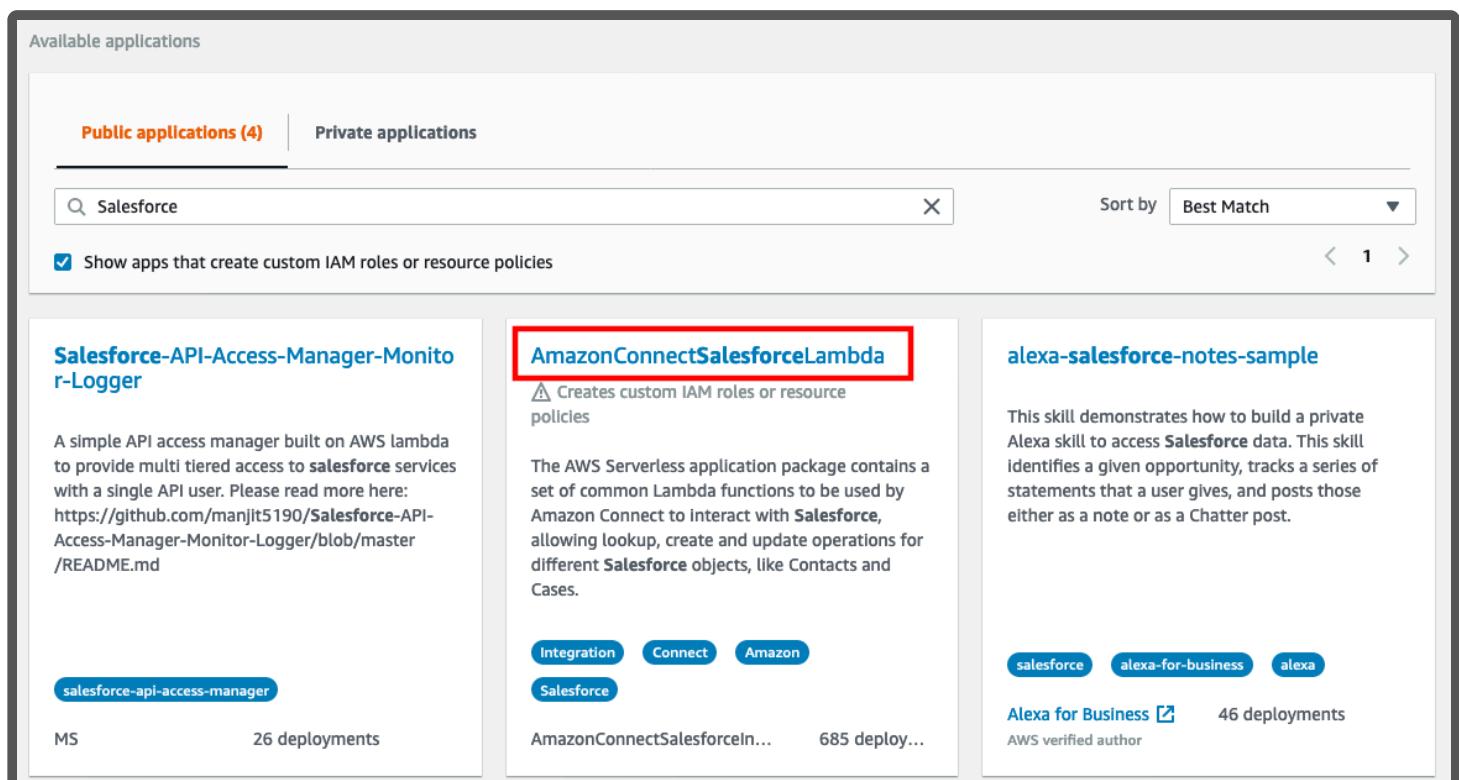
6. In the left navigation, select **Available Applications**



7. In the search area, make sure that **Public applications** is selected, check the box for **Show apps that create custom IAM roles or resource policies**, and enter **Salesforce** in the search field, this will automatically filter the available packages



8. Select AmazonConnectSalesForceLambda



9. When the Application loads, scroll down to the **Application settings** section
10. If you would like to use the Guided Setup feature, **don't change any parameters in the template** and select **Deploy**, and wait for the stack to finish deployment. Then, follow the section below on setting up the ExecuteAwsService named credential. If you are not using the Guided Setup feature, navigate to [here](#) for manual setup instructions (skipping the rest of the instructions on the page).

Deployment status for serverlessrepo-SFConsolidatedLambdaPackage

[Create a new app](#) [Test app](#)

 Your application has been deployed
Review the application's README for what to do next.

[Permissions](#) [Resources](#) [View CloudFormation Stack](#)

Setting up the ExecuteAwsService Named Credential

The ExecuteAwsService Named Credential is the entrypoint for the CTI Adapter to communicate with your AWS account. The Apex code uses the Named Credential to call the `sfExecuteAwsService.py` lambda, which uses boto3 to make changes in and retrieve data from your AWS account. Setting up this Named Credential is **not required** if you do not wish to use the features that rely on it (Guided Setup and Contact Lens). In addition, you can alter the permissions given to the `sfExecuteAwsService` lambda to match your security requirements (NOTE: if you choose to do so, do so after you configure up the lambdas as some permissions are added/removed based on how the lambdas are configured).

Before you create the ExecuteAwsService Named Credential, **confirm that the application was installed for admins only**. If not, then standard users may be able to invoke methods that call named credentials. If you did this by accident, then you will have to [manually edit the profiles](#) to remove the permissions to the objects and pages created by the app.



Install Amazon Connect - Universal Package

By

Install for Admins Only

Install for All Users

Install for Specific Profiles...

Install

Cancel

1. Navigate to the IAM console in your AWS account, select the **Users** tab, and select **Add Users** to create a new user.

The screenshot shows the AWS IAM console. On the left, there's a sidebar with 'Identity and Access Management (IAM)' at the top, followed by 'Dashboard', 'Access management' (with 'User groups', 'Users' highlighted in orange, 'Roles', 'Policies', and 'Identity providers'), and a 'x' icon. The main area has a blue header bar with the text 'Introducing the new Users experience' and 'We've redesigned the Users experience to make it easier to use. Let us know what you think.' Below this is a breadcrumb trail 'IAM > Users'. The main content area is titled 'IAM users (7) Info' with the sub-instruction 'An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.' There's a red box around the 'Add users' button. At the bottom, there's a search bar and a table with columns for 'User name', 'Groups', 'Last activity', 'MFA', and 'Console last sign-in'.

2. Give your IAM user a name (like `sfExecuteAwsServiceiamUser`). For the Access type, select **Programmatic access**. Click Next.
3. Select **Attach existing policies directly**, then search for and select `invokeSfExecuteAWSServicePolicy`.

The screenshot shows the 'Add user' wizard. At the top, there are five numbered tabs: 1, 2 (highlighted in blue), 3, 4, and 5. The current step is 'Set permissions'. It shows three options: 'Add user to group', 'Copy permissions from existing user', and 'Attach existing policies directly' (which is highlighted with a blue border). Below this is a 'Create policy' button and a 'Filter policies' dropdown set to 'sfexecute'. A search bar shows 'sfexecute' and 'Showing 2 results'. The results table has columns for 'Policy name', 'Type', and 'Used as'. One result is shown: a checkbox next to a policy name, followed by 'Customer managed' and 'None'.

4. Click next until the user is created. In the final screen, copy down the **Access Key ID** and the **Secret Access Key**.

Access key ID

Secret access key

5. Next, navigate to the Lambda Console. In the functions tab, search for `sfExecuteAwsService`.

The screenshot shows the AWS Lambda console with the 'Functions' tab selected. A search bar at the top contains the query "sfExecuteAWSservice". Below the search bar, a table lists one function: "sfExecuteAWSservice-", which is highlighted with a red box. The function details show it was created with Python 3.7, 3.8 MB of code size, and was last modified 22 days ago.

6. Copy down the name of the function. Make sure you are not copying any extra characters.

7. Navigate to your setup section of your Salesforce instance, and search for *Named Credentials*.

The screenshot shows the Salesforce Setup interface under the 'Security' section. The 'Named Credentials' tab is selected. A search bar on the left shows the query "named". The main table lists four named credentials: "AmazonConnectAPI", "AmazonConnectAPIAD", "AmazonConnectAPIIDX", and "ExecuteAwsService". The "ExecuteAwsService" row is highlighted with a red box. On the far right of the table, there are two buttons: "New" and "New Legacy", with "New Legacy" being highlighted by a red box.

8. Select **New Legacy**. For the values in the next screen, enter the following:

- **Label:** ExecuteAwsService
- **URL:** `https://lambda.{insert AWS region}.amazonaws.com/2015-03-31/functions/{insert lambda function name (copied above)}/invocations`
- **Identity Type:** Named Principle
- **Authentication Protocol:** AWS Signature Version 4
- **AWS Access Key ID:** Access Key ID copied above
- **AWS Secret Access Key:** Secret Access Key
- **AWS Region:** {insert AWS region}
- **AWS Service:** lambda

Save **Cancel**

Label	<input type="text" value="ExecuteAwsService"/>
Name	<input type="text" value="ExecuteAwsService"/>
URL	<input type="text" value="https://lambda.us-west-2.amazonaws.com/2015-03-31/functions/XXXXXXXXXXsfExecuteAWSservice-1XXXXXXXXXX3/invocations"/>

▼ Authentication

Certificate	<input type="text"/> 
Identity Type	<input type="text" value="Named Principal"/>
Authentication Protocol	<input type="text" value="AWS Signature Version 4"/>
AWS Access Key ID	<input type="text"/>
AWS Secret Access Key	<input type="text" value="....."/>
AWS Region	<input type="text" value="us-west-2"/>
AWS Service	<input type="text" value="lambda"/>

9. Click **Save**.

After following the above instructions, follow [these instructions](#) to navigate to the Guided Setup feature.

Setting Up The CTI Adapter Using Guided Setup

Guided Setup

Provision Amazon Connect Instance?	<input type="checkbox"/>
This setting will provision an Amazon Connect instance in your AWS account. You cannot provision an instance the same time you configure the Adapter or the Lambdas.	
Set up Amazon Connect Salesforce CTI Adapter?	<input type="checkbox"/>
This setting will configure the Salesforce CTI Adapter in your Salesforce instance.	
Set up Amazon Connect Salesforce Lambdas?	<input type="checkbox"/>
This setting will help you set up the Amazon Connect Salesforce Lambdas in your AWS account.	

Next

In order to navigate to the Guided Setup feature, perform the following steps (NOTE: If you are not an admin user then you must first add yourself to the AC_Administrator permission set, see [here](#) for more details):

1. Navigate to the Service Console in your Salesforce instance.
2. Click the drawdown button in the Service Console navigation bar, and select **Edit**.



Service Console

Cases



Cases

Recently Viewed ▾



0 items • Updated a few seconds ago



Case Number



Cases



Contacts



Accounts



Reports



Dashboards



Chatter



Quick Text



Knowledge



Edit

3. In the proceeding popup, select **Add More Items**.

Edit Service Console App Navigation Items

Personalize your nav bar for this app. Reorder items, and rename or remove items you've added.

[Learn More](#) 

NAVIGATION ITEMS (8)

[Add More Items](#)

-  Cases
-  Contacts
-  Accounts
-  Reports
-  Dashboards
-  Chatter
-  Quick Text
-  Knowledge

Reset Navigation to Default 

[Cancel](#)

[Save](#)

4. Click the + button next to **AC Guided Setup**, then add the item and **save**.

5. Select the newly added **AC Guided Setup** button in the drawdown menu.



Cases

Recently Viewed ▾

0 items • Updated 6 minutes ago

**Case Number**

Contacts



Accounts



Reports



Dashboards



Chatter



Quick Text



Knowledge



AC Guided Setup



Edit

Guided Setup Prerequisites

The below sections are linked to from the Guided Setup feature. Only perform the below steps when the Guided Setup feature links to them.

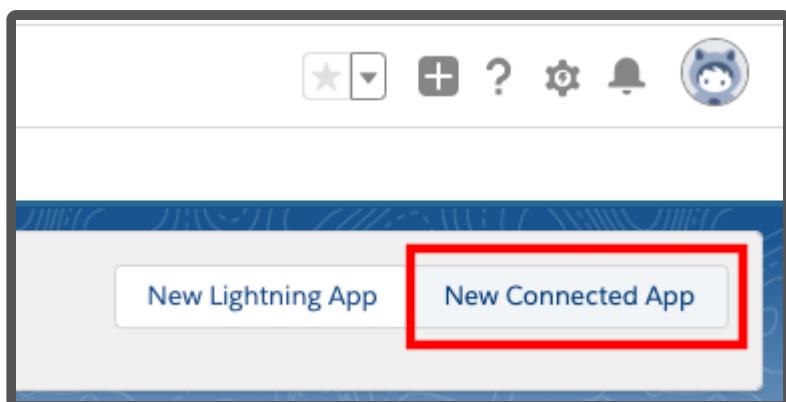
Create Named Credential

See [here](#) for instructions on setting up the Named Credential.

Create Connected App

To get access to the environment, a Connected App must be configured with OAuth settings enabled.

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** field, type **app manager**, then select **App Manager** from the results
3. In the upper right corner, select **New Connected App**



4. On the New Connected App form, enter a name for the Connected App, such as **Amazon Connect Integration** and press tab. This will populate the API Name automatically. Then provide a contact email address

New Connected App

Save Cancel

Basic Information

Connected App Name	Amazon Connect Integration
API Name	Amazon_Connect_Integration
Contact Email	dougjaso@amazon.com

5. Select the checkbox to **Enable OAuth Settings**

▼ API (Enable OAuth Settings)

Enable OAuth Settings

6. Set the **Callback URL** to your domain url. Find the domain at *Setup -> My Domain*.

Enable OAuth Settings

Enable for Device Flow

Callback URL https://[REDACTED].my.salesforce.com

7. In the Selected OAuth Scopes section, select the following and add them to the Selected OAuth Scopes:

8. Access the identity URL service (id, profile, email, address, phone)

9. Manage user data via APIs (api)

10. Select the checkbox for Require Secret for Web Server Flow, and the checkbox for Require Secret For Refresh Token Flow

11. The API (Enable OAuth Settings) section should now look like this

The screenshot shows the 'API (Enable OAuth Settings)' configuration page. The 'Selected OAuth Scopes' section contains two items: 'Access the identity URL service (id, profile, email, address, phone)' and 'Manage user data via APIs (api)'. The 'Available OAuth Scopes' list is expanded and shows the following options: Access Analytics REST API Charts Geodata resources (eclair_api), Access Analytics REST API resources (wave_api), Access Connect REST API resources (chatter_api), Access Lightning applications (lightning), Access Visualforce applications (visualforce), Access chatbot services (chatbot_api), Access content resources (content), Access custom permissions (custom_permissions), Access unique user identifiers (openid), and Full access (full). There are 'Add' and 'Remove' buttons between the lists. At the bottom of the page, the 'Require Secret for Web Server Flow' and 'Require Secret for Refresh Token Flow' checkboxes are checked.

12. Select **Save** at the bottom of the screen.

13. Select **Continue** on the New Connected App page

14. You should now be at the new app's page

15. Copy the value for **Consumer Key** to your notepad

16. Select **Click to reveal** next to Consumer Secret and copy the value to your notepad

17. At the top of the detail page, select **Manage**

18. On the Connected App Detail page, select the **Edit Policies** button

19. Set Permitted Users to **Admin approved users are pre-authorized** and choose OK on the pop-up dialog

20. Set IP Relaxation to **Relax IP restrictions**

21. The OAuth Policies section should now look like the following



22. Select **Save**

Guided Setup Additional Instructions

The below sections are linked to from the Guided Setup feature. Only perform the below steps when the Guided Setup feature links to them.

Retrieve Amazon Connect Instance Url

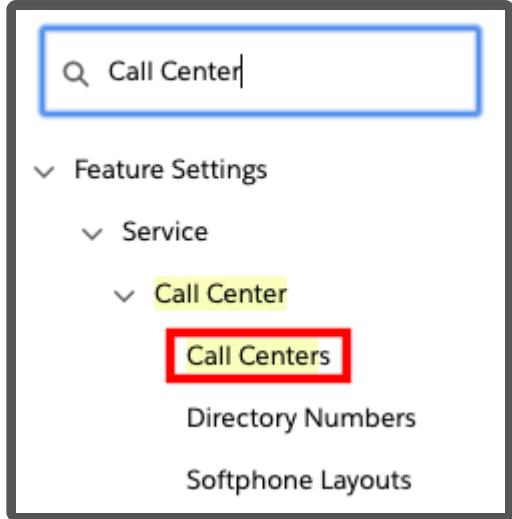
1. Navigate to the [Amazon Connect Console](#)
2. Select your Instance Alias
3. On the Overview page for your instance, copy the Login URL (if your Amazon Connect instance uses the `https://(instancename).awsapps.com/connect/login` domain, then remove everything after ".com"):

The screenshot shows the 'Account overview' page. Under the 'Access information' section, it displays the 'Access URL' as <https://guidedsetuptest-instance-w3dgh2.my.connect.aws>.

Add users to the Call Center

1. Log in into your Salesforce org and go to **Setup**

2. In the **Quick Find** field, enter **Call Center**, then select **Call Centers** from the result list



3. If you see the **Say Hello to Salesforce Call Center** page, select **Continue**

4. Select **AC Lightning Adapter**

The screenshot shows the "All Call Centers" page. At the top, it says "All Call Centers" and defines what a call center is. Below is a table with columns "Action" and "Name". The first row shows "Edit | Del" followed by "AC Lightning Adapter", which is highlighted with a red box. The second and third rows show "Edit | Del" followed by "Amazon Connect CCP Adapter Classic 3.11" and "Amazon Connect CCP Adapter Console 3.11" respectively.

Action	Name ↑
Edit Del	AC Lightning Adapter
Edit Del	Amazon Connect CCP Adapter Classic 3.11
Edit Del	Amazon Connect CCP Adapter Console 3.11

5. On the **AC Lightning Adapter** detail page, select **Edit**

6. On the **AC Lightning Adapter: Manage Users** page, select **Add More Users**.

7. Set filters (if desired) and then choose **Find**.

8. Select the checkbox next to the user to add, then choose **Add to Call Center**.

The screenshot shows the "Add to Call Center" dialog box. It has fields for "Full Name" (Douglas Jason), "Alias" (JDoug), "Username" (redacted), "Role" (System Administrator), and "Profile" (Analytics Cloud Integration User). There are checkboxes for "User_Integration" and "User_Security", with "User_Integration" checked and highlighted with a red box. At the bottom right are "Add to Call Center" and "Cancel" buttons.

9. Repeat the steps to add more users.

Add users to a Permission Set

All users must be assigned the required permission set to access Salesforce metadata. The Amazon Connect CTI Adapter includes Permission Sets-- one for agents, one for managers, one for administrators, and a few for specific features, that grant users the appropriate access for their role. More information on assigning user permissions can be found in the [Salesforce help documentation](#).

1. Log in into your Salesforce org and go to **Setup**
2. In **Quick Find**, enter **Permission** and select **Permission Sets** from the results
3. Choose **AC_Administrator**, **AC_Agent** or **AC_Manager** as appropriate for the user(s)

Permission Sets

On this page you can create, view, and manage permission sets.

In addition, you can use the SalesforceA mobile app to assign permission sets to a user. Download SalesforceA from the App Store or Google Play: [iOS](#) | [Android](#)

All Permission Sets [Edit](#) | [Delete](#) | [Create New View](#)

Action	Permission Set Label	Description
Del Clone	AC Administrator	Allows the user to configure Amazon Connect setup and provides full access to Am...
Del Clone	AC Agent	
Del Clone	AC Manager	

4. Choose **Manage Assignments**.
5. Choose **Add Assignments**.
6. Select the users to assign the permissions, then choose **Assign**.

Assign Users
All Users

[View](#) [All Users](#) [Edit](#) | [Create New View](#)

[Help for this Page](#)

Action	Full Name	Alias	Username	Last Login	Role	Active	Profile	Manager
Edit	Chatter Expert	Chatter	[REDACTED]@chatter.salesforce.com		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Chatter Free User	
Edit	Douglas Jason	iCloud	[REDACTED]	1/21/2020, 10:40 PM	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	System Administrator	
Edit	User Integration	integr	[REDACTED]		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Analytics Cloud Integration User	
Edit	User Security	sec	[REDACTED]		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Analytics Cloud Security User	

7. Repeat these steps as needed for all users

AC_Administrator

Object Name	Object Permissions	Total Fields	Tab Settings
AC Agent Performance	Read, Create, Edit, Delete, View All, Modify All	124	--
AC CCP Overlay Elements	No Access	9	--
AC Contact Channel Analytics	Read, Create, Edit, Delete, View All, Modify All	31	Visible
AC Contact Channels	Read, Create, Edit, Delete, View All, Modify All	24	--
AC Contact Trace Records	Read, Create, Edit, Delete, View All, Modify All	50	Visible
Accounts	No Access	25	--
AC CTI Adapters	Read, Create, Edit, Delete, View All, Modify All	22	Visible
AC CTI Attributes	Read, Create, Edit, Delete, View All, Modify All	11	--
AC CTI Scripts	Read, Create, Edit, Delete, View All, Modify All	10	--
AC Events	No Access	--	--
AC Features	Read, Create, Edit, Delete, View All, Modify All	6	--
AC Guided Setup	--	--	Visible
AC Historical Queue Metrics	Read, Create, Edit, Delete, View All, Modify All	119	--
AC Phone Calls	No Access	22	--
AC Presence Sync Rules	Read, Create, Edit, Delete, View All, Modify All	13	--
AC QueueMatrices	No Access	16	--
AC Queue Metric Events	No Access	--	--
AC Queue Metrics	--	--	Visible
AC Real Time Queue Metrics	Read, Create, Edit, Delete, View All, Modify All	16	--
AC Voice Id Channel	Read, Create, Edit, Delete, View All, Modify All	15	--
AC Voicemail Drops	Read, Create, Edit, Delete, View All, Modify All	10	Visible
AC Wisdom	--	--	Visible

AC_Manager

Object Name	Object Permissions	Total Fields	Tab Settings
AC Agent Performance	Read, View All	124	--
AC CCP Overlay Elements	No Access	9	--
AC Contact Channel Analytics	Read, Create, Edit, Delete, View All, Modify All	31	Visible
AC Contact Channels	Read, Create, Edit, View All	24	--
AC Contact Trace Records	Read, Create, Edit, View All, Modify All	50	--
Accounts	No Access	25	--
AC CTI Adapters	Read	22	Visible
AC CTI Attributes	Read	11	--
AC CTI Scripts	Read	10	--
AC Events	Read, Create	--	--
AC Features	Read	6	--
AC Guided Setup	--	--	--
AC Historical Queue Metrics	Read, View All	119	--
AC Phone Calls	No Access	22	--
AC Presence Sync Rules	Read, View All	13	--
AC QueueMatrices	No Access	16	--
AC Queue Metric Events	Read	--	--
AC Queue Metrics	--	--	Visible
AC Real Time Queue Metrics	Read, View All	16	--
AC Voice Id Channel	Read, Create, Edit, Delete, View All, Modify All	15	--
AC Voicemail Drops	Read, Create, Edit, Delete	10	Available
AC Wisdom	--	--	--

AC_Agent

Object Name	Object Permissions	Total Fields	Tab Settings
AC Agent Performance	Read	124	--
AC CCP Overlay Elements	No Access	9	--
AC Contact Channel Analytics	Read, View All	31	Visible
AC Contact Channels	Read, Create, Edit, View All	24	--
AC Contact Trace Records	Read, Edit, View All	50	--
Accounts	No Access	25	--
AC CTI Adapters	Read	22	--
AC CTI Attributes	Read	11	--
AC CTI Scripts	Read	10	--
AC Events	Read, Create	--	--
AC Features	Read	6	--
AC Guided Setup	--	--	--
AC Historical Queue Metrics	Read	119	--
AC Phone Calls	No Access	22	--
AC Presence Sync Rules	Read, View All	13	--
AC QueueMatrices	No Access	16	--
AC Queue Metric Events	Read	--	--
AC Queue Metrics	--	--	Visible
AC Real Time Queue Metrics	No Access	16	--
AC Voice Id Channel	Read, Create, Edit, Delete, View All, Modify All	15	--
AC Voicemail Drops	Read, Create, Edit, Delete	10	Available
AC Wisdom	--	--	--

Configure the Toolkit settings

1. Navigate to **Setup** then in type **Custom Settings** in Quick Find

The screenshot shows the Salesforce Setup interface. At the top, there is a blue cloud icon followed by the word "Setup". Below it is a navigation bar with three tabs: "Home" (which is selected and highlighted in blue), "Objects", and "Custom Settings". A search bar is located below the navigation bar, containing the text "custom settings". Under the "Custom Settings" tab, there is a list of items. The first item is "Custom Code", which has a sub-item "Custom Settings" highlighted with a yellow background.

2. Next to the Toolkit for Amazon Connect custom setting, choose **Manage**

The screenshot shows the "Custom Settings" page in Salesforce. At the top, there is a heading "Custom Settings" and a brief description: "Use custom settings to create and manage custom data at the organization, profile, and user levels. Custom settings data is stored in the database and can be used to access it efficiently, without the cost of repeated queries. Custom settings data can be used by formula fields, Visualforce, Apex, and triggers." Below this is a "Get Usage" button. The "View:" dropdown is set to "All" and there is a "Create New View" link. A navigation bar with letters A through Z is present. The main table lists custom settings. One row is selected, showing the "Label" as "Toolkit for Amazon Connect", "Visibility" as "Public", "Settings Type" as "Hierarchy", "Namespace Prefix" as "amazonconnect", and "Description" as "Configuration settings of the Toolkit for Amazon Connect." There is also a "Manage" link with a downward arrow icon.

3. Select **New**

The screenshot shows the "New" page for the "Toolkit for Amazon Connect" custom setting. The title is "Custom Setting" and the sub-title is "Toolkit for Amazon Connect". A note says: "If the custom setting is a list, click **New** to add a new set of data. For example, if you have a list of phone numbers, click **New** to add a new number." Another note says: "If the custom setting is a hierarchy, you can add data for the user, profile, or organization. For example, if a specific user is running the app, a specific profile, or just a general user, click **New** to add data for that user." Below this is a "New" button. A section titled "Default Organization Level Value" is shown with a downward arrow icon.

4. On the following page, provide the URL to your Amazon Connect instance. This value can be found in your Amazon Connect console.

Amazon Connect virtual contact center instances

Select a virtual contact center instance to manage its directory, administrator(s), telephony options, data storage,

Add an instance

Remove

Instance Alias	Access URL	Channels
<input type="checkbox"/> [REDACTED]	https://[REDACTED]f.my.connect.aws	Inbound, outbound telephony
<input type="checkbox"/> [REDACTED]	https://[REDACTED]awsapps.com...	Inbound, outbound telephony

Toolkit for Amazon Connect Edit

Provide values for the fields you created. This data is cached with the application.

Edit Toolkit for Amazon Connect

Save

Cancel

Toolkit for Amazon Connect Information

Location

Url 

5. You will also see the option to enable and disable certain triggers in the package, which you can configure to meet your needs. You can change these whenever you would like to. For more details, see below

These are options we provide that allow you to toggle certain functionality in the adapter.

- CCA Case Trigger - This trigger looks for any ContactChannelAnalytics records that could be related to a updated/inserted Case, and creates a relationships between the two records. This trigger uses batching to process the update requests.
- CCA Contact Trigger - This trigger looks for any ContactChannelAnalytics records that could be related to a updated/inserted Contact, and creates a relationships between the two records. This trigger uses batching to process the update requests.
- Case Contact CCA Trigger - This trigger looks for any Case/Contact records that could be related to an updated/inserted ContactChannelAnalytics record, and creates a relationships between the records.
- Task Trigger - This trigger creates a ContactChannel record for any inserted/updated task that with a `CallObject` field that does not currently have a ContactChannel record created before.

6. Select **Save**

Create the Softphone Layout

The softphone layout settings will tell the console what resources are available for screenpop by default and what to do under different match conditions.

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** box, type **Softphone**, then choose **Softphone Layouts** from the results
3. If you are presented with the Get Started message, choose **Continue**
4. On the Softphone Layouts page, choose **New**

A softphone is a customizable call control tool that appears in the sidebar of every salesforce.com page if a user is assigned to a call center and is working on a machine on which a CTI adapter has been installed. Similar to page layouts, you can design custom softphone layouts and assign them to call center users based on their user profile.

Name	Default	Created By Alias	Created Date	Softphone Layout Assignment	Last Modified By Alias	Last Modified Date
No records to display.						

5. Enter a name for the layout, such as **AmazonConnectDefault**, then select the **Is Default Layout** checkbox.

Each softphone layout allows you to customize the appearance of a softphone for inbound, outbound, and transfer calls. You can also define how the softphone handles screen pops.

Save Cancel

Name Is Default Layout

6. Expand **Display these salesforce.com objects** and select objects that CTI Connector should be able to search, for a screen-pop query. In this example, Case has been added to the default selection, allowing search and screen-pop by CaseID.

Display these salesforce.com objects:

▼ Account, Contact, Lead, Case

Available	Selections
AC Agent Performance	Account
AC Contact Channel	Contact
AC Contact Channel Analytics	Lead
AC Contact Trace Record	Case
AC CTI Adapter	
AC CTI Attribute	
AC CTI Script	
AC Event	
AC Historical Queue Metrics	
AC Phone Call	
AC Presence Sync Rule	
AC QueueMatrix	
AC Queue Metric Event	
AC Real Time Queue Metric	

Add Remove Up Down

7. If desired, configure the search behavior to your requirements

- If single Account found, display: Account Name
If multiple matches are found, only the Account Name is displayed in Salesforce Classic. In Lightning Experience, all the selected fields are displayed.
- If single Contact found, display: Name
If multiple matches are found, only the Name is displayed in Salesforce Classic. In Lightning Experience, all the selected fields are displayed.
- If single Lead found, display: Name
If multiple matches are found, only the Name is displayed in Salesforce Classic. In Lightning Experience, all the selected fields are displayed.
- If single Case found, display: Case Number
If multiple matches are found, only the Case Number is displayed in Salesforce Classic. In Lightning Experience, all the selected fields are displayed.

[Edit](#)[Edit](#)[Edit](#)[Edit](#)

8. Additionally, validate the Screen Pop settings. Please note that the default behavior is to not pop a screen if there is more than one result

Screen Pop Settings

Help about this section ?

- Screen pops open within: Existing browser window [Edit](#)
- No matching records: Don't pop any screen [Edit](#)
- Single-matching record: Pop detail page [Edit](#)
- Multiple-matching records: Pop to search page [Collapse](#)
 - Don't pop any screen
 - Pop to search page
 - Pop to Visualforce page
 - Pop to flow

9. Once you have configured the search behavior, choose **Save**

Retrieve the Salesforce API Version

1. Log in into your Salesforce org and go to **Setup**

2. In the **Quick Find** field, type **apex**, then select **Apex Classes** from the results

3. Select New

Action	Name ↑	Namespace Prefix	Api Version	Status	Size Without Comments	Last Modified By	Has Trace Flags
New							

4. Select the Version Settings tab

Apex Class

Apex Class Edit

Save Quick Save Cancel

Apex Class Version Settings



1

5. Note the Salesforce.com API version in your notepad. The pattern of this value is vXX.X.

Apex Class

Apex Class Edit

Save Quick Save Cancel

Apex Class Version Settings

Name	Version
Salesforce.com API	47.0
Amazon Connect - Universal Package	4.2

Setting up the Salesforce API User

The Lambda functions authenticate with Salesforce via user credentials. It is a common practice to create an API user account for this purpose.

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** field, type **profiles**, then select **Profiles** from the results
3. Select New Profile

A screenshot of the Salesforce Profiles page. The top navigation bar shows 'SETUP' and 'Profiles'. Below the header, there's a sub-header 'Profiles'. A button bar at the bottom left includes 'All Profiles', 'Edit | Delete | Create New View', and a red-bordered 'New Profile' button. To the right of the 'New Profile' button is a small icon.

4. Provide a Profile Name, such as **API_ONLY**

5. From the **Existing Profile** dropdown, select **System Administrator** **NOTE:** You're advised to use a full Salesforce License for the user to be able to set the below permissions and have full access to avoid any other errors.

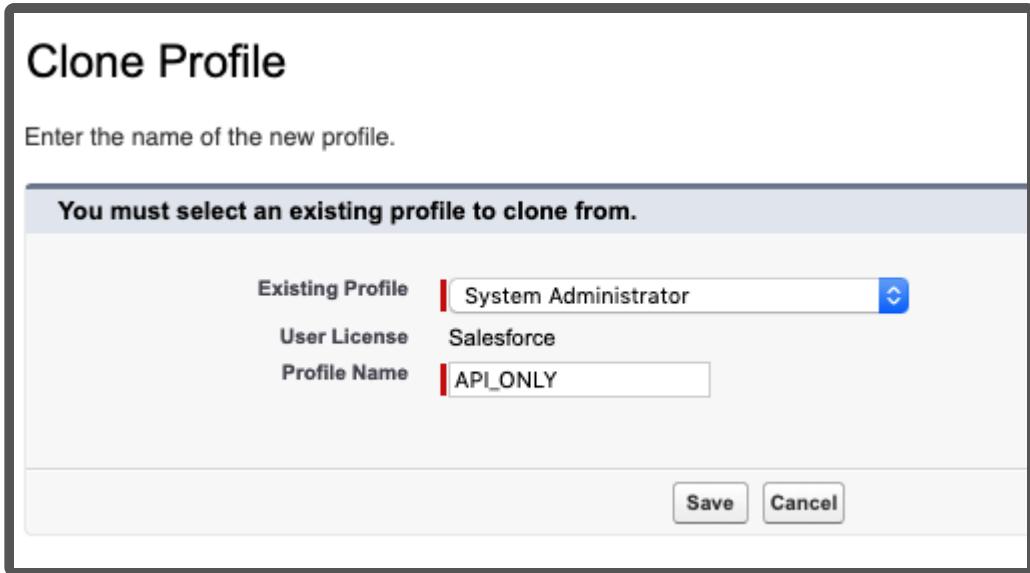
Clone Profile

Enter the name of the new profile.

You must select an existing profile to clone from.

Existing Profile	System Administrator
User License	Salesforce
Profile Name	API_ONLY

Save Cancel

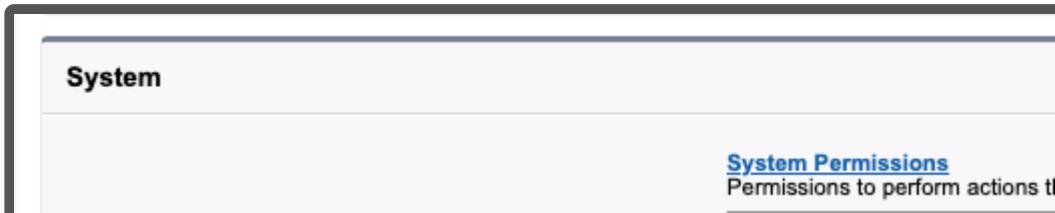


6. Select **Save** to create the new profile

7. Once the new profile page opens, scroll down to and select the **System Permissions** section

System

System Permissions
Permissions to perform actions t



8. When the next page opens, select **edit**

9. Make sure the **Lightning Experience User** option is unselected

Lightning Experience User



10. Select **Save**, and confirm the changes

11. Go back to the Profile Overview, scroll down, and select **Password Policies**

System

Settings that apply across all apps, such as record and user management
[Learn More](#)

System Permissions

Permissions to perform actions tha

Login Hours

Settings that control when users ca

Login IP Ranges

Settings that control the IP address

Service Providers

Permissions that let users switch to

Session Settings

Settings that control required sessi

Password Policies

Profile Based password policies

Default Experience

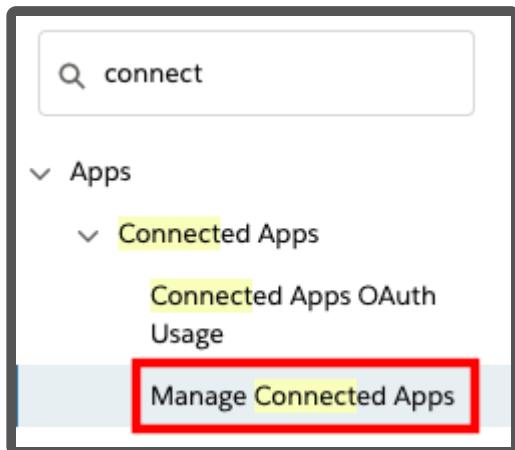
Setting for assigning a default com

12. Select **Edit**.

13. Set **User passwords expire in** to **Never expires** NOTE: Failure to this may lead to production outages.

14. Select **Save**.

15. In the **Quick Find** field, type **connect**, then select **Manage Connected Apps** from the results



16. Select the app you have created earlier, **Amazon Connect Integration**

17. In the profiles section, select **Manage Profiles**

18. Select the new **API_Only** profile that you just created

19. Select **Save** at the bottom of the page

20. In the **Quick Find** field, type **users** then select **Users** from the results

21. Select New User

22. Set the required fields as:

- a. Last Name: apiuser
- b. Alias: apiuser
- c. Email: provide a valid email address
- d. Username: apiuser@<yoursalesforcedomain>.com
- e. Nickname: apiuser

23. On the right-hand side, set **User License** to **Salesforce**

24. Set Profile to API_ONLY

25. Choose **Save**

26. In **Quick Find**, search for "Permission Sets". Select the **AC_Administrator** permission set.

The screenshot shows the Salesforce Setup interface. The left sidebar has sections for Users (Permission Set Groups, **Permission Sets**), Custom Code (Custom Permissions), and Global Search. The main content area is titled "Permission Sets" and contains a table of permission sets. One row, "AC Administrator", is highlighted with a red box. The table columns are Action, Permission Set Label, Description, and License. The "Description" for AC Administrator states: "Allows the user to configure Amazon Connect setup and provides ...".

Action	Permission Set Label	Description	Licenses
<input type="checkbox"/>	Clone	AC Administrator	Allows the user to configure Amazon Connect setup and provides ...
<input type="checkbox"/>	Clone	AC Agent	
<input type="checkbox"/>	Clone	AC_CallRecording	
<input type="checkbox"/>	Clone	AC_Manager	

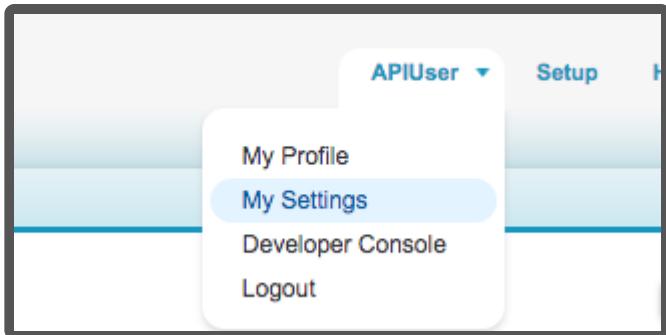
27. Select **Manage Assignments**. Add the apiuser you just created to the permission set.

28. A confirmation email with an **activation link** will be sent to the email address provided. Choose the link to activate your user and set their password

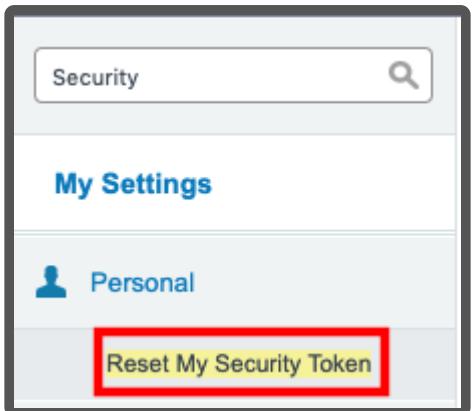
29. Fill out the form to set a password for the API user

30. Select **Change Password**. The API user will log into the Salesforce Classic view

31. Access the API user's personal settings by selecting the username in the top right corner, then choose **My Settings**



32. In the **Quick Find** field, type **security** then select **Reset My Security Token** from the results

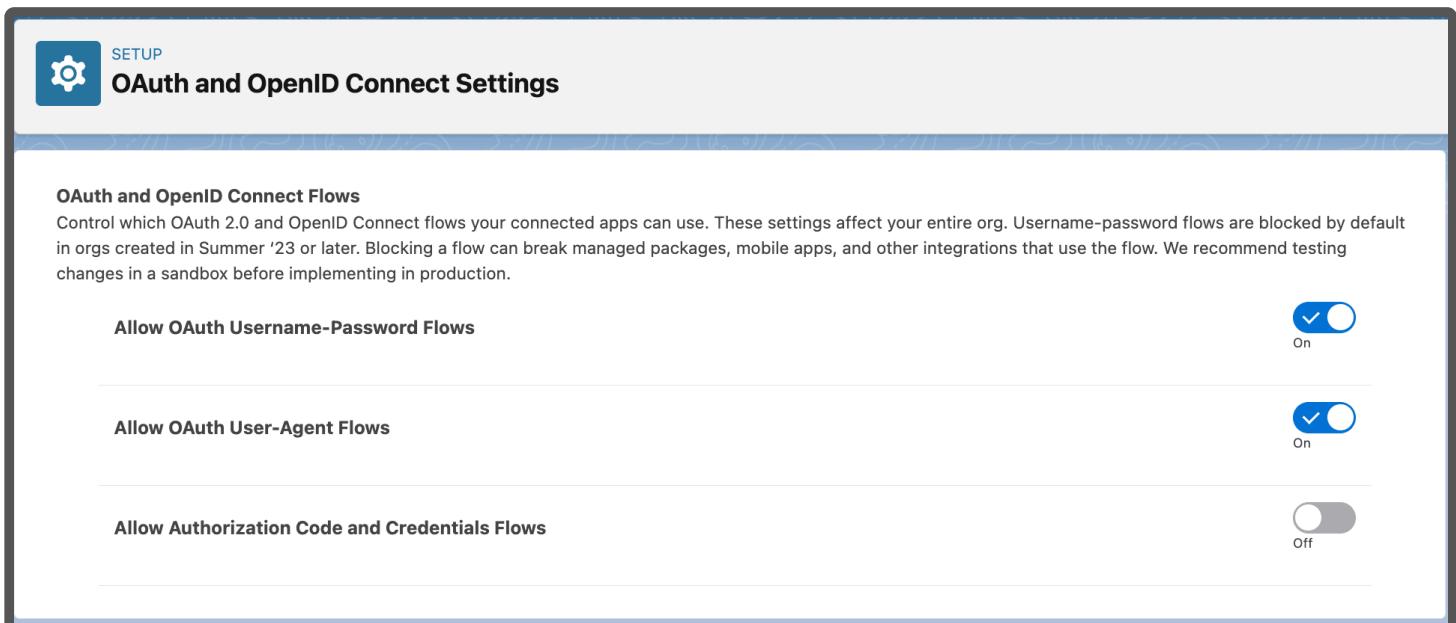


33. Select **Reset Security Token**. Your security token will be emailed to you

34. Copy the security token from the email to your notepad

Allowing the API user to authenticate using password

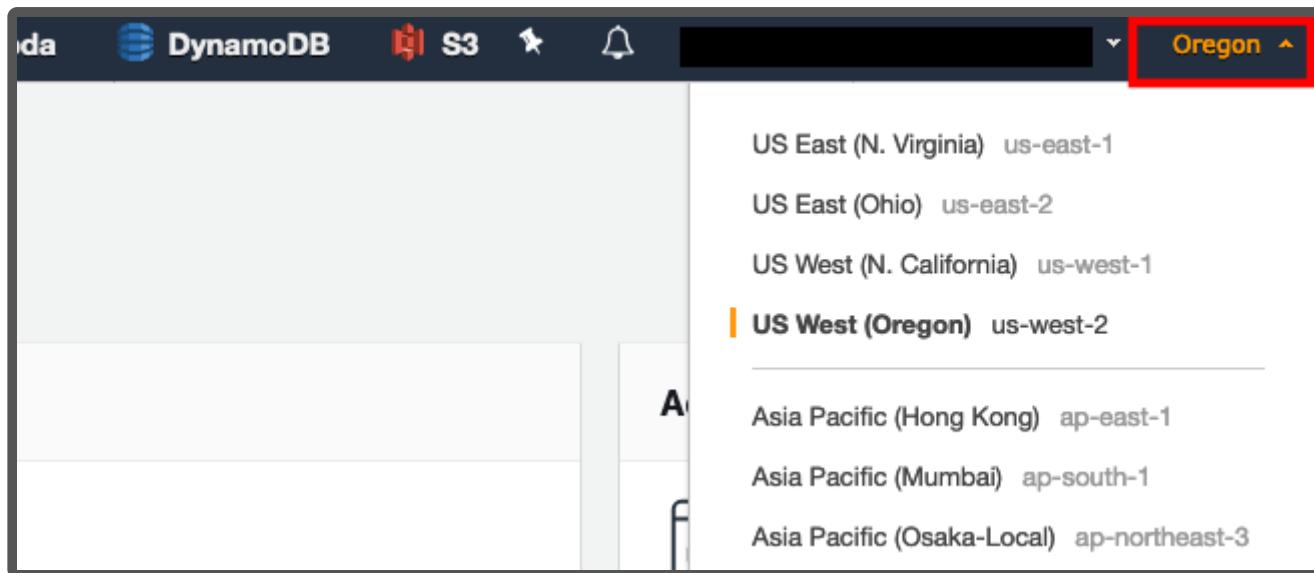
The api user created above authenticates using username-password flow in Salesforce. This flow needs to be unblocked and to do that, go to *Setup* and in the Quick Find box, search for **OAuth and OpenID Connect Settings**. After that, make sure that the toggles for **Allow OAuth Username-Password Flows** and **Allow OAuth User-Agent Flows** are turned ON, as shown in below image.



Setting up the SecretsManager Secret

To ensure that your Salesforce credentials are secure, the Lambdas require that the credentials are stored in AWS Secrets Manager. AWS Secrets Manager is a highly secure service that helps you store and retrieve secrets.

1. In a new browser tab, login to the AWS console
2. Make sure you are in the same region as your Amazon Connect instance. You can set the region by expanding the region selector in the upper right and choosing the region



3. Navigate to the [Secrets Manager console](#)
4. Select **Secrets**
5. Select **Store a new secret**
6. Select **Other types of secrets**
7. Make sure **Secret key/value** is selected
8. Enter key value pairs that match the following:
 - a. **Key:** Password, **Value:** the password for the API user that you configured in the previous section
 - b. **Key:** ConsumerKey, **Value:** the Consumer Key for the Connected App you created in the previous section
 - c. **Key:** ConsumerSecret, **Value:** the Consumer Secret for the Connected App you created in the previous section

d. **Key:** AccessToken, **Value:** this is the access token for the API user that you configured in the previous section

9. For the encryption key, click **Add new key**

10. Select **Create Key**

11. Make sure key type is set to **symmetric**

12. Give your key an **alias**, like *SalesforceCredentialsSecretsManagerKey*

13. Click Next

14. Select administrators you want to have access permission to change the key policy. Make sure you are being as restrictive as possible

15. Click Next

16. Select the users and roles you want to have access to the Salesforce credentials in Secrets Manager. Make sure you are being as restrictive as possible

17. Click Next

18. Click Finish

19. Click on the managed key that you just created (which is *SalesforceCredentialsSecretsManagerKey* in this case).

20. Note down the ARN. This is *SalesforceCredentialsKMSKeyARN* that will be used later when installing the Amazon Connect Salesforce Lambda package.

21. Navigate back to the Secrets Manager setup tab

22. Select the key you just created

Specify the key/value pairs to be stored in this secret [Info](#)

Secret key/value

Plaintext

Password

Password

Remove

ConsumerKey

ConsumerKey

Remove

ConsumerSecret

ConsumerSecret

Remove

AccessToken

AccessToken

Remove

[+ Add row](#)

Select the encryption key [Info](#)

Select the AWS KMS key to use to encrypt your secret information. You can encrypt using the default service encryption key that AWS Secrets Manager creates on your behalf or a customer master key (CMK) that you have stored in AWS KMS.

SalesforceCredentialsSecretsManagerKey



[Add new key](#)

Cancel

Next

23. Click Next

24. Give your secret a name, like *SalesforceCredentials*

25. Click Next

26. Make sure **automatic rotation** is disabled.

27. Click Next

28. Click Store

29. Select the secret you just created, and copy the Secret ARN

SalesforceCredentials

Secret details

Actions ▾

Encryption key
SalesforceCredentialsSecretsManagerKey

Secret name
SalesforceCredentials

Secret ARN
[REDACTED]

Secret description
-

Test the Salesforce Lambda Core Functionality

The package provides a core Lambda function (`sflInvokeAPI`) that supports multiple operations, like lookup, create and update. For the initial validation, sample events are provided within the function. Validating this function provides a good check that the installation and configuration is correct.

Validating the lambda functions requires the use of test events to simulate data coming into the function as it would in a typical deployment. Each function has a set of test event samples included to make validation easier.

Validate the core functionality

1. In a new browser tab, login to the [AWS console](#)
2. Open the [AWS Lambda Console](#)
3. In the Filter field, enter `sflInvokeAPI` and press enter, this will filter your list out to the core function that we just installed

Functions (77)

Actions ▾

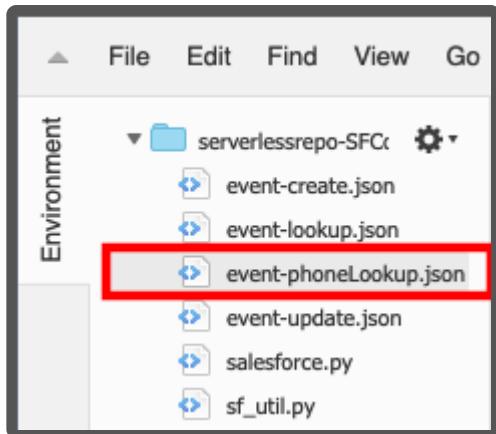
Add filter

Keyword : `sflInvokeAPI` (X)

Function name	Description	Runtime	Code size
serverlessrepo-SFConsolidatedLambdaPac-sflInvokeAPI-5504EV6KL9E8		Python 3.7	32.1 kB

4. Select the **function name**. First, we will validate a phone number lookup.

5. In the Environment pane, double-click the event-phoneLookup.json file



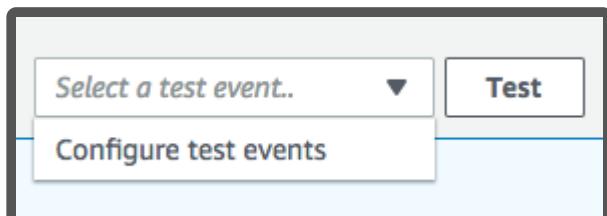
6. The test even JSON will open in the Lambda editor

7. Modify the value for sf_phone to match the phone number of the test contact you created when you setup the CTI adapter or for any valid contact in your Salesforce org| NOTE: The phone number must be in [E.164 format](#)

```
1 {  
2     "Details": {  
3         "Parameters": {  
4             "sf_operation" : "phoneLookup",  
5             "sf_phone": "+14155551212",  
6             "sf_fields": "Id, Name, Email"  
7         }  
8     }  
9 }
```

8. Select the entire JSON event and copy it, then close the **event-phoneLookup.json** tab.

9. In the top-right corner, select drop-down arrow next to **Test** and choose **Configure test events**



10. Select the radio button for **Create new test event** and provide an event name, for example: **phoneLookup**

11. Select the existing event JSON and **delete** it. Paste the modified JSON payload you copied from the **event-phoneLookup.json** file

Configure test event



A function can have up to 10 test events. The events are persisted so you can switch to another computer or web browser and test your function with the same events.

- Create new test event
- Edit saved test events

Event template

Hello World



Event name

phoneLookup

```
1 [{  
2   "Details": {  
3     "Parameters": {  
4       "sf_operation": "phoneLookup",  
5       "sf_phone": "+14155551212",  
6       "sf_fields": "Id, Name, Email"  
7     }  
8   }  
9 }]
```

12. Select **Create** to save your test event

13. By default, your new test event should be selected in the drop-down list to the left of the Test button.

A screenshot showing a horizontal button group. From left to right: a dropdown menu containing the text 'phoneLookup' and a downward arrow; a blue rectangular button labeled 'Test'; and an orange rectangular button labeled 'Save'.

14. Select **Test**

15. If successful, the result will contain fields defined in "sf_fields" parameter in the invocation event

Execution result: succeeded (logs)

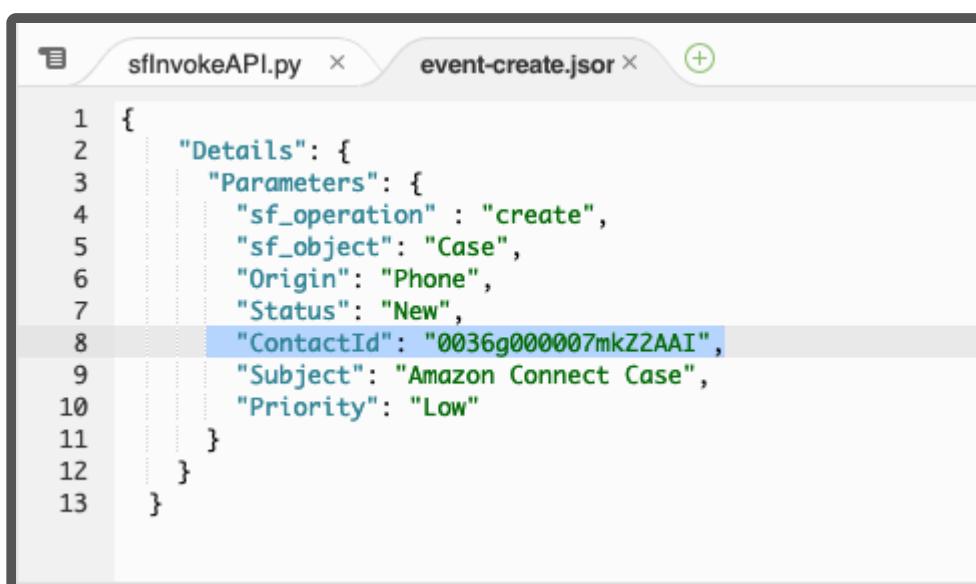
▼ Details

The area below shows the result returned by your function execution. [Learn more](#)

```
{  
    "Id": "0036g000007mkZ2AAI",  
    "Name": "John Smith",  
    "Email": null,  
    "sf_count": 1  
}
```

16. Copy the value for the **Id** key in the response. Next, we are going to use that Id to create a Case in Salesforce.

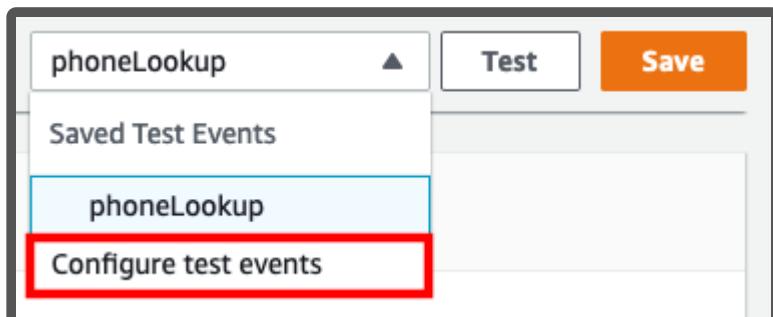
17. In the Environment pane, double-click the **event-create.json** file. Replace the existing ContactId value with the ID value you copied previously.



```
sfInvokeAPI.py x event-create.json x +  
1 {  
2     "Details": {  
3         "Parameters": {  
4             "sf_operation" : "create",  
5             "sf_object": "Case",  
6             "Origin": "Phone",  
7             "Status": "New",  
8             "ContactId": "0036g000007mkZ2AAI",  
9             "Subject": "Amazon Connect Case",  
10            "Priority": "Low"  
11        }  
12    }  
13 }
```

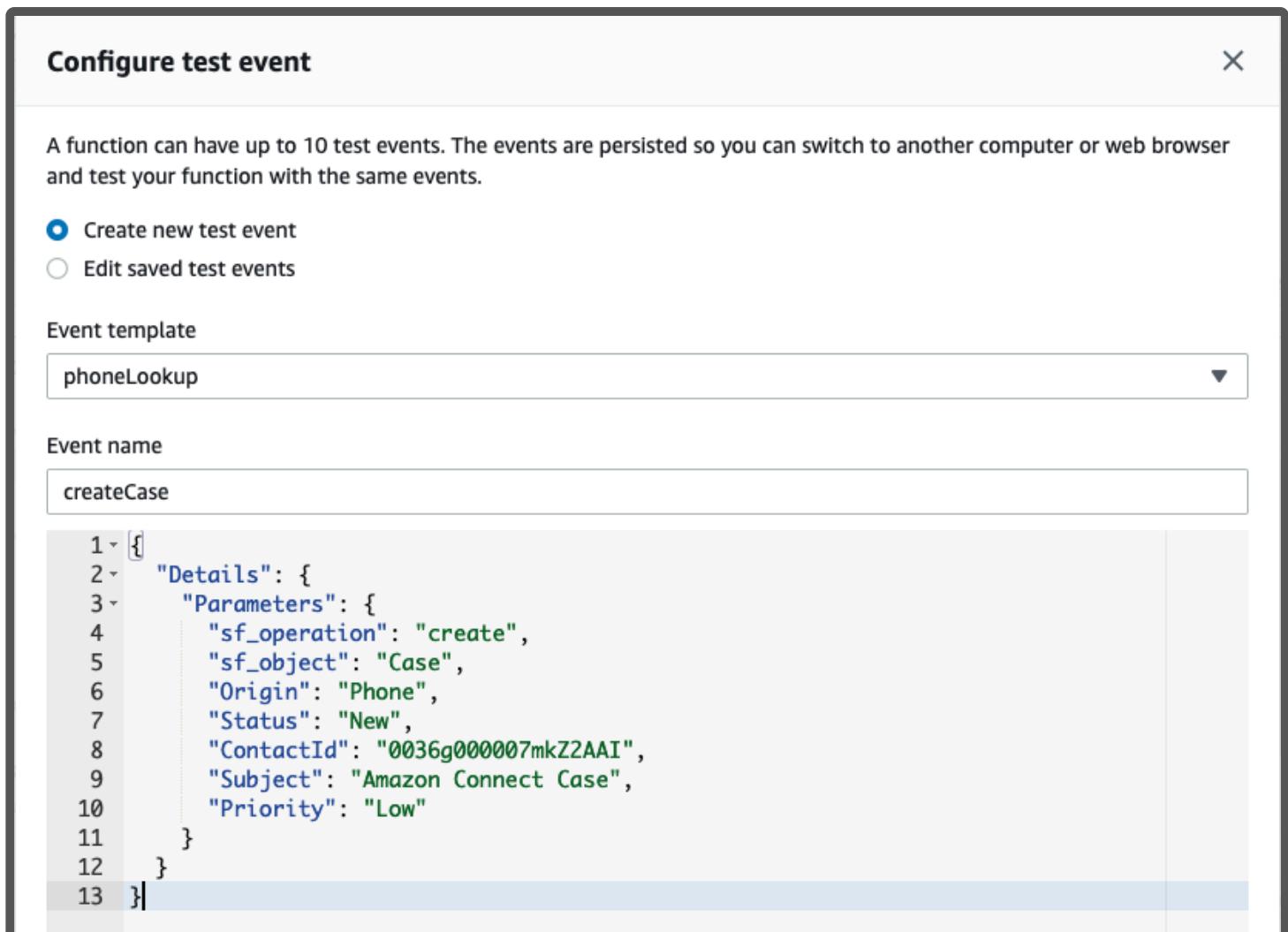
18. Select the entire JSON event and copy it, then close the **event-create.json** tab.

19. In the top-right corner, select drop-down arrow next to **Test** and choose **Configure test events**



20. Select the radio button for **Create new test event** and provide an event name, for example: **createCase**

21. Select the existing event JSON and **delete** it. Paste the modified JSON payload you copied from the **event-create.json** file



22. Select **Create** to save your test event

23. By default, your new test event should be selected in the drop-down list to the left of the Test button.



24. Select **Test**

25. If successful, the result will contain the Case Id

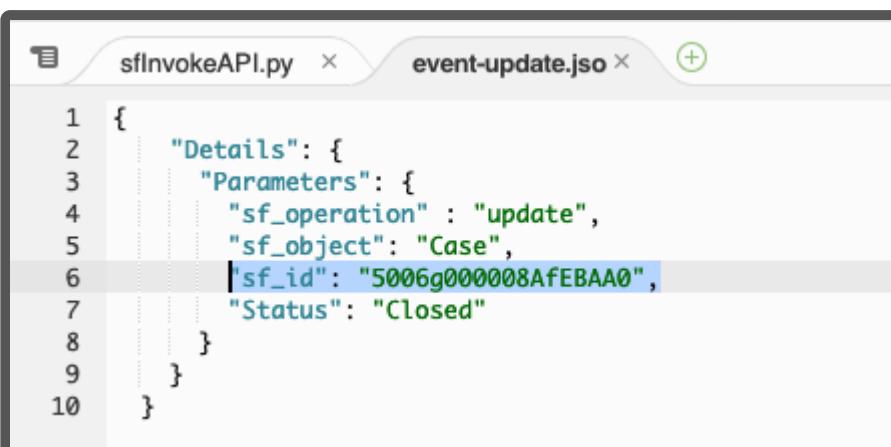
Execution result: succeeded (logs)

▼ Details

The area below shows the result returned by your function execution. [Learn](#)

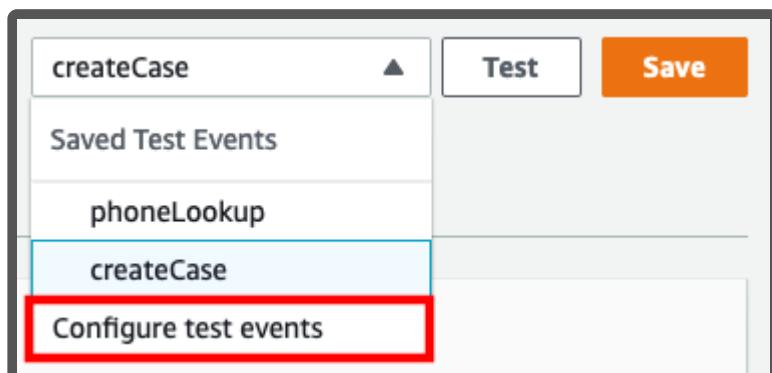
```
{  
    "Id": "5006g000008AfEBAA0"  
}
```

26. Copy the value for the **Id** key in the response.
27. When we created the case, the **Status was set to New** and the **Priority to Low**. We are going to use the update operation to close the case.
28. In the Environment pane, double-click the **event-update.json** file and replace the existing Case Id in "sf_id" parameter with the new one you copied from the last test result



```
1 {  
2     "Details": {  
3         "Parameters": {  
4             "sf_operation" : "update",  
5             "sf_object": "Case",  
6             "sf_id": "5006g000008AfEBAA0",  
7             "Status": "Closed"  
8         }  
9     }  
10 }
```

29. Select the **entire JSON event** and copy it, then close the **event-update.json** tab.
30. In the top-right corner, select drop-down arrow next to **Test** and choose **Configure test events**



31. Select the radio button for **Create new test event** and provide an event name, for example: **updateCase**

32. Select the existing event JSON and **delete** it. Paste the modified JSON payload you copied from the **event-update.json** file

Configure test event

A function can have up to 10 test events. The events are persisted so you can switch to another computer or web browser and test your function with the same events.

Create new test event
 Edit saved test events

Event template

createCase

Event name

closeCase

```
1 - {  
2 -   "Details": {  
3 -     "Parameters": {  
4 -       "sf_operation": "update",  
5 -       "sf_object": "Case",  
6 -       "sf_id": "5006g000008AFEBAA0",  
7 -       "Status": "Closed"  
8 -     }  
9 -   }  
10 }
```

33. Select **Create** to save your test event

34. By default, your new test event should be selected in the drop-down list to the left of the Test button.



The screenshot shows a user interface for testing a function. It features a dropdown menu labeled 'closeCase' with a downward arrow, a 'Test' button in a light blue box, and a 'Save' button in an orange box. The 'Save' button is currently highlighted, indicating it is the active or next step.

35. Select **Test**

36. If successful, the result will be the **HTTP 204 No Content** success status response code

Execution result: succeeded (logs)

▼ Details

The area below shows the result returned by your function:

```
{  
  "Status": 204  
}
```

37. Log in into your Salesforce org and go to the **Service Console**

38. In the search box, change the object type to Cases and type Amazon Connect Case, then press enter



39. You should find 1 case opened by the API user, and the status should be closed

Cases					
1 Result					
Case Number	Subject	Status	Date/Time Opened	Case Owner Alias	
00001026	Amazon Connect Case	Closed	1/23/2020, 10:13 PM	apiuser	

40. You have completed core function validation

Allow Amazon Connect to Access the `sflnvokeAPI` Lambda Function

Once you have validated function, you can use the Amazon Connect console to add the `sflnvokeAPI` Lambda function to your Amazon Connect instance. This automatically adds resource permissions that allow Amazon Connect to invoke the function.

Add the Lambda function to your Amazon Connect instance

1. In a new browser tab, login to the [AWS console](#)
2. Navigate to the [Amazon Connect Console](#)
3. Select your **Instance Alias**
4. In the navigation pane, choose **Contact flows**.

Amazon Connect

X

Instances

Overview

Telephony

Data storage

Data streaming

Contact flows

Analytics tools

Approved origins

Customer profiles

5. Scroll down to the **AWS Lambda** section, and select the function that includes `sflnvokeAPI` in the name

AWS Lambda

By using AWS Lambda function, you can retrieve data from database and other services and be routed to the appropriate contact flow branch. By adding Lambda functions, you can invoke them [Create a new Lambda function](#)

Lambda Functions

serverlessrepo-AmazonConnectSalesforce-sflnvokeAPI-Z... ▾

+ Add Lambda Function

Lambda Functions

6. Choose **Add Lambda Function**. Confirm that the ARN of the function is added under **Lambda Functions**.

Lambda Functions

Function Arn	Arn	Copy to clipboard	Action
serverlessrepo-AmazonConnectSalesforce-sflnvokeAPI-Z...	arn:aws:lambda:us-west-2:...	Copy	Remove

7. The AWS Lambda function has been added to your Amazon Connect instance.

Setting Up The CTI Adapter Managed Package Manually

Below are manual setup instructions for the Salesforce CTI Adapter Managed Package. After following the below steps, be sure to follow the instructions for setting up the Salesforce Lambdas [here](#).

When installing v5.15, please **confirm that the application was installed for admins only** (see [installation](#) for more details). If you did this by accident, then you will have to [manually edit the profiles](#) to remove the permissions to the objects and pages created by the app.

Set Access Permissions

All users must be assigned the required permission set to access Salesforce metadata. The Amazon Connect CTI Adapter includes two Permission Sets, one for agents and one for managers, that grant users the appropriate access for their role. More information on assigning user permissions can be found in the [Salesforce help documentation](#).

1. Log in into your Salesforce org and go to **Setup**
2. In **Quick Find**, enter **Permission** and select **Permission Sets** from the results
3. Choose **AC_Administrator**, **AC_Agent** or **AC_Manager** as appropriate for the user(s)

Permission Sets

On this page you can create, view, and manage permission sets.

In addition, you can use the SalesforceA mobile app to assign permission sets to a user. Download SalesforceA from the App Store or Google Play: [iOS](#) | [Android](#)

[All Permission Sets](#) [Edit](#) | [Delete](#) | [Create New View](#)

Action	Permission Set Label	Description
Del Clone	AC Administrator	Allows the user to configure Amazon Connect setup and provides full access to Am...
Del Clone	AC Agent	
Del Clone	AC Manager	

4. Choose **Manage Assignments**.
5. Choose **Add Assignments**.
6. Select the users to assign the permissions, then choose **Assign**.

Assign Users
All Users

View: All Users [Edit](#) | [Create New View](#)

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All

Action	Full Name	Alias	Username	Last Login	Role	Active	Profile	Manager
<input type="checkbox"/>	Chatter Expert	Chatter	[REDACTED]@chatter.salesforce.com			<input checked="" type="checkbox"/>	Chatter Free User	
<input checked="" type="checkbox"/>	Douglas Jason	jDoug	[REDACTED]	1/21/2020, 10:40 PM		<input checked="" type="checkbox"/>	System Administrator	
<input type="checkbox"/>	Edit User Integration	Integ	[REDACTED]			<input checked="" type="checkbox"/>	Analytics Cloud Integration User	
<input type="checkbox"/>	Edit User Security	sec	[REDACTED]			<input checked="" type="checkbox"/>	Analytics Cloud Security User	

[Assign](#) [Cancel](#)

7. Repeat these steps as needed for all users

AC_Administrator

Object Name	Object Permissions	Total Fields	Tab Settings
AC Agent Performance	Read, Create, Edit, Delete, View All, Modify All	124	--
AC CCP Overlay Elements	No Access	9	--
AC Contact Channel Analytics	Read, Create, Edit, Delete, View All, Modify All	31	Visible
AC Contact Channels	Read, Create, Edit, Delete, View All, Modify All	24	--
AC Contact Trace Records	Read, Create, Edit, Delete, View All, Modify All	50	Visible
Accounts	No Access	25	--
AC CTI Adapters	Read, Create, Edit, Delete, View All, Modify All	22	Visible
AC CTI Attributes	Read, Create, Edit, Delete, View All, Modify All	11	--
AC CTI Scripts	Read, Create, Edit, Delete, View All, Modify All	10	--
AC Events	No Access	--	--
AC Features	Read, Create, Edit, Delete, View All, Modify All	6	--
AC Guided Setup	--	--	Visible
AC Historical Queue Metrics	Read, Create, Edit, Delete, View All, Modify All	119	--
AC Phone Calls	No Access	22	--
AC Presence Sync Rules	Read, Create, Edit, Delete, View All, Modify All	13	--
AC QueueMatrices	No Access	16	--
AC Queue Metric Events	No Access	--	--
AC Queue Metrics	--	--	Visible
AC Real Time Queue Metrics	Read, Create, Edit, Delete, View All, Modify All	16	--
AC Voice Id Channel	Read, Create, Edit, Delete, View All, Modify All	15	--
AC Voicemail Drops	Read, Create, Edit, Delete, View All, Modify All	10	Visible
AC Wisdom	--	--	Visible

AC_Manager

Object Name	Object Permissions	Total Fields	Tab Settings
AC Agent Performance	Read, View All	124	--
AC CCP Overlay Elements	No Access	9	--
AC Contact Channel Analytics	Read, Create, Edit, Delete, View All, Modify All	31	Visible
AC Contact Channels	Read, Create, Edit, View All	24	--
AC Contact Trace Records	Read, Create, Edit, Delete, View All, Modify All	50	--
Accounts	No Access	25	--
AC CTI Adapters	Read	22	Visible
AC CTI Attributes	Read	11	--
AC CTI Scripts	Read	10	--
AC Events	Read, Create	--	--
AC Features	Read	6	--
AC Guided Setup	--	--	--
AC Historical Queue Metrics	Read, View All	119	--
AC Phone Calls	No Access	22	--
AC Presence Sync Rules	Read, View All	13	--
AC QueueMatrices	No Access	16	--
AC Queue Metric Events	Read	--	--
AC Queue Metrics	--	--	Visible
AC Real Time Queue Metrics	Read, View All	16	--
AC Voice Id Channel	Read, Create, Edit, Delete, View All, Modify All	15	--
AC Voicemail Drops	Read, Create, Edit, Delete	10	Available
AC Wisdom	--	--	--

AC_Agent

Object Name	Object Permissions	Total Fields	Tab Settings
AC Agent Performance	Read	124	--
AC CCP Overlay Elements	No Access	9	--
AC Contact Channel Analytics	Read, View All	31	Visible
AC Contact Channels	Read, Create, Edit, View All	24	--
AC Contact Trace Records	Read, Edit, View All	50	--
Accounts	No Access	25	--
AC CTI Adapters	Read	22	--
AC CTI Attributes	Read	11	--
AC CTI Scripts	Read	10	--
AC Events	Read, Create	--	--
AC Features	Read	6	--
AC Guided Setup	--	--	--
AC Historical Queue Metrics	Read	119	--
AC Phone Calls	No Access	22	--
AC Presence Sync Rules	Read, View All	13	--
AC QueueMatrices	No Access	16	--
AC Queue Metric Events	Read	--	--
AC Queue Metrics	--	--	Visible
AC Real Time Queue Metrics	No Access	16	--
AC Voice Id Channel	Read, Create, Edit, Delete, View All, Modify All	15	--
AC Voicemail Drops	Read, Create, Edit, Delete	10	Available
AC Wisdom	--	--	--

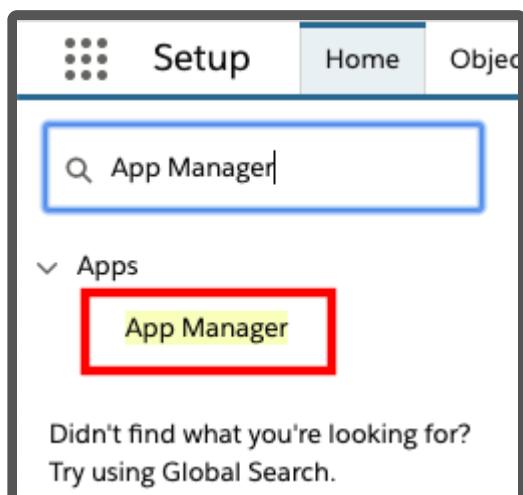
Configure the Lightning Experience

In this guide, we will configure the CTI Adapter for Service Console (Lightning Experience). You may use the same procedure described in this section for other applications.

Configure Service Console

First, you need to add the CTI softphone to your Service Console.

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** box, type **App Manager**, then choose **App Manager** from the result list.



3. Expand the drop-down menu associated to Service Console and select **Edit**.

12	Salesforce Chatter	Chatter	The Salesforce Chatter social network, including profiles and feeds	1/21/2020, 8:46 PM	Classic	✓	▼
13	Service	Service	Manage customer service with accounts, contacts, cases, and more	1/21/2020, 8:46 PM	Classic	✓	▼
14	Service Console	LightningService	(Lightning Experience) Lets support agents work with multiple re...	1/21/2020, 8:46 PM	Lightning	✓	▼
15	Site.com	Sites	Build pixel-perfect, data-rich websites using the drag-and-drop Sit...	1/21/2020, 8:46 PM	Classic	Edit	▼

4. Once the **Lightning App Builder** opens, select **Utility Items (Desktop Only)** from the left Navigation



App Settings

App Details & Branding

App Options

Utility Items (Desktop Only)

Navigation Items

Navigation Rules

User Profiles

5. Choose **Add Utility Item**, then select **Open CTI Softphone**.

The screenshot shows the 'Utility Items' page in the Salesforce setup. At the top, there's a search bar labeled 'Search...'. Below it, a section titled 'Standard (16)' is expanded, showing a list of utility items. The 'Open CTI Softphone' item is highlighted with a red box.

- Chatter Feed
- Chatter Publisher
- Einstein Analytics Dashboard
- Einstein Next Best Action
- Flow
- History
- List View
- Macros
- Notes
- Open CTI Softphone**

6. Change the Label, if desired, then choose **Save**.

Allowlist Your Salesforce Org with Amazon Connect

In order to embed the Amazon Connect Contact Control Panel (CCP) into your Service Console, you need to allowlist two (2) domains for your org with Amazon Connect. This allows for cross domain access to the underlying resources required for the CCP to function.

1. Log in into your Salesforce org and go to **Setup -> My Domain**

2. Copy the entire URL of this page and past it to a text document.

The screenshot shows a web browser window with the address bar containing the URL: 'ed.lightning.force.com/lightning/setup/SetupOneHome/home'. The URL is highlighted with a red box.

3. In the **Quick Find** field, type **visual**, then select **Visual Force Pages** from the results

visual

Custom Code

Visualforce Components

Visualforce Pages

Didn't find what you're looking for?
Try using Global Search.

4. Choose the AC_LightningAdapter Visualforce page

Visualforce Pages

Visualforce Pages provide a robust and easy to use mechanism to create new and exciting user experiences for your application or to enhance existing applications to optimize your users' productivity.

View: All Create New View

Action	Label	Name	Namespace Prefix	Api Version	Description	Created By Alias	Created Date	Last Modified By Alias	Last Modified Date
Security	AC_CreateCSAdapter	AC_CreateCSAdapter	amazonconnect	47.0		JDoug	1/21/2020, 10:41 PM	JDoug	1/21/2020, 10:42 PM
Security	AC_LightningAdapter	AC_LightningAdapter	amazonconnect	47.0		JDoug	1/21/2020, 10:41 PM	JDoug	1/21/2020, 10:42 PM
Security	AC_LightningScriptIncludes	AC_LightningScriptIncludes	amazonconnect	47.0		JDoug	1/21/2020, 10:41 PM	JDoug	1/21/2020, 10:42 PM

5. On the Visualforce detail page, select the **Preview** button. This will open a new browser tab showing the page content, which should only be a button labelled Sign in to CCP. Copy the entire URL of this page and past it to a text document.

Visualforce Pages | Salesforce

https://sfseorga-dev-ed--amazonconnect.visualforce.com/apex/AC_LightningAdapter

6. In a new browser tab, login to the [AWS console](#)

7. Navigate to the [Amazon Connect Console](#)

8. Validate that you are in the correct **AWS region** for your instance, then select your instance alias from the list of instances

Amazon Connect virtual contact center instances

Select a virtual contact center instance to manage its directory, administrator(s), telephony options, data storage, and advanced features.

Add an Instance	Remove	Instance Alias	Access URL	Channels	Create Date	Status
<input type="checkbox"/> sfsatestconsolidated		https://[REDACTED].awsapps...	Inbound, outbound telephony	1/21/2020	Active	

9. Choose **Approved Origins** from the left navigation

10. Select + Add origin

11. In the Enter origin URL field, enter the URL of the page that you copied in step 2. The URL can vary depending on the instance due to a setting named "enhanced domain". You should copy and paste the domain name URL into the field.

12. Select Add. You should see your org domain listed in the Approved origins section.

Approved origins

Once you integrated with a CRM product, add the origins (scheme + host + port) that Amazon Connect will need to have access to.

`https://[REDACTED]dev-ed.lightning.force.com` [remove](#)

13. Select + Add origin

14. In the Enter origin URL field, enter the URL of the visualforce page that you copied in step 5. Only enter the url through the .com. The URL can vary depending on the instance due to a setting named "enhanced domain". You should copy and paste the visual force page domain URL into the field.

15. Select Add. You should see your org domain listed in the Approved origins section

Approved origins

Once you integrated with a CRM product, add the origins (scheme + host + port) that Amazon Connect will need to have access to.

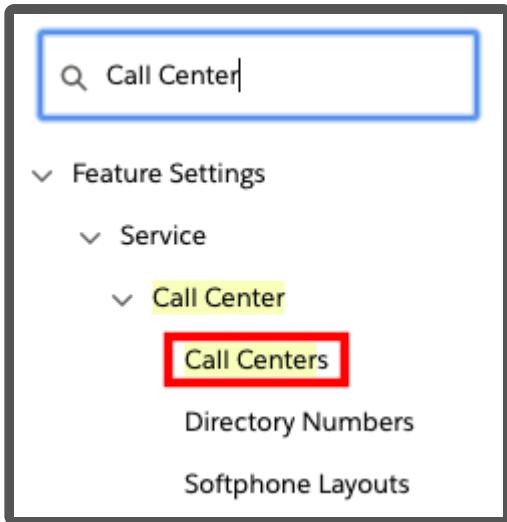
`https://[REDACTED]-dev-ed--amazonconnect.visualforce.com` [remove](#)

Modify the Call Center

Now that you have allowlisted the org in the Amazon Connect Console, you will need to modify the Call Center that was configured in Salesforce when the AppExchange package was installed. Once you complete the configuration, you add users to the Call Center to provide access to it.

1. Log in into your Salesforce org and go to **Setup**

2. In the **Quick Find** field, enter **Call Center**, then select **Call Centers** from the result list



3. If you see the **Say Hello to Salesforce Call Center** page, select **Continue**

4. Select **AC Lightning Adapter**

All Call Centers

A call center corresponds to a single computer-telephony integration (CTI) system already in place. You can manage Call Center features.

Action	Name ↑
Edit Del	AC Lightning Adapter
Edit Del	Amazon Connect CCP Adapter Classic 3.11
Edit Del	Amazon Connect CCP Adapter Console 3.11

5. On the **AC Lightning Adapter** detail page, select **Edit**

6. Replace the **CTI Adapter URL** with the AC Lightning Adapter visualforce page url you copied in the previous section.

7. Next, change the values for **Softphone Height to 570** and the **Softphone Width to 330**, and choose **Save**.

8. Once you return to the AC Lightning Adapter detail page, choose **Manage Call Center Users** in the Call Center Users section

9. On the **AC Lightning Adapter: Manage Users** page, select **Add More Users**.

10. Set filters (if desired) and then choose **Find**.

11. Select the checkbox next to the user to add, then choose **Add to Call Center**.

Full Name	Alias	Username	Role	Profile
Douglas Jason	JDSud	[REDACTED]		System Administrator
User_Integration	Integ	Integration@00d690000004znmwseak.com		Analytics Cloud Integration User
User_Security	sec	insightsecurity@00d690000004znmwseak.com		Analytics Cloud Security User

12. Repeat the steps to add more users.

Configure the Toolkit settings

1. Navigate to **Setup** then in type **Custom Settings** in Quick Find

2. Next to the Toolkit for Amazon Connect custom setting, choose **Manage**

Action	Label	Visibility	Settings Type	Namespace Prefix	Description
Manage	Toolkit for Amazon Connect	Public	Hierarchy	amazonconnect	Configuration settings of the Toolkit for Amazon Connect.

3. Select **New**

If the custom setting is a list, click **New** to add a new set of data. For dialing code.

If the custom setting is a hierarchy, you can add data for the user, the specific user is running the app, a specific profile, or just a general user.

New

▼ Default Organization Level Value

4. On the following page, provide the URL to your Amazon Connect instance. This value can be found in your Amazon Connect console.

Amazon Connect virtual contact center instances

Select a virtual contact center instance to manage its directory, administrator(s), telephony options, data storage,

Add an instance	Remove	
Instance Alias	Access URL	Channels
<input type="checkbox"/> [REDACTED]	https://[REDACTED]f.my.connect.aws	Inbound, outbound telephony
<input type="checkbox"/> [REDACTED]	https://[REDACTED]awsapps.com...	Inbound, outbound telephony

Toolkit for Amazon Connect Edit

Provide values for the fields you created. This data is cached with the application.

Edit Toolkit for Amazon Connect [Save](#) [Cancel](#)

Toolkit for Amazon Connect Information

Location

Url <https://yourinstancename.a>

5. You will also see the option to enable and disable certain triggers in the package, which you can configure to meet your needs. You can change these whenever you would like to. For more details, see below

These are options we provide that allow you to toggle certain functionality in the adapter.

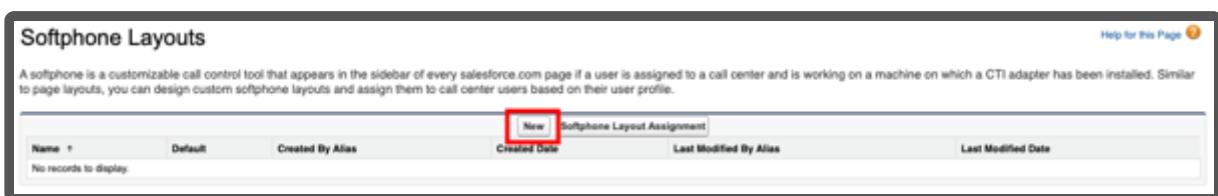
- CCA Case Trigger - This trigger looks for any ContactChannelAnalytics records that could be related to a updated/inserted Case, and creates a relationship between the two records. This trigger uses batching to process the update requests.
- CCA Contact Trigger - This trigger looks for any ContactChannelAnalytics records that could be related to a updated/inserted Contact, and creates a relationship between the two records. This trigger uses batching to process the update requests.
- Case Contact CCA Trigger - This trigger looks for any Case/Contact records that could be related to an updated/inserted ContactChannelAnalytics record, and creates a relationship between the records.
- Task Trigger - This trigger creates a ContactChannel record for any inserted/updated task that with a CallObject field that does not currently have a ContactChannel record created before.

6. Select **Save**

Create the Softphone Layout

Next, we need to create a softphone layout for the solution. The softphone layout settings will tell the console what resources are available for screenpop by default and what to do under different match conditions.

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** box, type **Softphone**, then choose **Softphone Layouts** from the results
3. If you are presented with the Get Started message, choose **Continue**
4. On the Softphone Layouts page, choose **New**



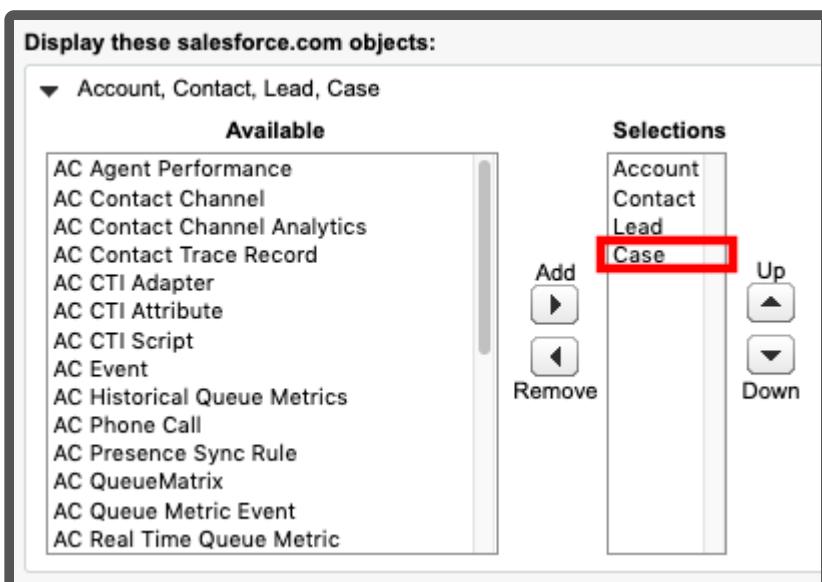
The screenshot shows the 'Softphone Layouts' page in the Salesforce setup. At the top, there's a header with tabs like 'Name', 'Default', 'Created By Alias', 'Created Date', 'Softphone Layout Assignment', 'Last Modified By Alias', and 'Last Modified Date'. Below the header, there's a message about softphones being customizable call control tools. A table below shows no records. The 'New' button is highlighted with a red box.

5. Enter a name for the layout, such as **AmazonConnectDefault**, then select the **Is Default Layout** checkbox.



The screenshot shows the 'Softphone Layout Edit' page. It has a title 'Softphone Layout Edit' and a sub-instruction: 'Each softphone layout allows you to customize the appearance of a softphone for inbound, outbound, or mobile calls.' Below this, there are 'Save' and 'Cancel' buttons. A form with a 'Name' field containing 'AmazonConnectDefault' and a checked 'Is Default Layout' checkbox is shown. This entire section is highlighted with a red box.

6. Expand **Display these salesforce.com objects** and select objects that CTI Connector should be able to search, for a screen-pop query. In this example, Case has been added to the default selection, allowing search and screen-pop by CaseID.



The screenshot shows the 'Display these salesforce.com objects' configuration screen. It has two main sections: 'Available' (left) and 'Selections' (right). The 'Available' section lists various objects like AC Agent Performance, AC Contact Channel, etc. The 'Selections' section shows objects selected for screenpop: Account, Contact, Lead, and Case. The 'Case' object is highlighted with a red box. Between the sections are 'Add' and 'Remove' buttons, and 'Up' and 'Down' buttons for reordering.

7. If desired, configure the search behavior to your requirements

The screenshot shows a list of search behavior configurations:

- If single Account found, display: Account Name
If multiple matches are found, only the Account Name is displayed in Salesforce Classic. In Lightning Experience, all the selected fields are displayed.
- If single Contact found, display: Name
If multiple matches are found, only the Name is displayed in Salesforce Classic. In Lightning Experience, all the selected fields are displayed.
- If single Lead found, display: Name
If multiple matches are found, only the Name is displayed in Salesforce Classic. In Lightning Experience, all the selected fields are displayed.
- If single Case found, display: Case Number
If multiple matches are found, only the Case Number is displayed in Salesforce Classic. In Lightning Experience, all the selected fields are displayed.

Each item has an 'Edit' link to its right.

8. Additionally, validate the Screen Pop settings. Please note that the default behavior is to not pop a screen if there is more than one result

The screenshot shows the 'Screen Pop Settings' configuration page with the following options:

- Screen pops open within: Existing browser window (Edit)
- No matching records: Don't pop any screen (Edit)
- Single-matching record: Pop detail page (Edit)
- Multiple-matching records: Pop to search page (Collapse)
 - Don't pop any screen
 - Pop to search page
 - Pop to Visualforce page
 - Pop to flow

9. Once you have configured the search behavior, choose **Save**

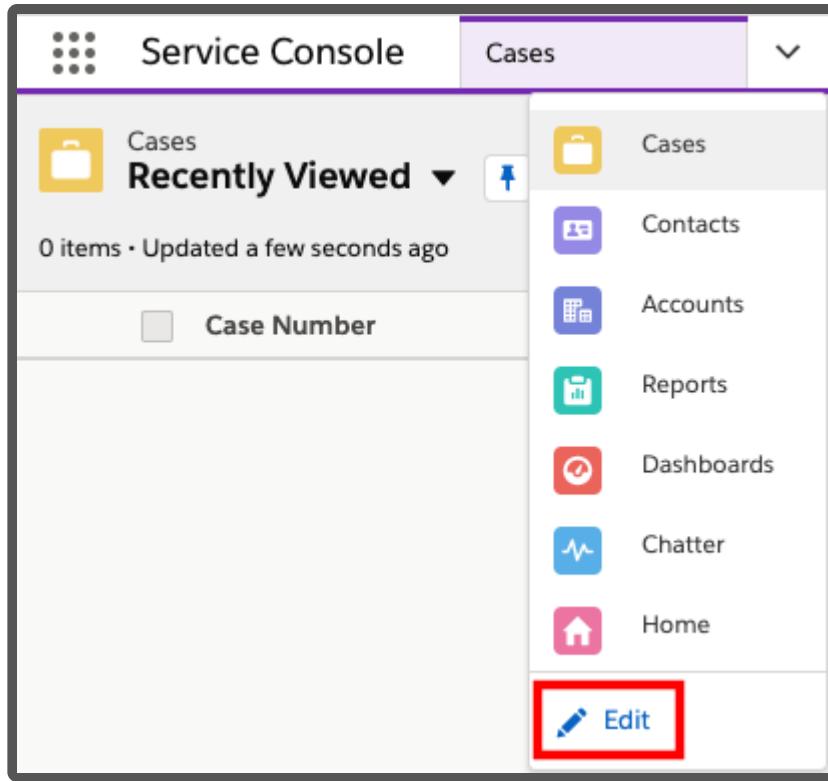
Initial CTI Adapter Configuration

Once we have setup the Call Center, we need to do a final configuration of the CTI Adapter before we can test the basic configuration. This will tie the Lightning CTI adapter settings to the Call Center.

Add the CTI Adapter Console App

1. Log in into your Salesforce org and go to the **Service Console**

2. Expand the **navigation menu** by selecting the down arrow and choose **Edit**.



3. On the Edit Service Console App Navigation Items page, select **Add More Items**

The screenshot shows the 'Edit Service Console App Navigation Items' page. The title is 'Edit Service Console App Navigation Items'. Below it, a message says 'Personalize your nav bar for this app. Reorder items, and rename or remove items you've added.' with a 'Learn More' link. The main area is titled 'NAVIGATION ITEMS (7)' and contains a list of items. In the bottom right corner, there is a button labeled 'Add More Items' which is highlighted with a red rectangular box.

4. Select the + next to **AC CTI Adapters** and select the **Add 1 Nav Item** button

Add Items

AVAILABLE ITEMS All 1

Search all items...

AC CTI Adapters X
1 item selected

AC CTI Adapters

AC Contact Channel Analytics

AC Contact Trace Records

AC Queue Metrics

AC Real Time Queue Metrics

App Launcher

Approval Requests

Assets

Authorization Form

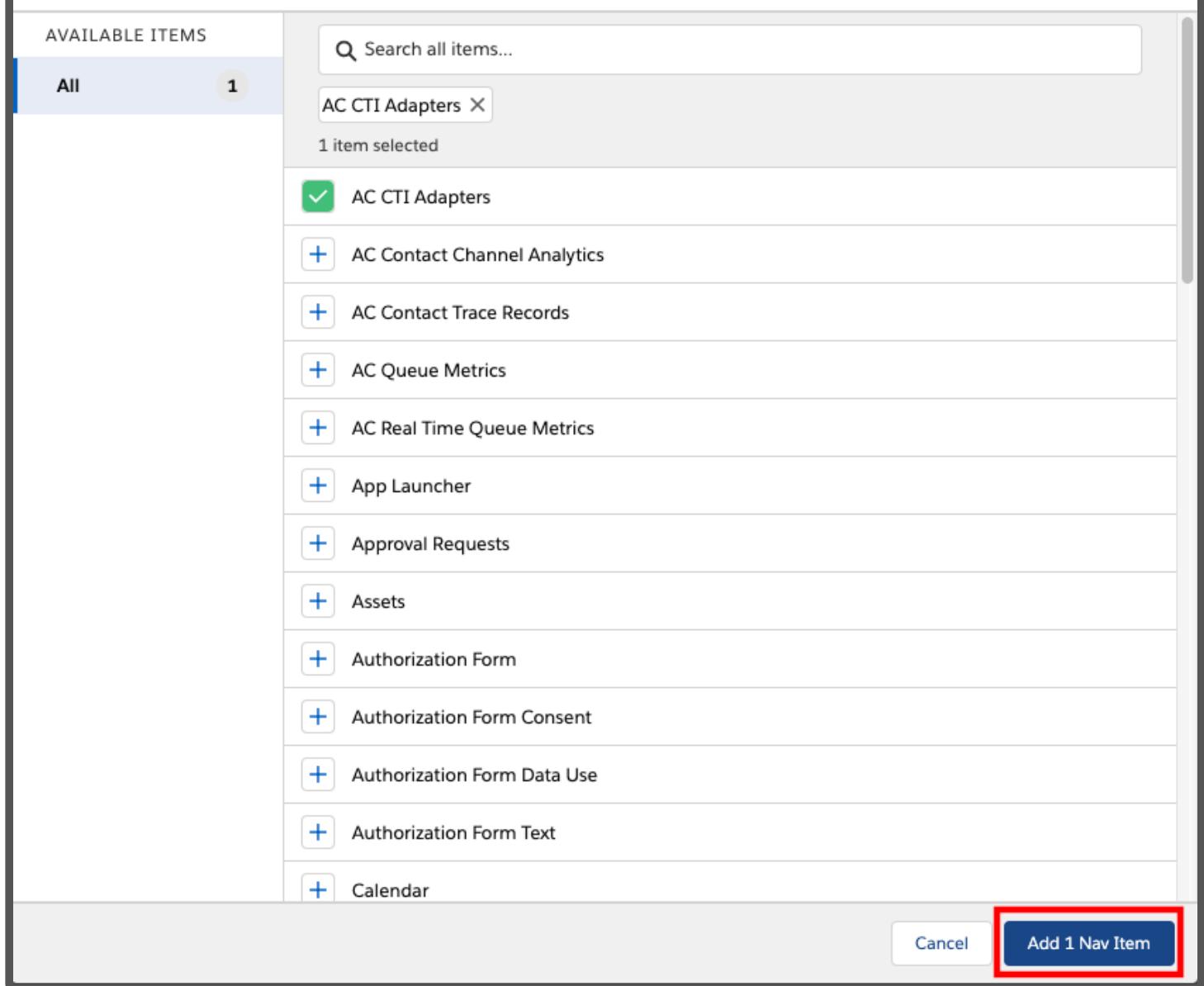
Authorization Form Consent

Authorization Form Data Use

Authorization Form Text

Calendar

Cancel Add 1 Nav Item



5. If desired, move the **AC CTI Adapters** button up in the navigation Items menu by dragging it up or down the list, then choose **Save** to save changes
6. Select **AC CTI Adapters** from navigation menu
7. If Recently Viewed is selected, select the drop-down and select **All** from the List Views menu.

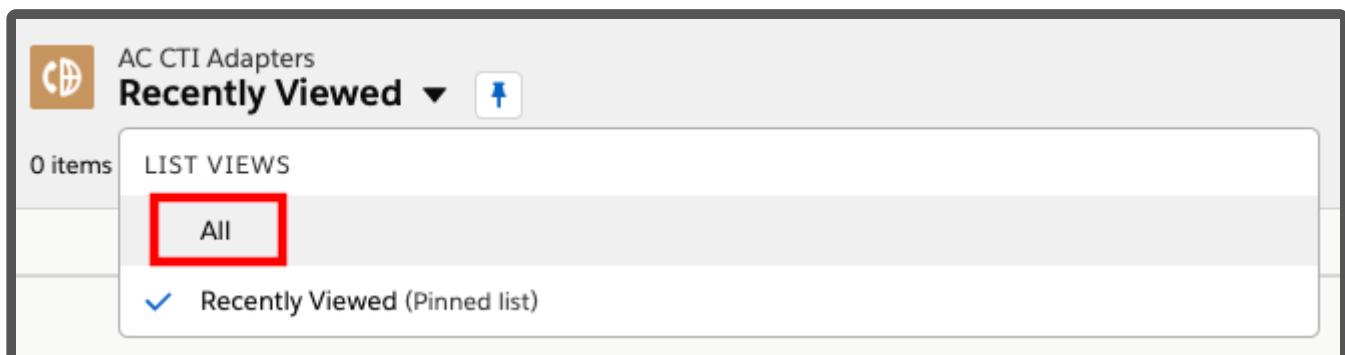
AC CTI Adapters

Recently Viewed ▾ 

0 items LIST VIEWS

All

Recently Viewed (Pinned list)



8. If no AC Lightning Adapter entry exists, then select the new button to configure your AC CTI adapters, otherwise select the **AC Lightning Adapter**
9. Fill out or confirm the Details as follows:
10. CTI Adapter Name: **AC Lightning Adapter**
11. Amazon Connect Instance: The url of your Amazon Connect Instance. You can find this in the Amazon Connect Console as shown below (if your Amazon Connect instance uses the `https://(instancename).awsapps.com/connect/login` domain, then remove everything after ".com"):

Account overview

Access information

Access URL
<https://guidedsetuptest-instance-w3dgh2.my.connect.aws>

12. Amazon Connect Instance Region: This is the region that your Amazon Connect instance is deployed in. For this field, you will enter the region code. For example, if you have deployed your Amazon Connect instance in US East (N. Virginia), you would enter us-east-1. For a list of region codes, please refer to the [AWS Service Endpoints](#) reference
13. Call Center Definition Name: **AC Lightning Adapter Note:** This is the value of the Internal Name in the call center in the Call Center definition
14. Leave all other settings at the default for now, and choose Save

The screenshot shows the 'AC CTI Adapter' configuration page. The adapter name is 'ACLightningAdapter'. The 'Amazon Connect Instance' field contains the URL 'https://[REDACTED].awsapps.com/'. The 'Custom Ringtone' field is empty. Under 'Softphone Popout Enabled', there is a checked checkbox. The 'Medialess' field has an unchecked checkbox. On the right side, there are sections for 'Owner' (with a user icon and redacted email), 'Amazon Connect Instance Region' (set to 'us-east-1'), 'Call Center Definition Name' (set to 'ACLightningAdapter'), 'Debug Level' (set to 'Off'), and 'Presence Sync Enabled' (which is checked).

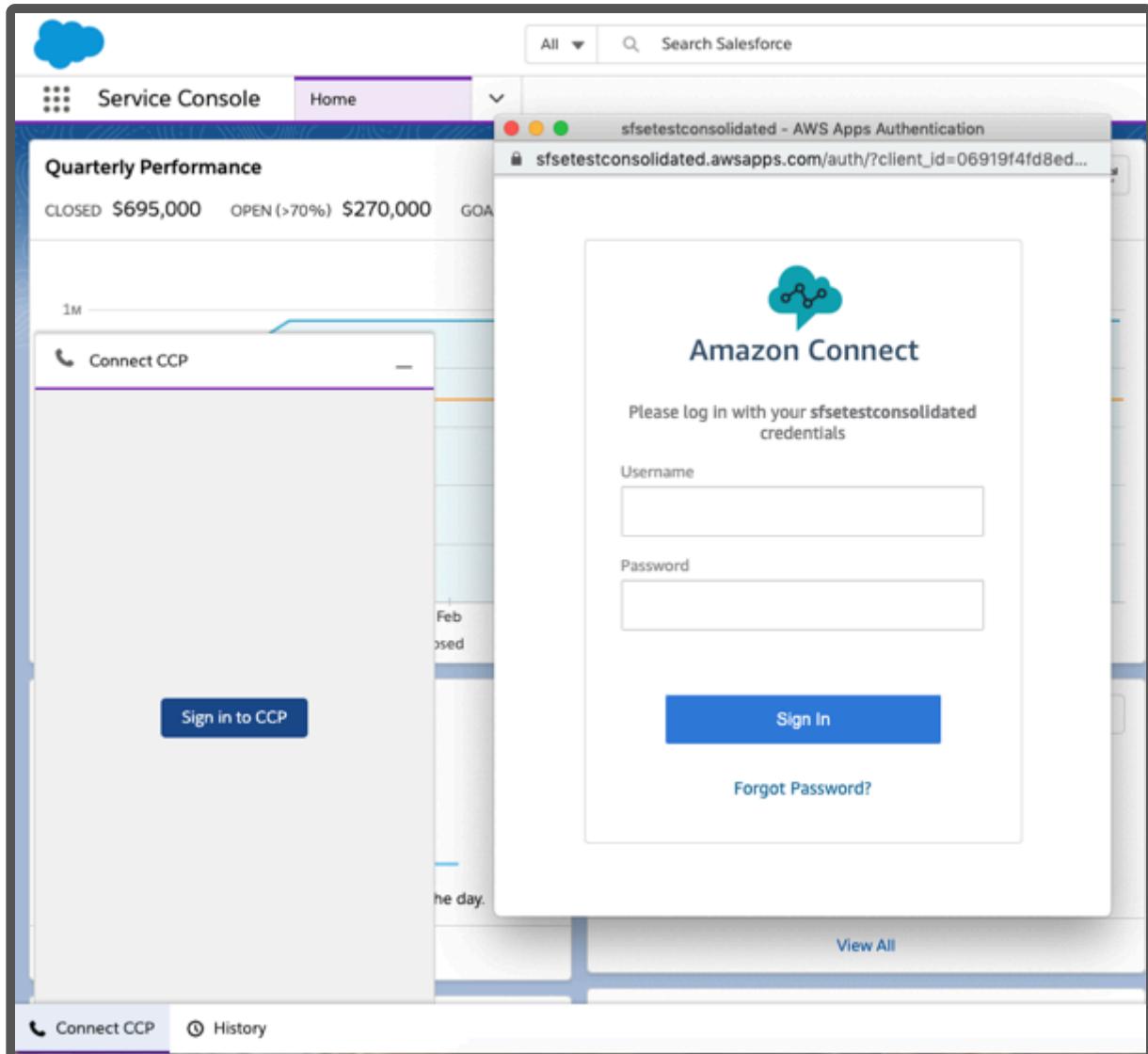
15. Refresh the browser

16. In the bottom left corner of the Service Console, select the CTI Softphone icon



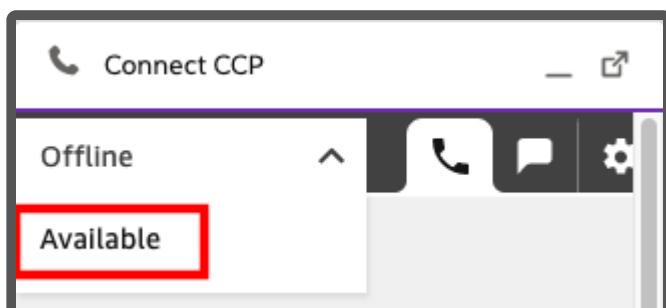
17. Select the **Sign in to CCP** button. A new window will pop up. Enter your Amazon Connect login credentials and select **Sign In**. Make sure to allow Microphone access (if asked by browser)

NOTE: At this point, this process will only work for Amazon Connect instances configured for local user storage. If you are configuring SAML, please follow the SAML setup process in the [Single Sign On Settings](#) section before continuing.

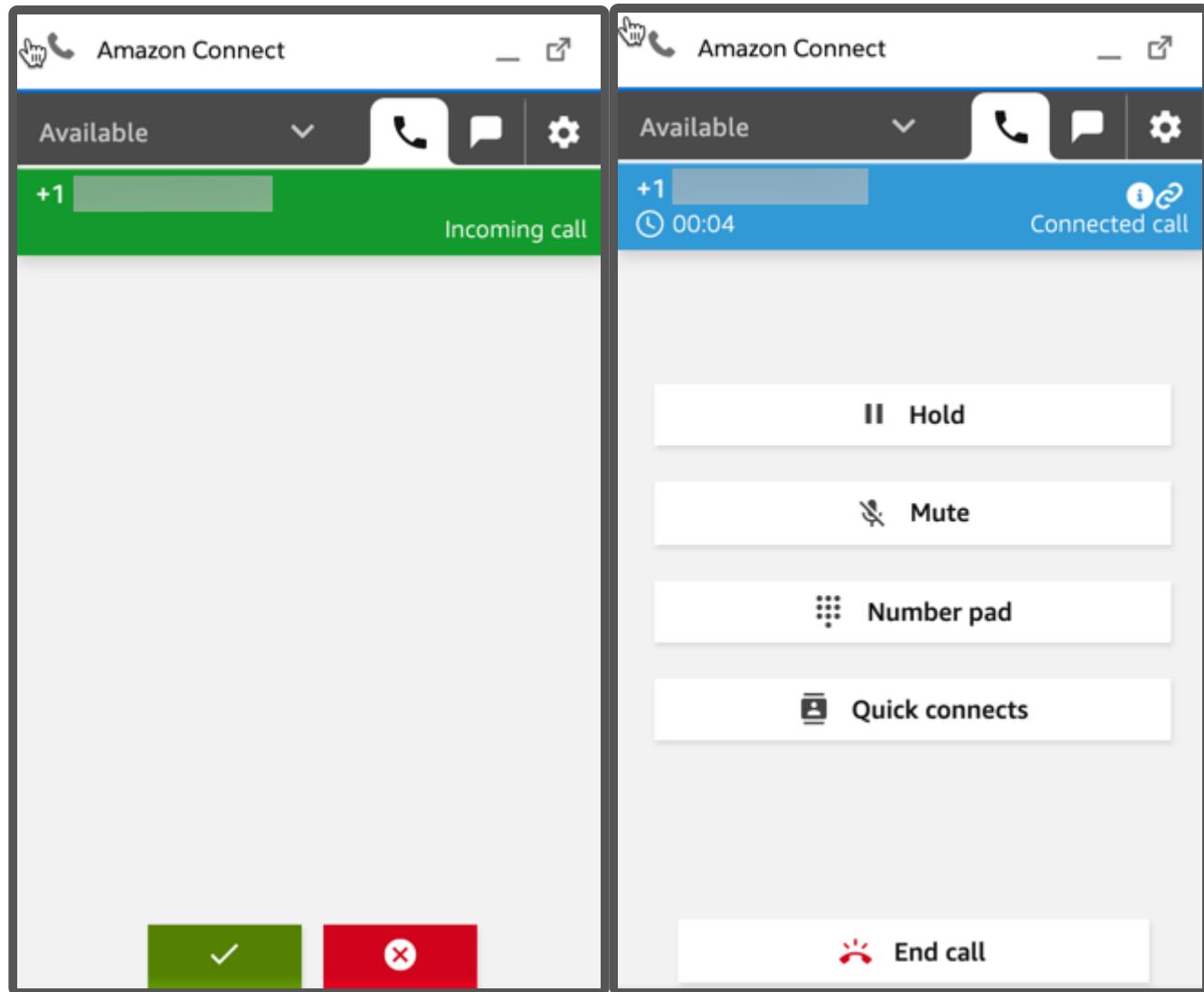


18. Once Login is successful, the pop-up window will automatically close.

19. Expand the status menu and choose Available



20. Make an inbound phone call to your Amazon Connect instance. The CCP will alert you to the incoming call and allow you to accept it. Once you do, the call will be connected



21. End the call and clear the contact

22. Set your agent back to **Available**

Enhanced Agent Logout

You can configure an agent status within "Manage agent status" with "Logout" (case-sensitive) in the status name to enable enhanced agent logout. When the agent selects that logout status in the Contact Control Panel, it will first set the agent in an offline status. It will then logout the agent in Connect and the AWS Console. Here is an example of the agent status configured within Connect:

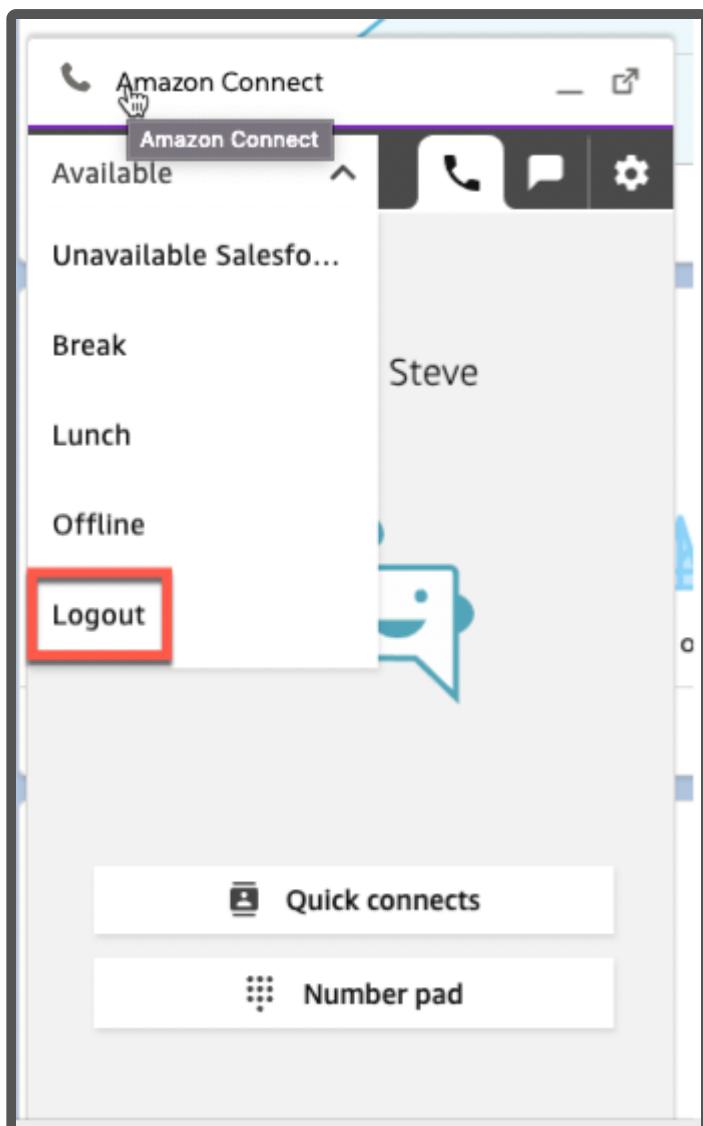
Manage agent status

Create new agent status, and drag table rows to specify the order these statuses appear in the Contact Control Panel (CCP). To maintain integrity of historical metrics, agent status cannot be deleted. However, they can be disabled so that they no longer show in the CCP.

Status name	Description	Type	Enabled for use in CCP
Unavailable Salesforce	Unavailable Salesforce	Custom	<input checked="" type="checkbox"/>
Break	Break	Custom	<input checked="" type="checkbox"/>
Lunch	Lunch	Custom	<input checked="" type="checkbox"/>
Available	Available state	Routable	<input checked="" type="checkbox"/>
Offline	Offline state	Offline	<input checked="" type="checkbox"/>
Logout	Sets the Connect user to offline and then completes logs out the Connect user	Custom	<input checked="" type="checkbox"/>

Save Cancel

Here is an example of an agent selecting the "Logout" status within the Contact Control Panel:

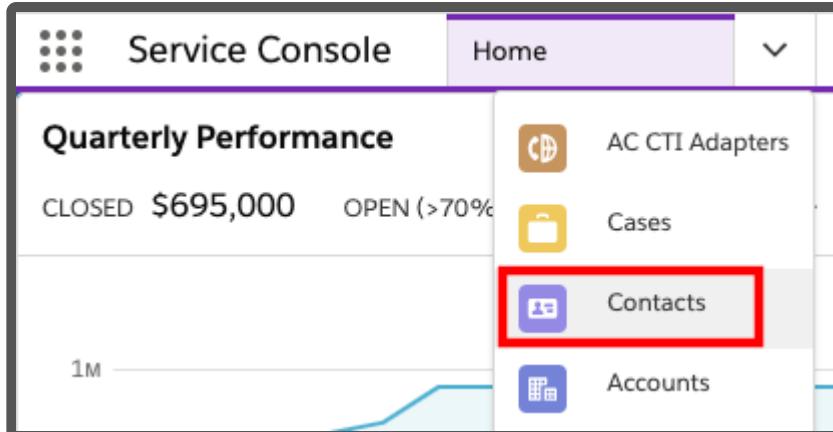


See how you can add custom agent statuses [here](#).

Validate Basic Screenpop

Next, we will add a contact to Salesforce that has your phone number assigned to it. This will allow us to validate the basic screenpop functionality that is provided with the CTI adapter.

1. Select **Contacts** from the dropdown menu



2. Select **New** from top-right corner

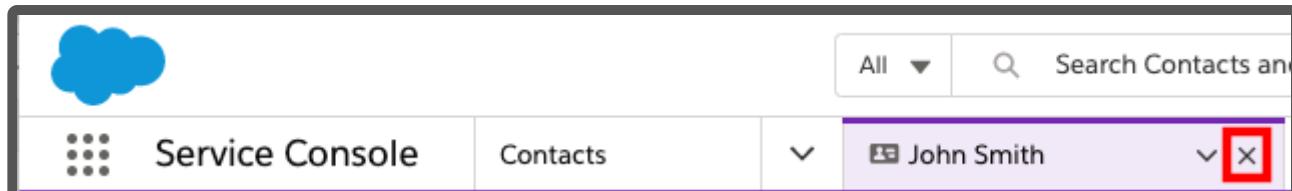
3. Complete the required fields. Make sure that your phone number is entered for the Phone field.

The screenshot shows the 'New Contact' form. The 'Contact Information' section is visible. Required fields are marked with an asterisk (*). The 'Name' field (Salutation: Mr., First Name: John) and the 'Last Name' field (Smith) are grouped together and highlighted with a red box. The 'Phone' field (containing the value 7048076561) is also highlighted with a red box. Other fields shown include Account Name, Mobile, Title, Other Phone, Department, Fax, Birthdate, and Email. At the bottom, there are 'Cancel', 'Save & New', and 'Save' buttons.

Contact Information	
Contact Owner	Jason Douglas
* Name	John Smith
Salutation	Mr.
First Name	John
* Last Name	Smith
Account Name	Search Accounts... <input type="button" value="🔍"/>
Mobile	<input type="text"/>
Title	<input type="text"/>
Other Phone	<input type="text"/>
Department	<input type="text"/>
Fax	<input type="text"/>
Birthday	<input type="text"/>
Email	<input type="text"/>
<input type="button" value="Cancel"/> <input type="button" value="Save & New"/> <input type="button" value="Save"/>	

4. Select **Save**

5. Close the Contact tab by selecting the X next to the name of the contact that you just created



6. **Refresh** your browser

7. Place another phone **call** into your instance

8. The new contact should automatically pop-up as it has been recognized by incoming phone number.

Setting Up The Salesforce Lambdas Manually

Below are manual setup instructions for the Salesforce Lambdas.

Prerequisite Configuration and Data Collection

In order to successfully deploy and utilize the functions in the Amazon Connect Salesforce Lambda package, you will need to validate and configure some items in your Salesforce Org and gather some information from your Amazon Connect instance.

- Check your Salesforce API version
- Create a new Connected App
- Create a new API user
- Gather Amazon Connect information

As you are preparing to deploy the package, it is a good idea to open a text editor and note information as you configure the environment. We will point out the items you will need to provide.

Check your Salesforce API Version

1. Log in into your Salesforce org and go to **Setup**

2. In the Quick Find field, type **apex**, then select **Apex Classes** from the results

The screenshot shows the Salesforce search interface. A search bar at the top contains the text "apex". Below the search bar, there is a tree view of search results. Under the "Custom Code" category, the "Apex Classes" item is highlighted with a red box. Other items in this category include "Apex Settings", "Apex Test Execution", "Apex Test History", and "Apex Triggers".

3. Select New

The screenshot shows the Apex Classes list page. At the top, there is a navigation bar with links for "Developer Console", "New" (which is highlighted with a red box), "Generate from WSDL", "Run All Tests", and "Schedule Apex". Below the navigation bar is a table header row with columns: Action, Name ↑, Namespace Prefix, Api Version, Status, Size Without Comments, Last Modified By, and Has Trace Flags. The "New" button is located in the "Action" column of the first row.

4. Select the Version Settings tab

The screenshot shows the Apex Class Edit page. The title is "Apex Class". Below the title, there is a sub-header "Apex Class Edit" with three buttons: "Save", "Quick Save", and "Cancel". There are two tabs: "Apex Class" (selected) and "Version Settings" (highlighted with a red box). Below the tabs is a toolbar with icons for search, create, refresh, and font size. The main area shows a table with one row, indicated by the number "1" in a yellow-highlighted cell.

5. Note the Salesforce.com API version in your notepad. The pattern of this value is **vXX.X**.

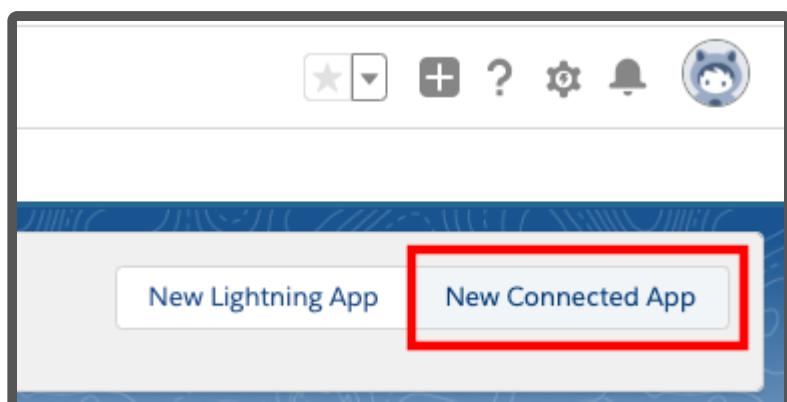
The screenshot shows the Apex Class Edit page with the "Version Settings" tab selected. The title is "Apex Class". Below the title, there is a sub-header "Apex Class Edit" with three buttons: "Save", "Quick Save", and "Cancel". There are two tabs: "Apex Class" (selected) and "Version Settings". The main area shows a table with two rows. The first row has a "Name" column with "Salesforce.com API" and a "Version" column with a dropdown menu set to "47.0". The second row has a "Name" column with "Amazon Connect - Universal Package" and a "Version" column with a dropdown menu set to "4.2".

Create a New Connected App

To leverage the full potential of the integration, Salesforce data needs to be accessed from AWS environment. The package comes with a set of pre-built AWS Lambda functions to lookup, update and create Salesforce objects within Amazon Connect Contact Flows. These Lambda function access Salesforce using the Salesforce REST API.

To get access to the environment, a Connected App must be configured with OAuth settings enabled.

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** field, type **app manager**, then select **App Manager** from the results
3. In the upper right corner, select **New Connected App**



4. On the New Connected App form, enter a name for the Connected App, such as **Amazon Connect Integration** and press tab. This will populate the API Name automatically. Then provide a contact email address

A screenshot of the 'New Connected App' configuration form. The title bar says 'New Connected App' and has 'Save' and 'Cancel' buttons. Below is a section titled 'Basic Information' with three fields:

Connected App Name	Amazon Connect Integration
API Name	Amazon_Connect_Integration
Contact Email	dougjaso@amazon.com

The entire input area is highlighted with a red border.

5. Select the checkbox to **Enable OAuth Settings**

A screenshot of the 'API (Enable OAuth Settings)' section. It shows a dropdown menu with the option selected and a checkbox labeled 'Enable OAuth Settings' which is checked with a blue checkmark.

6. Set the **Callback URL** to your domain url. Find the domain at *Setup -> My Domain*.

Enable OAuth Settings

Enable for Device Flow

Callback URL https://[REDACTED].my.salesforce.com

7. In the Selected OAuth Scopes section, select the following and add them to the Selected OAuth Scopes:

8. Access the identity URL service (id, profile, email, address, phone)

9. Manage user data via APIs (api)

10. Select the checkbox for Require Secret for Web Server Flow, and the checkbox for Require Secret For Refresh Token Flow

11. The **API (Enable OAuth Settings)** section should now look like this

API (Enable OAuth Settings)

Enable OAuth Settings

Enable for Device Flow

Callback URL https://www.salesforce.com

Use digital signatures

Selected OAuth Scopes

Available OAuth Scopes	
<input type="checkbox"/>	Access Analytics REST API Charts Geodata resources (clair_api)
<input type="checkbox"/>	Access Analytics REST API resources (wave_api)
<input type="checkbox"/>	Access Connect REST API resources (chatter_api)
<input type="checkbox"/>	Access Lightning applications (lightning)
<input type="checkbox"/>	Access Visualforce applications (visualforce)
<input type="checkbox"/>	Access chatbot services (chatbot_api)
<input type="checkbox"/>	Access content resources (content)
<input type="checkbox"/>	Access custom permissions (custom_permissions)
<input type="checkbox"/>	Access unique user identifiers (openid)
<input type="checkbox"/>	Full access (full)

Add

Remove

Selected OAuth Scopes

Access the identity URL service (id, profile, email, address, phone)
Manage user data via APIs (api)

Require Secret for Web Server Flow

Require Secret for Refresh Token Flow

Introspect All Tokens

Configure ID Token

Enable Asset Tokens

Enable Single Logout

12. Select **Save** at the bottom of the screen.

13. Select **Continue** on the New Connected App page

14. You should now be at the new app's page

15. Copy the value for **Consumer Key** to your notepad

16. Select **Click to reveal** next to Consumer Secret and copy the value to your notepad

17. At the top of the detail page, select **Manage**

18. On the Connected App Detail page, select the **Edit Policies** button
19. Set Permitted Users to **Admin approved users are pre-authorized** and choose OK on the pop-up dialog
20. Set IP Relaxation to **Relax IP restrictions**
21. The OAuth Policies section should now look like the following

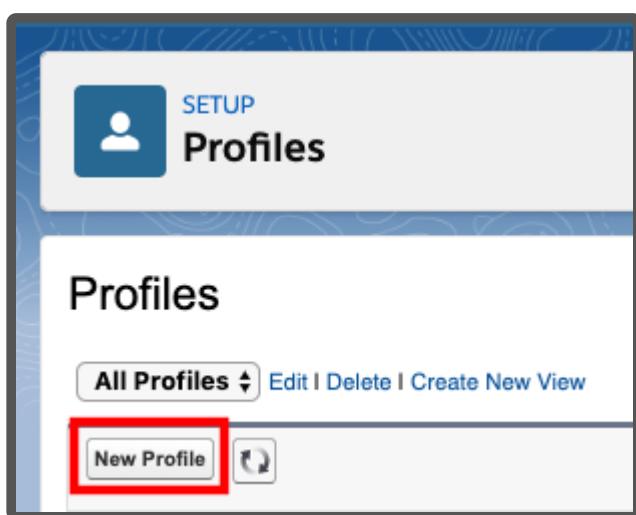


22. Select **Save**

Create a new API user

The Lambda functions authenticate with Salesforce via user credentials. It is a common practice to create an API user account for this purpose.

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** field, type **profiles**, then select **Profiles** from the results
3. Select New Profile



4. Provide a Profile Name, such as **API_ONLY**
5. From the **Existing Profile** dropdown, select **System Administrator** **NOTE:** You're advised to use a full Salesforce License for the user to be able to set the below permissions and have full access to avoid any other errors.

Clone Profile

Enter the name of the new profile.

You must select an existing profile to clone from.

Existing Profile	<input type="text" value="System Administrator"/> <input type="button" value="▼"/>
User License	Salesforce
Profile Name	<input type="text" value="API_ONLY"/>

6. Select **Save** to create the new profile

7. Once the new profile page opens, scroll down to and select the **System Permissions** section

System

System Permissions

Permissions to perform actions t

8. When the next page opens, select **edit**

9. Make sure the **Lightning Experience User** option is unselected

Lightning Experience User



10. Select **Save**, and confirm the changes

11. Go back to the Profile Overview, scroll down, and select **Password Policies**

System

Settings that apply across all apps, such as record and user management
[Learn More](#)

System Permissions

Permissions to perform actions tha

Login Hours

Settings that control when users ca

Login IP Ranges

Settings that control the IP address

Service Providers

Permissions that let users switch to

Session Settings

Settings that control required sessi

Password Policies

Profile Based password policies

Default Experience

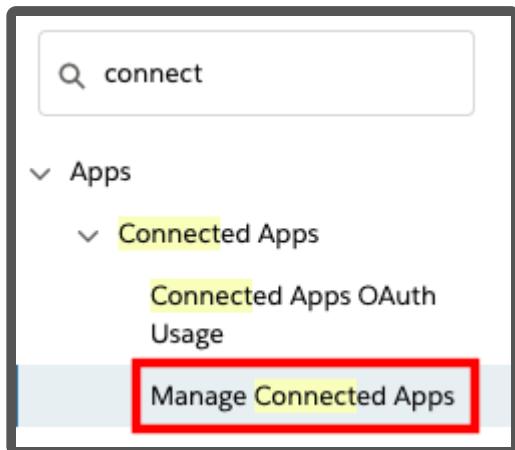
Setting for assigning a default com

12. Select **Edit**.

13. Set **User passwords expire in** to **Never expires** NOTE: Failure to this may lead to production outages.

14. Select **Save**.

15. In the **Quick Find** field, type **connect**, then select **Manage Connected Apps** from the results



16. Select the app you have created earlier, **Amazon Connect Integration**

17. In the profiles section, select **Manage Profiles**

18. Select the new **API_Only** profile that you just created

19. Select **Save** at the bottom of the page

20. In the **Quick Find** field, type **users** then select **Users** from the results

21. Select New User

22. Set the required fields as:

- a. Last Name: apiuser
- b. Alias: apiuser
- c. Email: provide a valid email address
- d. Username: apiuser@<yoursalesforcedomain>.com
- e. Nickname: apiuser

23. On the right-hand side, set **User License** to **Salesforce**

24. Set Profile to API_ONLY

25. Choose **Save**

26. In **Quick Find**, search for "Permission Sets". Select the **AC_Administrator** permission set.

The screenshot shows the Salesforce Setup interface. The left sidebar has sections for Users (Permission Set Groups, **Permission Sets**), Custom Code (Custom Permissions), and Global Search. The main content area is titled "Permission Sets" and contains a table of permission sets. One row, "AC Administrator", is highlighted with a red box. The table columns are Action, Permission Set Label, Description, and License. The "Description" for AC Administrator states: "Allows the user to configure Amazon Connect setup and provides ...".

Action	Permission Set Label	Description	Licenses
<input type="checkbox"/>	Clone	AC Administrator	Allows the user to configure Amazon Connect setup and provides ...
<input type="checkbox"/>	Clone	AC Agent	
<input type="checkbox"/>	Clone	AC_CallRecording	
<input type="checkbox"/>	Clone	AC_Manager	

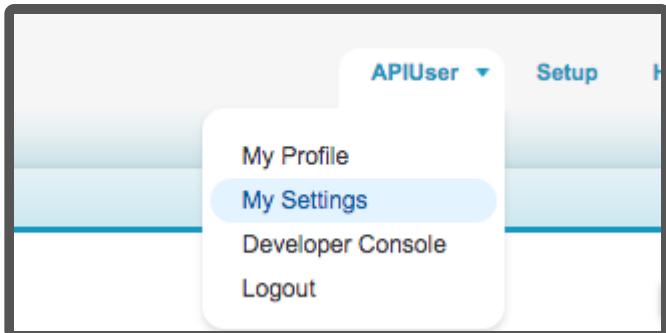
27. Select **Manage Assignments**. Add the apiuser you just created to the permission set.

28. A confirmation email with an **activation link** will be sent to the email address provided. Choose the link to activate your user and set their password

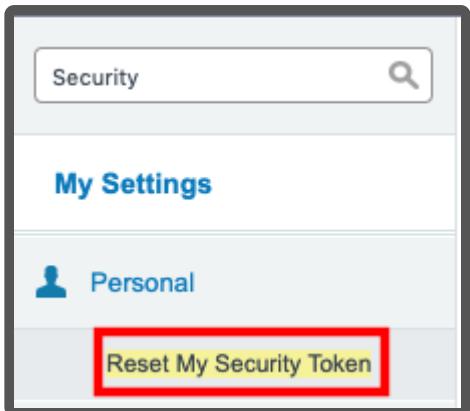
29. Fill out the form to set a password for the API user

30. Select **Change Password**. The API user will log into the Salesforce Classic view

31. Access the API user's personal settings by selecting the username in the top right corner, then choose **My Settings**



32. In the **Quick Find** field, type **security** then select **Reset My Security Token** from the results

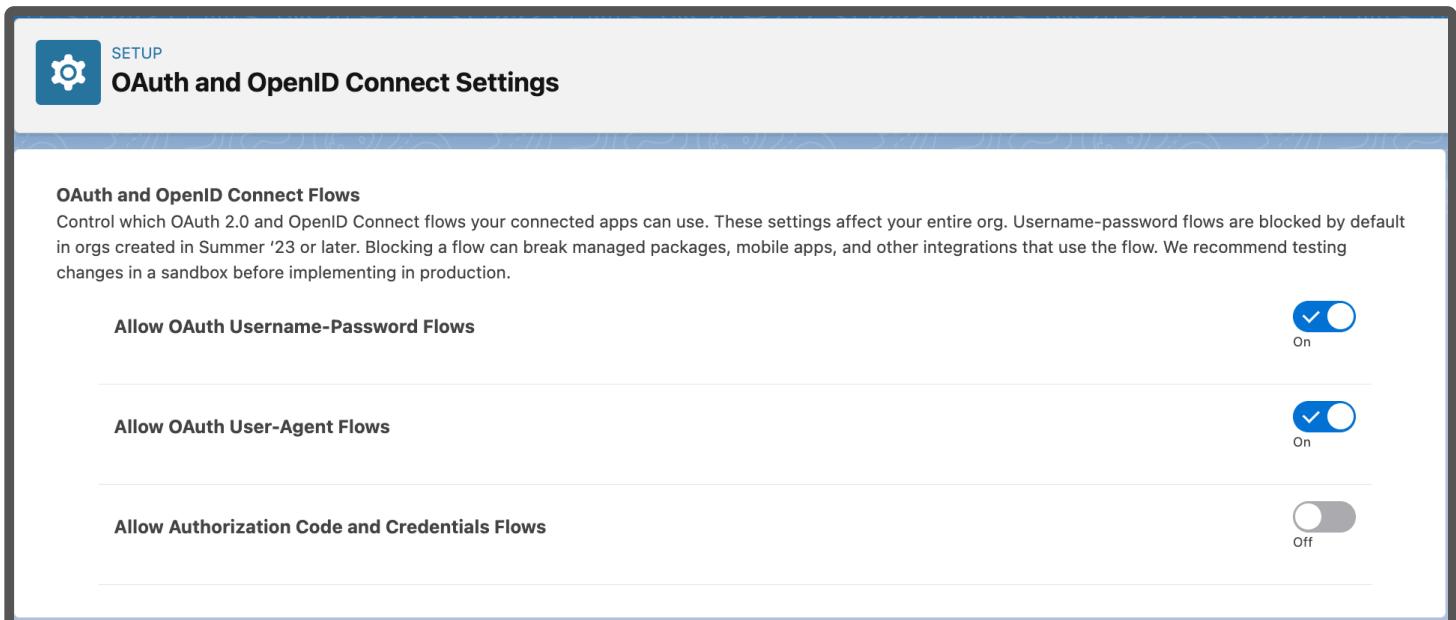


33. Select **Reset Security Token**. Your security token will be emailed to you

34. Copy the security token from the email to your notepad

Allowing the API user to authenticate using password

The api user created above authenticates using username-password flow in Salesforce. This flow needs to be unblocked and to do that, go to *Setup* and in the Quick Find box, search for **OAuth and OpenID Connect Settings**. After that, make sure that the toggles for **Allow OAuth Username-Password Flows** and **Allow OAuth User-Agent Flows** are turned ON, as shown in below image.



Gather Amazon Connect information

The last thing to do before you can install the Amazon Connect Salesforce Lambda Package is gather some details about your Amazon Connect instance. These will be used during the package installation.

1. In a new browser tab, login to the [AWS console](#)
2. Navigate to the [Amazon Connect Console](#)
3. Select your Instance Alias
4. On the Overview page for your instance, copy the string following instance/ in the Instance ARN and paste it to your notepad. This is your Instance ID.

Distribution settings

Instance ARN

arn:aws:connect:us-west-2:YOUR-ACCOUNT-ID:instance/YOUR-INSTANCE-ID-XXX-XXXXXXX

5. In the left nav, select **Data storage**
6. On the **Data storage** page, copy the S3 bucket names for your Call recordings and Exported Reports. The bucket name is everything preceding the first / in the XX will be stored here sections

Data storage

Saving Amazon Connect data such as call recordings or scheduled reports required by Connect is reflected below.

Call recordings

Call recordings will be stored in this S3 bucket

YOUR_BUCKET_NAME/CallRecordings

Chat transcripts

Chat transcripts will be stored in this S3 bucket

YOUR_BUCKET_NAME/ChatTranscripts

Live media streaming

Not enabled

Exported reports

Exported reports will be stored in this S3 bucket

YOUR_BUCKET_NAME/Reports

7. In the left nav, select **Data streaming**

8. Note the name of the Kinesis stream configured in the Contact Trace Records section, then select **Create a new Kinesis Stream**. This will take you to the list of Kinesis streams configured in this region.

9. Select the **Kinesis stream name** that matches what was configured in the previous step

10. On the stream detail page, copy the entire value for Stream ARN

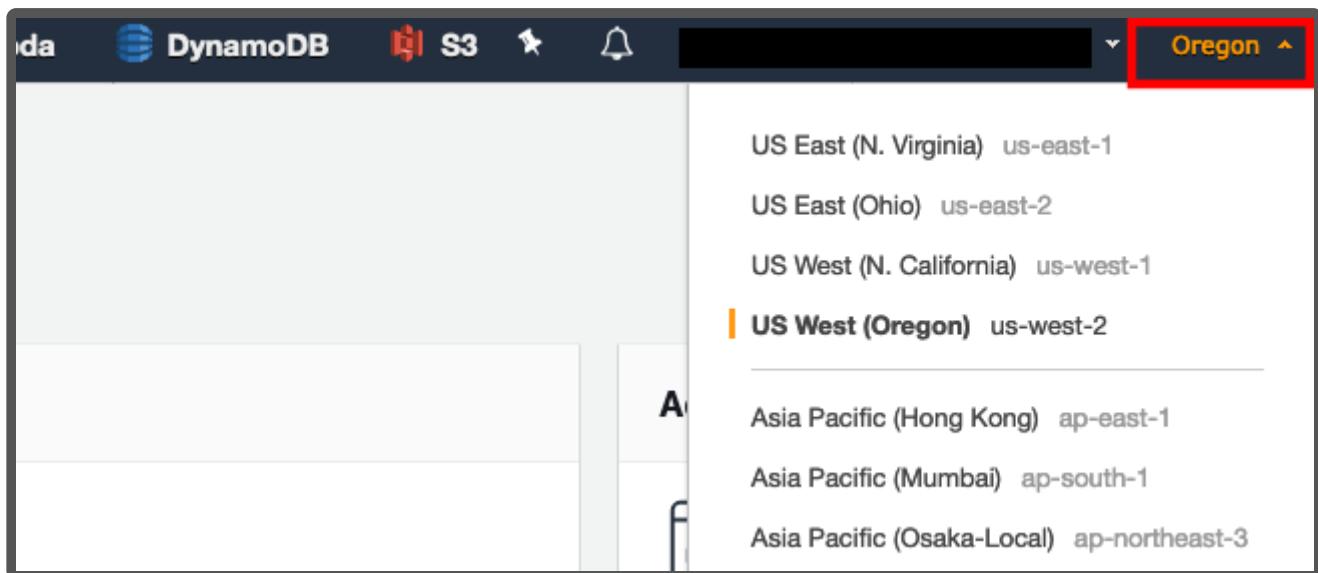
Stream ARN `arn:aws:kinesis:us-east-1:YOUR_ACCOUNT_NUMBER:stream/YOUR_STREAM_NAME`

Status ACTIVE

Store Salesforce Credentials in AWS Secrets Manager

To ensure that your Salesforce credentials are secure, the Lambdas require that the credentials are stored in AWS Secrets Manager. AWS Secrets Manager is a highly secure service that helps you store and retrieve secrets.

1. In a new browser tab, login to the AWS console
2. Make sure you are in the same region as your Amazon Connect instance. You can set the region by expanding the region selector in the upper right and choosing the region



3. Navigate to the [Secrets Manager console](#)
4. Select **Secrets**
5. Select **Store a new secret**
6. Select **Other types of secrets**
7. Make sure **Secret key/value** is selected
8. Enter key value pairs that match the following:
 - a. **Key:** Password, **Value:** the password for the API user that you configured in the previous section

- b. **Key:** ConsumerKey, **Value:** the Consumer Key for the Connected App you created in the previous section
- c. **Key:** ConsumerSecret, **Value:** the Consumer Secret for the Connected App you created in the previous section
- d. **Key:** AccessToken, **Value:** this is the access token for the API user that you configured in the previous section

9. For the encryption key, click **Add new key**

10. Select **Create Key**

11. Make sure key type is set to **symmetric**

12. Give your key an **alias**, like *SalesforceCredentialsSecretsManagerKey*

13. Click Next

14. Select administrators you want to have access permission to change the key policy. Make sure you are being as restrictive as possible

15. Click Next

16. Select the users and roles you want to have access to the Salesforce credentials in Secrets Manager. Make sure you are being as restrictive as possible

17. Click Next

18. Click Finish

19. Click on the managed key that you just created (which is *SalesforceCredentialsSecretsManagerKey* in this case).

20. Note down the ARN. This is *SalesforceCredentialsKMSKeyARN* that will be used later when installing the Amazon Connect Salesforce Lambda package.

21. Navigate back to the Secrets Manager setup tab

22. Select the key you just created

Specify the key/value pairs to be stored in this secret [Info](#)

Secret key/value

Plaintext

Password

Password

Remove

ConsumerKey

ConsumerKey

Remove

ConsumerSecret

ConsumerSecret

Remove

AccessToken

AccessToken

Remove

[+ Add row](#)

Select the encryption key [Info](#)

Select the AWS KMS key to use to encrypt your secret information. You can encrypt using the default service encryption key that AWS Secrets Manager creates on your behalf or a customer master key (CMK) that you have stored in AWS KMS.

SalesforceCredentialsSecretsManagerKey



[Add new key](#)

Cancel

Next

23. Click Next

24. Give your secret a name, like *SalesforceCredentials*

25. Click Next

26. Make sure **automatic rotation** is disabled.

27. Click Next

28. Click Store

29. Select the secret you just created, and copy the Secret ARN

SalesforceCredentials

Secret details		Actions ▾
Encryption key	SalesforceCredentialsSecretsManagerKey	
Secret name	SalesforceCredentials	
Secret ARN		
Secret description	-	

30. You should now have all of the information you need to install the package

Install the Amazon Connect Salesforce Lambda package

Compatibility Table

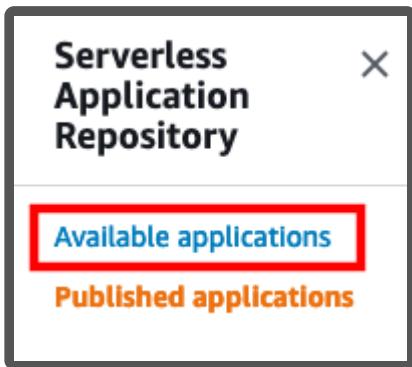
The following table instructs users on the best CTI Adapter version to use with the corresponding lambda version. If a minor version for the CTI Adapter is not listed (ex. v5.21.1), it will be grouped with its major version unless otherwise specified

CTI Adapter Version	Lambda Version
v5.22	v5.22
v5.21	v5.19 - v5.19.7
v5.20	v5.19 - v5.19.7
v5.19	v5.19 - v5.19.7
v5.18	v5.18
v5.17	v5.17
v5.16	v5.16
v5.15	v5.15
v5.14	v5.14

v5.13	v5.13
v5.12	v5.11 - v5.12
v5.11	v5.11 - v5.12
v5.10	v5.10
v5.9	v5.9
v5.7	v5.7 - v5.8
v5.6	v5.7 - v5.8

Instructions

1. In a new browser tab, login to the [AWS console](#)
2. Make sure you are in the same region as your Amazon Connect instance
3. Once you have selected the region, navigate to the [Amazon Connect Console](#)
4. Verify that the Amazon Connect instance that you wish to configure is listed
5. Once you have verified your Amazon Connect instance, Open the [Serverless Application Repository Console](#)
6. In the left navigation, select **Available Applications**



7. In the search area, make sure that **Public applications** is selected, check the box for **Show apps that create custom IAM roles or resource policies**, and enter **Salesforce** in the search field, this will automatically filter the available packages

Public applications (4)

Private applications

Salesforce

X

Show apps that create custom IAM roles or resource policies

8. Select AmazonConnectSalesForceLambda

The screenshot shows the AWS Lambda console interface. At the top, there are tabs for 'Public applications (4)' and 'Private applications'. A search bar contains the text 'Salesforce'. Below the search bar is a checkbox labeled 'Show apps that create custom IAM roles or resource policies'. The main area displays a list of functions:

- Salesforce-API-Access-Manager-Monitor-Logger**: Description: A simple API access manager built on AWS lambda to provide multi tiered access to salesforce services with a single API user. Please read more here: <https://github.com/manjits190/Salesforce-API-Access-Manager-Monitor-Logger/blob/master/README.md>. Tags: salesforce-api-access-manager. Deployment count: 26.
- AmazonConnectSalesForceLambda**: Description: Creates custom IAM roles or resource policies. The AWS Serverless application package contains a set of common Lambda functions to be used by Amazon Connect to interact with Salesforce, allowing lookup, create and update operations for different Salesforce objects, like Contacts and Cases. Tags: Integration, Connect, Amazon, Salesforce. Deployment count: 685.
- alexa-salesforce-notes-sample**: Description: This skill demonstrates how to build a private Alexa skill to access Salesforce data. This skill identifies a given opportunity, tracks a series of statements that a user gives, and posts those either as a note or as a Chatter post. Tags: alexa, alexa-for-business, alexa. Deployment count: 46. Author: Alexa for Business (AWS verified author).

9. When the Application loads, scroll down to the **Application settings** section

10. Fill in the parameters using the data you gathered in your notepad in the previous section using the following notes:

- i. **Application name:** You can accept the default here or change it as desired
- ii. **AmazonConnectInstanceId:** Your Amazon Connect Instance Id. Only required if you enable real time reporting
- iii. **CTRKinesisARN:** This is the ARN for the Kinesis stream that was configured for Contact Trace Record streaming in Amazon Connect. This is the complete ARN. Amazon Kinesis Firehose is not supported.
- iv. **ConnectReportingS3BucketName:** This is the name of the S3 bucket used to store exported reports for your Amazon Connect instance. This is ONLY the bucket name, no sub-folders or suffixes

- v. **HistoricalReportingImportEnabled:** true | false - if set to true, the package will include a feature to import Amazon Connect Queue and Agent Historical Metrics into your Salesforce Org. This feature requires you to provide **ConnectReportingS3BucketName**
- vi. **LambdaLoggingLevel:** DEBUG | INFO | WARNING | ERROR | CRITICAL - Logging level for Lambda functions
- vii. **PrivateVpcEnabled:** Set to true if functions should be deployed to a private VPC. Set VpcSecurityGroupList and VpcSubnetList if this is set to true.
- viii. **RealtimeReportingImportEnabled:** true | false - if set to true, the package will include a feature to publish Amazon Connect Queue Metrics into your Salesforce Org. This feature requires you to provide **AmazonConnectInstanceId**
- ix. **SalesforceAdapterNamespace:** This is the namespace for CTI Adapter managed package. The default value is **amazonconnect**. If a non-managed package is used, leave this field blank.
- x. **SalesforceCredentialsKMSKeyARN:** This is the ARN for KMS customer managed key that you created in the previous section.
- xi. **SalesforceCredentialsSecretsManagerARN:** This is the ARN for the Secrets Manager Secret that you created in the previous section.
- xii. **SalesforceHost:** The full domain for your salesforce org. For example
`https://mydevorg-dev-ed.my.salesforce.com`. Please make sure that the host starts with `https`, and that the url ends with `.my.salesforce.com`. This url can be found in `Setup -> My Domain`.
- xiii. **SalesforceProduction:** true | false - True for Production Environment, False for Sandbox
- xiv. **SalesforceUsername:** The username for the API user that you configured in the previous section. Salesforce usernames are in the form of an email address.
- xv. **SalesforceVersion:** This is the Salesforce.com API version that you noted in the previous section. The pattern of this value is `vXX.X`.
- xvi. **TranscribeOutputS3BucketName:** This is the S3 bucket where Amazon Transcribe stores the output. Typically, this is the same bucket that call recordings are stored in, so you can use the same value as found in **ConnectRecordingS3BucketName**. Not required if PostcallRecordingImportEnabled, PostcallTranscribeEnabled, ContactLensImportEnabled set to false.

- xvii. **VpcSecurityGroupList:** The list of SecurityGroupIds for Virtual Private Cloud (VPC). Not required if PrivateVpcEnabled is set to false.
- xviii. **VpcSubnetList:** The list of Subnets for the Virtual Private Cloud (VPC). Not required if PrivateVpcEnabled is set to false.
- xix. **AmazonConnectQueueMaxRecords:** Enter record set size for list queue query. Max is 100.
- xx. **AmazonConnectQueueMetricsMaxRecords:** Enter record set size for queue metrics query. Max is 100.
- xi. **CTREventSourceMappingMaximumRetryAttempts:** Maximum retry attempts on failure for lambdas triggered by Kinesis Events.
- xxii. **ConnectRecordingS3BucketName:** This is the name of the S3 bucket used to store recordings for your Amazon Connect instance. This is ONLY the bucket name, no sub-folders or suffixes
- xxiii. **ContactLensImportEnabled:** true | false - Set to false if importing Contact Lens into Salesforce should not be enabled.
- xxiv. **PostcallCTRImportEnabled:** true | false - Set to false if importing CTRs into Salesforce should not be enabled on the package level. This setting can be disabled on a call-by-call basis.
- xxv. **PostcallRecordingImportEnabled:** true | false - Set to false if importing call recordings into Salesforce should not be enabled on the package level. This setting can be disabled on a call-by-call basis.
- xxvi. **PostcallTranscribeEnabled:** true | false - Set to false if post-call transcription should not be enabled on the package level. This setting can be disabled on a call-by-call basis.
- xxvii. **TranscriptionJobCheckWaitTime:** Time between transcription job checks

11. Once you have completed the form, select **Deploy**

12. Deployment will take some time, with status updates being provided by the UI. Once it has completely deployed, you will receive a notification on the screen

Deployment status for serverlessrepo-SFConsolidatedLambdaPackage

[Create a new app](#)[Test app](#)

Your application has been deployed

Review the application's README for what to do next.

[Permissions](#)[Resources](#)[View CloudFormation Stack](#)

Test the Core Functionality

The package provides a core Lambda function (`sflInvokeAPI`) that supports multiple operations, like lookup, create and update. For the initial validation, sample events are provided within the function. Validating this function provides a good check that the installation and configuration is correct.

Validating the lambda functions requires the use of test events to simulate data coming into the function as it would in a typical deployment. Each function has a set of test event samples included to make validation easier.

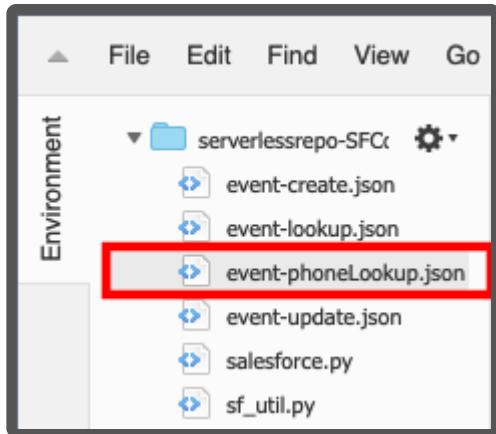
Validate the core functionality

1. In a new browser tab, login to the [AWS console](#)
2. Open the [AWS Lambda Console](#)
3. In the Filter field, enter `sflInvokeAPI` and press enter, this will filter your list out to the core function that we just installed

The screenshot shows the AWS Lambda Functions list page. At the top, there is a search bar with the placeholder "Add filter" and a keyword input field containing "Keyword : sflInvokeAPI". Below the search bar, there is a table header with columns: "Function name", "Description", "Runtime", and "Code size". A single function row is visible, showing the function name "serverlessrepo-SFConsolidatedLambdaPac-sflInvokeAPI-5504EV6KL9E8", runtime "Python 3.7", and code size "32.1 kB".

Function name	Description	Runtime	Code size
serverlessrepo-SFConsolidatedLambdaPac-sflInvokeAPI-5504EV6KL9E8		Python 3.7	32.1 kB

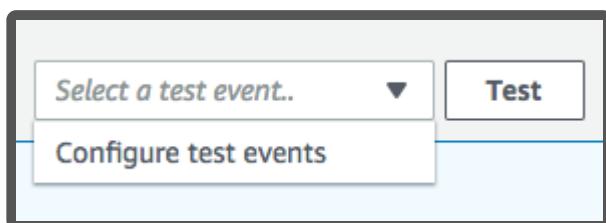
4. Select the **function name**. First, we will validate a phone number lookup.
5. In the Environment pane, double-click the event-phoneLookup.json file



6. The test even JSON will open in the Lambda editor
7. Modify the value for sf_phone to match the phone number of the test contact you created when you setup the CTI adapter or for any valid contact in your Salesforce org| NOTE: The phone number must be in [E.164 format](#)

```
1 {  
2     "Details": {  
3         "Parameters": {  
4             "sf_operation" : "phoneLookup",  
5             "sf_phone": "+14155551212",  
6             "sf_fields": "Id, Name, Email"  
7         }  
8     }  
9 }
```

8. Select the entire JSON event and copy it, then close the **event-phoneLookup.json** tab.
9. In the top-right corner, select drop-down arrow next to **Test** and choose **Configure test events**



10. Select the radio button for **Create new test event** and provide an event name, for example: **phoneLookup**
11. Select the existing event JSON and **delete** it. Paste the modified JSON payload you copied from the **event-phoneLookup.json** file

Configure test event



A function can have up to 10 test events. The events are persisted so you can switch to another computer or web browser and test your function with the same events.

- Create new test event
- Edit saved test events

Event template

Hello World



Event name

phoneLookup

```
1 [{  
2   "Details": {  
3     "Parameters": {  
4       "sf_operation": "phoneLookup",  
5       "sf_phone": "+14155551212",  
6       "sf_fields": "Id, Name, Email"  
7     }  
8   }  
9 }]
```

12. Select **Create** to save your test event

13. By default, your new test event should be selected in the drop-down list to the left of the Test button.

A screenshot showing a horizontal button group. From left to right: a dropdown menu containing the text 'phoneLookup' and a downward arrow; a blue rectangular button labeled 'Test'; and an orange rectangular button labeled 'Save'.

14. Select **Test**

15. If successful, the result will contain fields defined in "sf_fields" parameter in the invocation event

Execution result: succeeded (logs)

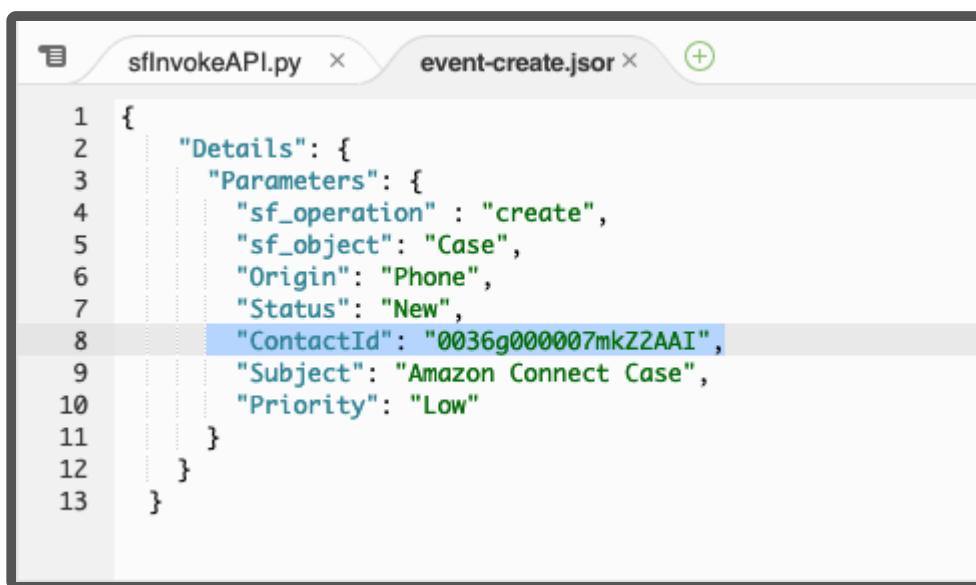
▼ Details

The area below shows the result returned by your function execution. [Learn more](#)

```
{  
  "Id": "0036g000007mkZ2AAI",  
  "Name": "John Smith",  
  "Email": null,  
  "sf_count": 1  
}
```

16. Copy the value for the **Id** key in the response. Next, we are going to use that Id to create a Case in Salesforce.

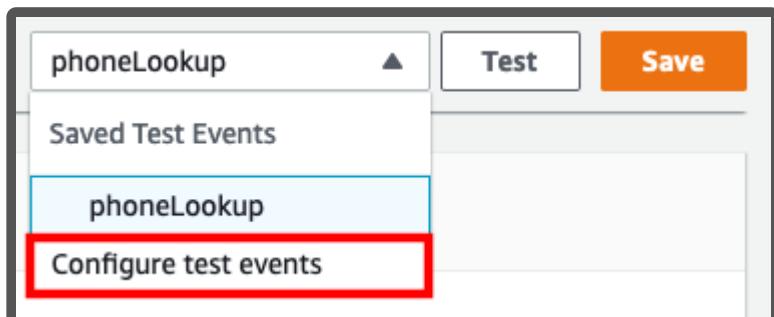
17. In the Environment pane, double-click the **event-create.json** file. Replace the existing ContactId value with the ID value you copied previously.



```
sfInvokeAPI.py x event-create.json x +  
1 {  
2   "Details": {  
3     "Parameters": {  
4       "sf_operation" : "create",  
5       "sf_object": "Case",  
6       "Origin": "Phone",  
7       "Status": "New",  
8       "ContactId": "0036g000007mkZ2AAI",  
9       "Subject": "Amazon Connect Case",  
10      "Priority": "Low"  
11    }  
12  }  
13 }
```

18. Select the entire JSON event and copy it, then close the **event-create.json** tab.

19. In the top-right corner, select drop-down arrow next to **Test** and choose **Configure test events**



20. Select the radio button for **Create new test event** and provide an event name, for example: **createCase**

21. Select the existing event JSON and **delete** it. Paste the modified JSON payload you copied from the **event-create.json** file

```
1 {  
2   "Details": {  
3     "Parameters": {  
4       "sf_operation": "create",  
5       "sf_object": "Case",  
6       "Origin": "Phone",  
7       "Status": "New",  
8       "ContactId": "0036g000007mkZ2AAI",  
9       "Subject": "Amazon Connect Case",  
10      "Priority": "Low"  
11    }  
12  }  
13 }
```

22. Select **Create** to save your test event

23. By default, your new test event should be selected in the drop-down list to the left of the Test button.



24. Select **Test**

25. If successful, the result will contain the Case Id

Execution result: succeeded (logs)

▼ Details

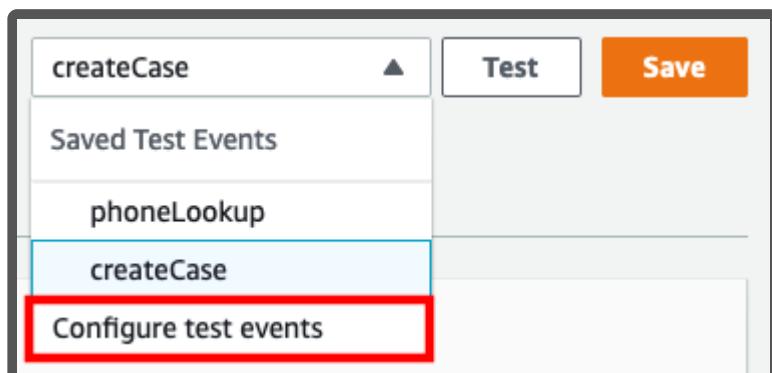
The area below shows the result returned by your function execution. [Learn](#)

```
{  
  "Id": "5006g000008AfEBAA0"  
}
```

26. Copy the value for the **Id** key in the response.
27. When we created the case, the **Status was set to New** and the **Priority to Low**. We are going to use the update operation to close the case.
28. In the Environment pane, double-click the **event-update.json** file and replace the existing Case Id in "sf_id" parameter with the new one you copied from the last test result

```
1 {  
2   "Details": {  
3     "Parameters": {  
4       "sf_operation" : "update",  
5       "sf_object": "Case",  
6       "sf_id": "5006g000008AfEBAA0",  
7       "Status": "Closed"  
8     }  
9   }  
10 }
```

29. Select the **entire JSON event** and copy it, then close the **event-update.json** tab.
30. In the top-right corner, select drop-down arrow next to **Test** and choose **Configure test events**



31. Select the radio button for **Create new test event** and provide an event name, for example: **updateCase**

32. Select the existing event JSON and **delete** it. Paste the modified JSON payload you copied from the **event-update.json** file

Configure test event

A function can have up to 10 test events. The events are persisted so you can switch to another computer or web browser and test your function with the same events.

Create new test event
 Edit saved test events

Event template

createCase

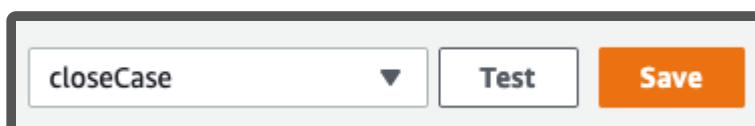
Event name

closeCase

```
1 - {  
2 -   "Details": {  
3 -     "Parameters": {  
4 -       "sf_operation": "update",  
5 -       "sf_object": "Case",  
6 -       "sf_id": "5006g000008AFEBAA0",  
7 -       "Status": "Closed"  
8 -     }  
9 -   }  
10 }
```

33. Select **Create** to save your test event

34. By default, your new test event should be selected in the drop-down list to the left of the Test button.



The screenshot shows a user interface for testing a function. It features a dropdown menu labeled 'closeCase' with a downward arrow, a 'Test' button in a light blue box, and a 'Save' button in an orange box. The 'Save' button is currently highlighted, indicating it is the active or next step.

35. Select **Test**

36. If successful, the result will be the **HTTP 204 No Content** success status response code

Execution result: succeeded (logs)

▼ Details

The area below shows the result returned by your function:

```
{  
  "Status": 204  
}
```

37. Log in into your Salesforce org and go to the **Service Console**

38. In the search box, change the object type to Cases and type Amazon Connect Case, then press enter



39. You should find 1 case opened by the API user, and the status should be closed

Cases					
1 Result					
Case Number	Subject	Status	Date/Time Opened	Case Owner Alias	
00001026	Amazon Connect Case	Closed	1/23/2020, 10:13 PM	apiuser	

40. You have completed core function validation

Allow Amazon Connect to Access the sflnvokeAPI Lambda Function

Once you have validated function, you can use the Amazon Connect console to add the sflnvokeAPI Lambda function to your Amazon Connect instance. This automatically adds resource permissions that allow Amazon Connect to invoke the function.

Add the Lambda function to your Amazon Connect instance

1. In a new browser tab, login to the [AWS console](#)
2. Navigate to the [Amazon Connect Console](#)
3. Select your **Instance Alias**
4. In the navigation pane, choose **Contact flows**.

Amazon Connect

X

Instances

Overview

Telephony

Data storage

Data streaming

Contact flows

Analytics tools

Approved origins

Customer profiles

5. For **AWS Lambda**, select the function that includes sfInvokeAPI in the name

AWS Lambda

By using AWS Lambda function, you can retrieve data from database and other services and be routed to the appropriate contact flow branch. By adding Lambda functions, you can invoke them [Create a new Lambda function](#)

Lambda Functions

serverlessrepo-AmazonConnectSalesforce-sfInvokeAPI-Z... ▾

+ Add Lambda Function

Lambda Functions

6. Choose **Add Lambda Function**. Confirm that the ARN of the function is added under **Lambda Functions**.

Lambda Functions

Function Arn	Arn	Copy to clipboard	Action
serverlessrepo-AmazonConnectSalesforce-sfInvokeAPI-Z...	arn:aws:lambda:us-west-2:...	Copy	Remove

7. The AWS Lambda function has been added to your Amazon Connect instance.

Upgrading from an Earlier Version

Upgrading the Salesforce Lambdas

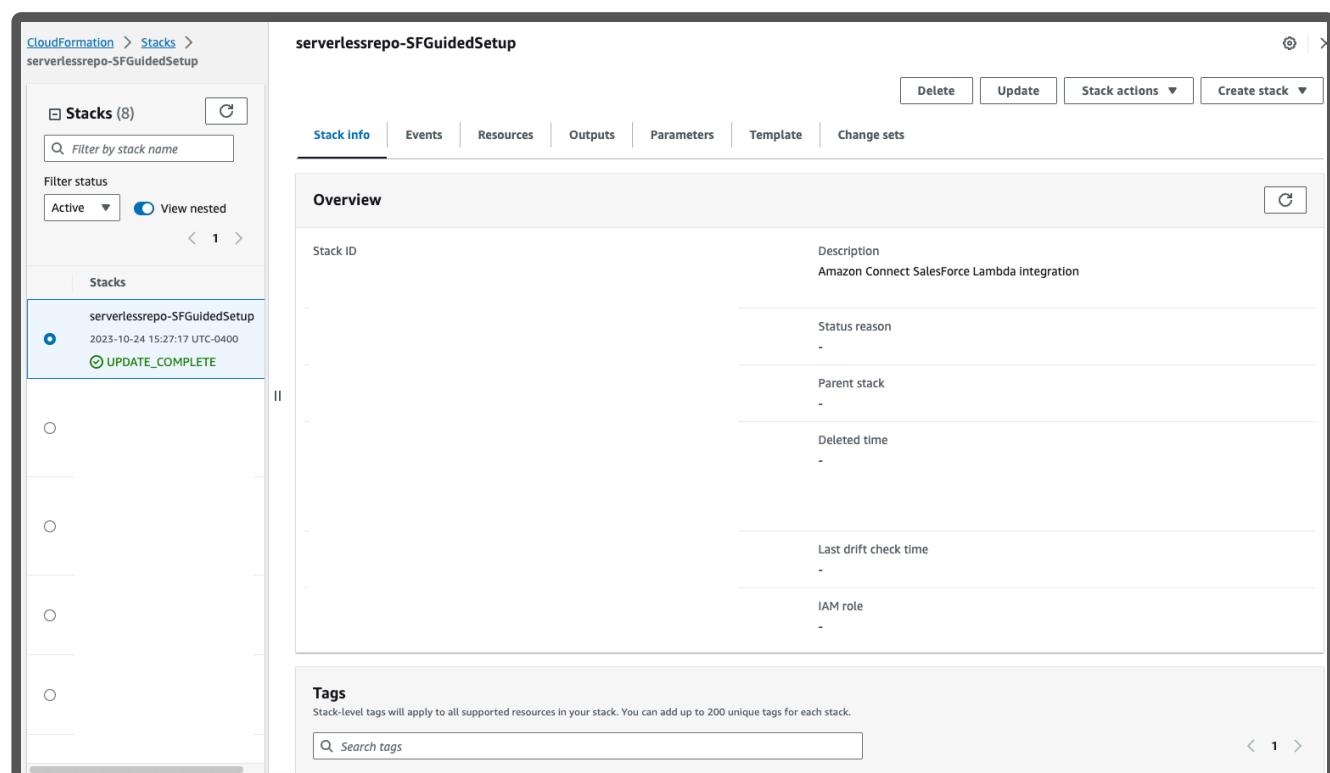
The Amazon Connect Salesforce Lambda (ACSF Lambda) package contains a set of common Lambda functions to be used by Amazon Connect to interact with Salesforce, allowing lookup, create and update operations for different Salesforce objects, like Contacts and Cases. Upgrading the Salesforce lambdas is necessary if you have an existing lambda package and you wish to upgrade to a newer runtime version.

To upgrade your lambdas to use the new python version, continue reading.

Deploying the New Lambda Package

Gather the data in your lambda CloudFormation stack.

1. In a new browser tab, login to the [AWS console](#).
2. Navigate to the CloudFormation console.
3. Ensure you are in the correct region.
4. If you are not already on the Stacks page, click Stacks listed in the left-hand navigation pane.
 - It should look similar to this



5. Find and click the previously-deployed stack in the list.
6. In the split panel that appears for the stack, click on the Parameters tab.
7. The list of parameters used for the previously-deployed stack will appear.

The screenshot shows the AWS CloudFormation console with the 'Parameters' tab selected for the 'serverlessrepo-SFGuidedSetup' stack. The table displays the following parameters:

Key	Value	Resolved value
AmazonConnectInstanceId		-
AmazonConnectQueueMaxRecords		-
AmazonConnectQueueMetricsMaxRecords		-
ConnectRecordingS3BucketName		-
ConnectReportingS3BucketName		-
ContactLensImportEnabled		-
CTREventSourceMappingBatchSize		-
CTREventSourceMappingMaximumRetryAttempts		-
CTRKineticsARN		-
HistoricalReportingImportEnabled		-
LambdaLoggingLevel		-
PostcallCTRImportEnabled		-
PostcallRecordingImportEnabled		-
PostcallSFImportEnabled		-

Verify the information you used to create the lambdas. For a refresher, this would be the information you found as part of [Prerequisite Configuration and Data Collection](#), [Finding your salesforce version](#), and [Gather Amazon Connect Information](#).

Once you have verified and gathered all of the information, you will install the salesforce lambda package. Instructions on how to install the lambda package [can be found here](#), and instructions on testing the lambda package after installation [can be found here](#)

Removing the Previous Lambda Package

At this point, you now have the newer version of the lambda package and the older version of the lambda package. We now want to remove the original lambda package. but this needs to be done with care since this lambda package was configured to work with the connect and salesforce instances. Before removing the old lambda package, we must first do two (2) things

Unlink the sfInvokeAPI Function

1. In a new browser tab, login to the [AWS console](#).
2. Navigate to the Amazon Connect Console.
3. Select your Instance Alias.
4. In the navigation pane, choose "Flows".

The screenshot shows the 'Flows' section of the Amazon Connect console. On the left, there's a sidebar with various navigation options like Instances, Overview, Telephony, etc. The main area is titled 'AWS Lambda'. It contains a section for 'Lambda Functions' with a button '+ Add Lambda Function'. Below it, a table titled 'Lambda Functions 1' lists one item: 'serverlessrepo-SFGuidedSetup-sfInvokeAPI-'. The 'Action' column for this entry has a 'Remove' link.

Once in the "Flows" pane of the Connect console of the instance to be configured, verify that the previous Lambda function is listed, then remove it by clicking "Remove."

This screenshot is identical to the one above, but the 'Remove' link in the 'Action' column of the Lambda function table is highlighted with a red rectangle.

After the old Lambda function has been removed, proceed with the remaining steps:

1. For AWS Lambda, select the function that includes sfInvokeAPI from the new lambda package in the name.
2. Choose Add Lambda Function. Confirm that the function is added under Lambda Functions.
3. The AWS Lambda function has been added to your Amazon Connect instance!

Remove triggers from the Lambda functions of the previously-deployed stack

In the documentation steps "[Amazon Connect Historical Metrics in Salesforce](#)" and "[Postcall Contact Lens Import](#)", you are instructed to create AWS Lambda Triggers for certain Lambda functions in order to store relevant data in S3 bucket folders. If the Lambdas of the previously-deployed stack have triggers which point to the same file path (i.e., S3 bucket, Prefix, and Suffix)

that is listed in the documentation instructions, this will prevent the creation of triggers for new Lambda functions pointing to that same path.

To delete the old triggers,

1. In a new browser tab, login to the AWS console.
2. Navigate to the Lambda console.
3. Navigate to a specific function which holds an old trigger.

The screenshot shows the 'Function overview' page for a Lambda function named 'sfInvokeAPI'. On the left, there's a summary card with the function name, a code icon, and a 'Layers' section showing '(1)'. Below it is a list of triggers, currently showing one 'S3' trigger. A button '+ Add trigger' is visible. On the right, there are sections for 'Description', 'Last modified' (6 days ago), 'Function ARN', 'Application', and 'Function URL' (Info). At the bottom, a navigation bar includes tabs for 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions', with 'Configuration' being the active tab.

4. For each function, go to its Configuration tab (under the Function overview) and ensure you are in the Triggers tab (in the left-hand side, listed under General Configuration). Clicking on the rectangle with the S3 bucket icon (representing the trigger) in the Function overview should also bring you to the correct pane.

The screenshot shows the 'Configuration' tab for the 'sfInvokeAPI' function. On the left, a sidebar lists various configuration options, with 'Triggers' being the selected item. The main pane displays the 'Triggers (1) info' section. It shows a single trigger named 'Trigger' associated with an 'S3' bucket. A 'Details' arrow indicates that clicking it will expand the trigger's configuration. At the top right of the main pane, there are buttons for 'Fix errors', 'Edit', 'Delete', and 'Add trigger'. Navigation arrows at the bottom right show the user is on page 1 of 1.

5. Once there, click the Details arrow to expand the details section and verify that the trigger you see has the same file path you wish to reclaim for your new Lambda function. If so, click its

checkbox and then click delete in the upper right-hand corner of the Triggers pane.

The image contains two screenshots of the AWS Lambda Triggers configuration page. Both screenshots show a sidebar on the left with options like General configuration, Triggers (selected), Permissions, Destinations, Function URL, Environment variables, Tags, VPC, Monitoring and operations tools, Concurrency, Asynchronous invocation, Code signing, Database proxies, File systems, and State machines. The main pane displays a list of triggers under 'Triggers (1) Info'. In the first screenshot, there is one trigger named 'S3' with details: Bucket arn: arn:aws:s3:::arn:a, Event types: s3:ObjectCreated:Put, Notification name: , Service principal: , Source account: !, Statement ID: l:ar. In the second screenshot, the same trigger is shown, but the 'Delete' button in the top right corner of the trigger card is highlighted with a red box.

6. Repeat this process (step 3 - step 5) for each old Lambda function until all the file paths you wish to use are freed up. Those file paths will now be available for use in the Amazon Connect Salesforce Lambda setup steps which involve creating triggers.

Delete the Previously-Deployed Stack

Once you have confirmed your successful upgrade of the Amazon Connect Salesforce Lambda, you may delete your previously-deployed stack of the older version of the Lambda package. To delete the previously-deployed stack,

1. In a new browser tab, login to the AWS console.
2. Navigate to the CloudFormation console.
3. Ensure you are in the correct region.
4. If you are not already on the Stacks page, click Stacks listed in the left-hand navigation pane.

The screenshot shows the AWS CloudFormation console. On the left, there's a sidebar with a search bar and a 'Stacks (8)' section. The main area is titled 'serverlessrepo-SFGuidedSetup' and contains tabs for 'Stack info', 'Events', 'Resources', 'Outputs', 'Parameters', 'Template', and 'Change sets'. The 'Stack info' tab is selected. It displays the 'Stack ID' as 'serverlessrepo-SFGuidedSetup', the 'Description' as 'Amazon Connect SalesForce Lambda integration', and other details like 'Status reason', 'Parent stack', 'Deleted time', 'Last drift check time', and 'IAM role'. Below this is a 'Tags' section with a search bar.

5. Find and click the previously-deployed stack in the list.
6. In the split panel that appears for the stack, validate in the Stack info tab that the selected stack is the one you wish to delete.

This screenshot shows the 'Parameters' tab for the same stack. The 'Delete' button in the top navigation bar is highlighted with a red box. The main area lists 27 parameters with their keys and resolved values. The parameters include various AWS service names like AmazonConnect, ConnectRecordingS3BucketName, and LambdaLoggingLevel.

Key	Value	Resolved value
AmazonConnectInstanceId		-
AmazonConnectQueueMaxRecords		-
AmazonConnectQueueMetricsMaxRecords		-
ConnectRecordingS3BucketName		-
ConnectReportingS3BucketName		-
ContactLensImportEnabled		-
CTREventSourceMappingBatchSize		-
CTREventSourceMappingMaximumRetryAttempts		-
CTRKinesisARN		-
HistoricalReportingImportEnabled		-
LambdaLoggingLevel		-
PostcallCTRImportEnabled		-
PostcallRecordingImportEnabled		-
PostcallTranscribeEnabled		-

7. Click the Delete button in the split panel, then Delete in the pop-up confirmation message that appears.

Delete stack?

X

Delete stack
undo.

permanently? This action cannot be

 Deleting this stack will delete all stack resources. Resources will be deleted according to their DeletionPolicy. [Learn more](#) 

Cancel

Delete

- o 8. The stack will then begin to be deleted. This may take a few minutes; it is not necessary to stay on the page.
- o 9. Confirm that the stack was successfully completed. If you have navigated away from the page, change the Filter status at the top of the Stacks list to Deleted, and verify that the previously-deployed Lambda stack appears marked DELETE_COMPLETE.

 2023-08-24 13:13:45 UTC-0400
 **DELETE_COMPLETE**

Upgrading the CTI Adapter

If you are upgrading from an earlier version of CTI Adapter, there are a few additional things you need to do.

AC CTI Adapter

1. Go to the **Setup** section and search for **Object Manager**.
2. In Object Manager section, search for "AC CTI"

SETUP	Object Manager	ac ct		
3 Items, Sorted by Label				
LABEL	API NAME	DESCRIPTION	LAST MODIFIED	DEP
AC CTI Adapter	amazonconnect__AC_CtiAdapter__c		8/6/2020	✓
AC CTI Attribute	amazonconnect__AC_CtiAttribute__c		8/6/2020	✓
AC CTI Flow	amazonconnect__AC_CtiScript__c		8/6/2020	✓

3. Open up AC CTI Adapter

4. On the left sidebar, click on Page Layouts

5. Click on Page Layout Assignment

6. On the next page, click on Edit Assignments

7. Click on the grey bar at the top of the table to select all rows.

Setup > Object Manager
AC CTI Adapter

Edit Page Layout Assignment
AC CTI Adapter

The table below shows the page layout assignments for different profiles. Use SHIFT + click or click and drag to select a range of adjacent cells. Use CTRL + click to select multiple cells that are not adjacent. Then choose a new page layout from the drop-down.

Profiles	Page Layout
Analytics Cloud Integration User	AC CTI Adapter Layout
Analytics Cloud Security User	AC CTI Adapter Layout
Chatter External User	AC CTI Adapter Layout
Chatter Free User	AC CTI Adapter Layout
Chatter Moderator User	AC CTI Adapter Layout
Contract Manager	AC CTI Adapter Layout
Cross Org Data Proxy User	AC CTI Adapter Layout
Custom: Marketing Profile	AC CTI Adapter Layout
Custom: Sales Profile	AC CTI Adapter Layout
Custom: Support Profile	AC CTI Adapter Layout
Force.com - App Subscription User	AC CTI Adapter Layout
Force.com - Free User	AC CTI Adapter Layout
Gold Partner User	AC CTI Adapter Layout
Identity User	AC CTI Adapter Layout
Marketing User	AC CTI Adapter Layout
Minimum Access - Salesforce	AC CTI Adapter Layout
Partner App Subscription User	AC CTI Adapter Layout

Setup > Object Manager
AC CTI Adapter

Edit Page Layout Assignment
AC CTI Adapter

The table below shows the page layout assignments for different profiles. Use SHIFT + click or click and drag to select a range of adjacent cells. Use CTRL + click to select multiple cells that are not adjacent. Then choose a new page layout from the drop-down.

Profiles	Page Layout
Analytics Cloud Integration User	AC CTI Adapter Layout
Analytics Cloud Security User	AC CTI Adapter Layout
Chatter External User	AC CTI Adapter Layout
Chatter Free User	AC CTI Adapter Layout
Chatter Moderator User	AC CTI Adapter Layout
Contract Manager	AC CTI Adapter Layout
Cross Org Data Proxy User	AC CTI Adapter Layout
Custom: Marketing Profile	AC CTI Adapter Layout
Custom: Sales Profile	AC CTI Adapter Layout
Custom: Support Profile	AC CTI Adapter Layout
Force.com - App Subscription User	AC CTI Adapter Layout
Force.com - Free User	AC CTI Adapter Layout

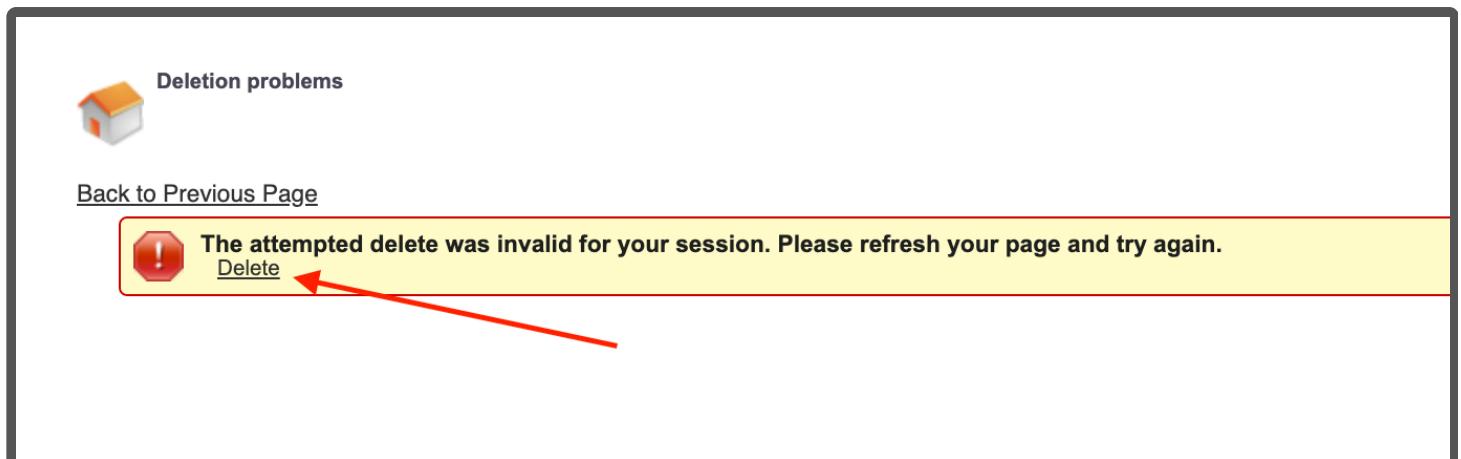
8. Open the **Page Layout to Use** dropdown and select **AC CTI Adapter Layout -- August 2020**.

9. Click **Save** and go back to **Page Layouts**.

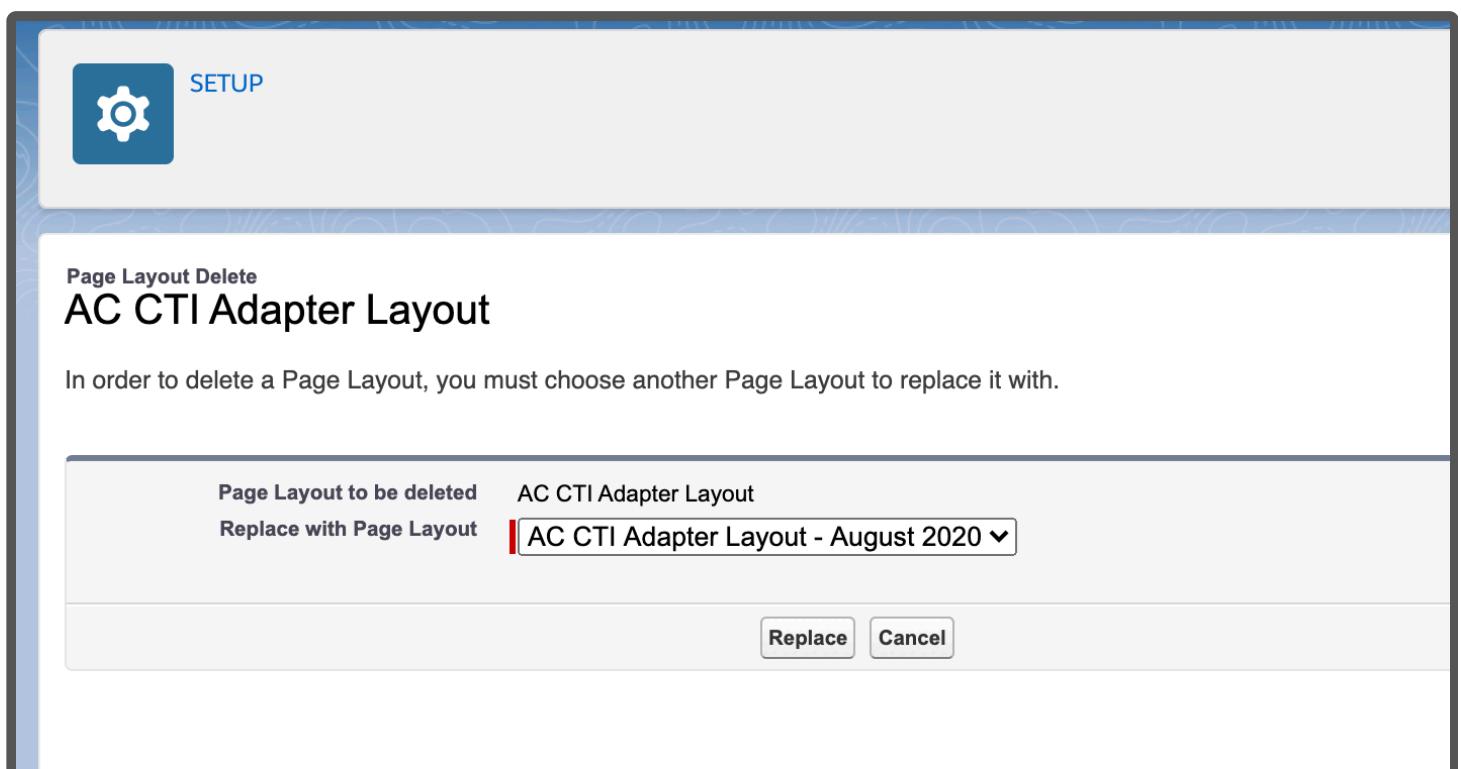
10. Click on the dropdown next to the item labelled **AC CTI Adapter Layout** and click **Delete**.

11. Confirm **Yes** in the next dialogue where you will be asked "Are you sure?"

12. If you see a screen titled **Deletion Problems**, find and click **Delete**.



13. You will be asked which layout you want to replace it with. Select **AC CTI Adapter Layout -- August 2020** and click **Replace**.



AC CTI Script

Now we are going to do the same thing for **AC CTI Script Layout**.

1. Open up **AC CTI Script Layout**

2. On the left sidebar, click on **Page Layouts**

3. Click on **Page Layout Assignment**

4. On the next page, click on **Edit Assignments**

5. Click on the grey bar at the top of the table to select all rows.

SETUP > OBJECT MANAGER
AC CTI Flow

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Deleted Objects

Edit Page Layout Assignment
AC CTI Flow

The table below shows the page layout assignments for different profiles. Use SHIFT + click or click and drag to select a range of adjacent cells. Use CTRL + click to select multiple cells that are not adjacent. Then choose a new page layout from the drop-down.

Save Cancel

Page Layout To Use: -- Select Page Layout -- 0 Selected 0 Changed

Profiles	Page Layout
Analytics Cloud Integration User	AC CTI Script Layout
Analytics Cloud Security User	AC CTI Script Layout
Chatter External User	AC CTI Script Layout
Chatter Free User	AC CTI Script Layout
Chatter Moderator User	AC CTI Script Layout
Contract Manager	AC CTI Script Layout
Cross Org Data Proxy User	AC CTI Script Layout
Custom: Marketing Profile	AC CTI Script Layout
Custom: Sales Profile	AC CTI Script Layout

SETUP > OBJECT MANAGER
AC CTI Flow

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Deleted Objects

Edit Page Layout Assignment
AC CTI Flow

The table below shows the page layout assignments for different profiles. Use SHIFT + click or click and drag to select a range of adjacent cells. Use CTRL + click to select multiple cells that are not adjacent. Then choose a new page layout from the drop-down.

Save Cancel

Page Layout To Use: -- Select Page Layout -- 26 Selected 0 Changed

Profiles	Page Layout
Analytics Cloud Integration User	AC CTI Script Layout
Analytics Cloud Security User	AC CTI Script Layout
Chatter External User	AC CTI Script Layout
Chatter Free User	AC CTI Script Layout
Chatter Moderator User	AC CTI Script Layout
Contract Manager	AC CTI Script Layout
Cross Org Data Proxy User	AC CTI Script Layout
Custom: Marketing Profile	AC CTI Script Layout
Custom: Sales Profile	AC CTI Script Layout

6. Open the **Page Layout to Use** dropdown and select **AC CTI Flow Layout**.

7. Click **Save** and go back to **Page Layouts**.

8. Click on the dropdown next to the item labelled **AC CTI Script Layout** and click **Delete**.

9. Confirm **Yes** in the next dialogue where you will be asked "Are you sure?"

10. If you see a screen titled **Deletion Problems**, find and click **Delete**.



Deletion problems

[Back to Previous Page](#)



The attempted delete was invalid for your session. Please refresh your page and try again.
[Delete](#)

11. You will be asked which layout you want to replace it with. Select **AC CTI Flow Layout** and click **Replace**.

SETUP

Page Layout Delete
AC CTI Script Layout

In order to delete a Page Layout, you must choose another Page Layout to replace it with.

Page Layout to be deleted	AC CTI Script Layout
Replace with Page Layout	AC CTI Flow Layout ▾

Replace Cancel

12. Go to your **CTI Adapter**.

13. Click on any of the CTI Flows and scroll down to the section labeled **CTI Flow**. You should see something like this:

Invalid Script

Please note that starting from version 4.6, your scripts will need to be migrated to our new CTI Flows.

You can download your current script below



When you are ready to try out the CTI Flow editor, click Continue.

[Continue →](#)

14. Click **Download** and save your script before clicking **Continue**.

15. Use the CTI Block primitives in the editor to re-create your script as a CTI Flow.

16. Refer to the Sample Flows in the Appendix of this manual.

CTI Adapter Installation Troubleshooting and Common Issues

I upgraded my adapter to v5.10, but I cannot see the CCP Config changes

There is a bug with Salesforce that doesn't update a page layout when you upgrade a package. To fix this, go to Setup and search for **Object Manager**. Once you're on the Object Manager page, search for the **AC CTI Adapter** object and click on it. Then go into **Page Layouts** and click on the layout you are using (Typically **AC CTI Adapter Layout – August 2020**). Then, drag and drop the **Audio Device Settings** and **Page Layout Settings** into the desired spot on the page. Finally, hit save.

AC CTI Adapter

Fields

Field Name	Connect Instance ...	Custom Ringtone	Medialess	Softphone Popout ...
Section	Audio Device Sett...	Bottom Panel	Created By	Debug Level
Blank Space		Call Center Defin...	CTI Adapter Name	Enable Softphon ...
Amazon Connect In...		CCP Version	Custom Chat Ringt...	Phone Type Settings
Amazon Connect In...			Last Modified By	SSO Relay State
			Presence Sync Ena...	User Defined

Salesforce Mobile and Lightning Experience Actions

Actions in this section are predefined by Salesforce. You can [override the predefined actions](#) to set a customized list of actions on Lightning Experience and mobile app pages that use this layout. If you customize the actions and have saved the layout, then this section inherits that set of actions by default when you click to override.

AC CTI Adapter Detail

Standard Buttons		Custom Buttons																									
Edit	Delete	Clone	Change Owner																								
Change Record Type	Printable View	Sharing	Get Alerts																								
Information (Header visible on edit only) <table border="1"> <tr> <td>CTI Adapter Name</td> <td>Sample Text</td> <td>Owner</td> <td>Sample Text</td> </tr> <tr> <td>Amazon Connect Instance</td> <td>Sample Text</td> <td>Amazon Connect Instance Region</td> <td>Sample Text</td> </tr> <tr> <td>Custom Ringtone</td> <td>www.salesforce.com</td> <td>Call Center Definition Name</td> <td>Sample Text</td> </tr> <tr> <td>Softphone Popout Enabled</td> <td><input checked="" type="checkbox"/></td> <td>Debug Level</td> <td>Sample Text</td> </tr> <tr> <td>Medialess</td> <td><input checked="" type="checkbox"/></td> <td>Presence Sync Enabled</td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>Audio Device Settings</td> <td><input checked="" type="checkbox"/></td> <td>Phone Type Settings</td> <td><input checked="" type="checkbox"/></td> </tr> </table>				CTI Adapter Name	Sample Text	Owner	Sample Text	Amazon Connect Instance	Sample Text	Amazon Connect Instance Region	Sample Text	Custom Ringtone	www.salesforce.com	Call Center Definition Name	Sample Text	Softphone Popout Enabled	<input checked="" type="checkbox"/>	Debug Level	Sample Text	Medialess	<input checked="" type="checkbox"/>	Presence Sync Enabled	<input checked="" type="checkbox"/>	Audio Device Settings	<input checked="" type="checkbox"/>	Phone Type Settings	<input checked="" type="checkbox"/>
CTI Adapter Name	Sample Text	Owner	Sample Text																								
Amazon Connect Instance	Sample Text	Amazon Connect Instance Region	Sample Text																								
Custom Ringtone	www.salesforce.com	Call Center Definition Name	Sample Text																								
Softphone Popout Enabled	<input checked="" type="checkbox"/>	Debug Level	Sample Text																								
Medialess	<input checked="" type="checkbox"/>	Presence Sync Enabled	<input checked="" type="checkbox"/>																								
Audio Device Settings	<input checked="" type="checkbox"/>	Phone Type Settings	<input checked="" type="checkbox"/>																								

Error “refused to run the JavaScript URL because it violates the following Content Security Policy directive...”

This is an allowlisting issue, please review the installation and ensure that both URLs are properly allowlisted.

Error “refused to frame” Visualforce page

s.com/feature/5633521622188032.

▶ Object BeaconLibrary.js:38

✖ Refused to frame '[https://\[REDACTED\].amazonconnect.\[REDACTED\].visual.force.com/](https://[REDACTED].amazonconnect.[REDACTED].visual.force.com/)' because an ancestor violates the following Content Security Policy directive: "frame-ancestors 'self'" [REDACTED]

▶ Object BeaconLibrary.js:38

⚠ DevTools failed to load SourceMap: Could not load content for <https://c.lal-c1.cs-ord.salesforcedeliveagent.com/content/dev/resources/js/scrt.min.js.map>: HTTP

This can happen if the customer has checked “Enable clickjack protection” on Salesforce session settings. The solution is to uncheck that.

The screenshot shows the 'Session Settings' page under the 'Setup' menu. The 'Session Management' tab is selected. In the 'Clickjack Protection' section, there are four checkboxes:

- Enable clickjack protection for Setup pages
- Enable clickjack protection for non-Setup Salesforce pages (highlighted with a red box and a red arrow)
- Enable clickjack protection for customer Visualforce pages with standard headers
- Enable clickjack protection for customer Visualforce pages with headers disabled

A tooltip 'Protect against clickjack attacks and allow framing on whitelisted external domains' is displayed next to the second checkbox.

Browser refreshing when trying to open lightning components

This issue was first seen when trying to use the `screenPop()` method provided by Salesforce (this is the method we use for our Screenpop CTI Flow blocks).

How do you fix it?

Remove the "`&0.source=aloha`" value from the browser. This can often be added to the current URL when a Salesforce page is navigated through the use of a bookmark.

Why does this happen?

The cause of this issue is correlated with the presence of "`&0.source=aloha`" in the home page URL after logging into Salesforce. For context on what this value means, you can consult [this reference](#). This code is set by Salesforce to forcefully navigate a user to a url, which is needed in cases when a user tries to navigate to a salesforce page that can only be viewed by a logged in user. After logging in, this code is still present, and this is why this issue occurs. When this value is present, opening new tabs would result in the entire browser refreshing because it's still forcefully trying to navigate the user. This browser refresh eventually converts incoming calls to missed calls, which is the expected behavior while using the CTI Adapter.

What are the Disable X Trigger options in the Custom Settings?

Edit Toolkit for Amazon Connect

[Save](#) [Cancel](#)

Toolkit for Amazon Connect Information

Location

Disable the CCA Case Trigger

Disable the CCA Contact Trigger

Disable the Case Contact CCA Trigger

Disable the Task Trigger

Url 

These are options we provide that allow you to toggle certain functionality in the adapter.

- CCA Case Trigger - This trigger looks for any ContactChannelAnalytics records that could be related to a updated/inserted Case, and creates a relationship between the two records. This trigger uses batching to process the update requests.
- CCA Contact Trigger - This trigger looks for any ContactChannelAnalytics records that could be related to a updated/inserted Contact, and creates a relationship between the two records. This trigger uses batching to process the update requests.
- Case Contact CCA Trigger - This trigger looks for any Case/Contact records that could be related to an updated/inserted ContactChannelAnalytics record, and creates a relationship between the records.
- Task Trigger - This trigger creates a ContactChannel record for any inserted/updated task that with a `CallObject` field that does not currently have a ContactChannel record created before.
- Voice Id Channel Trigger - This trigger associates Cases and Contacts with a Voice Id Channel Record

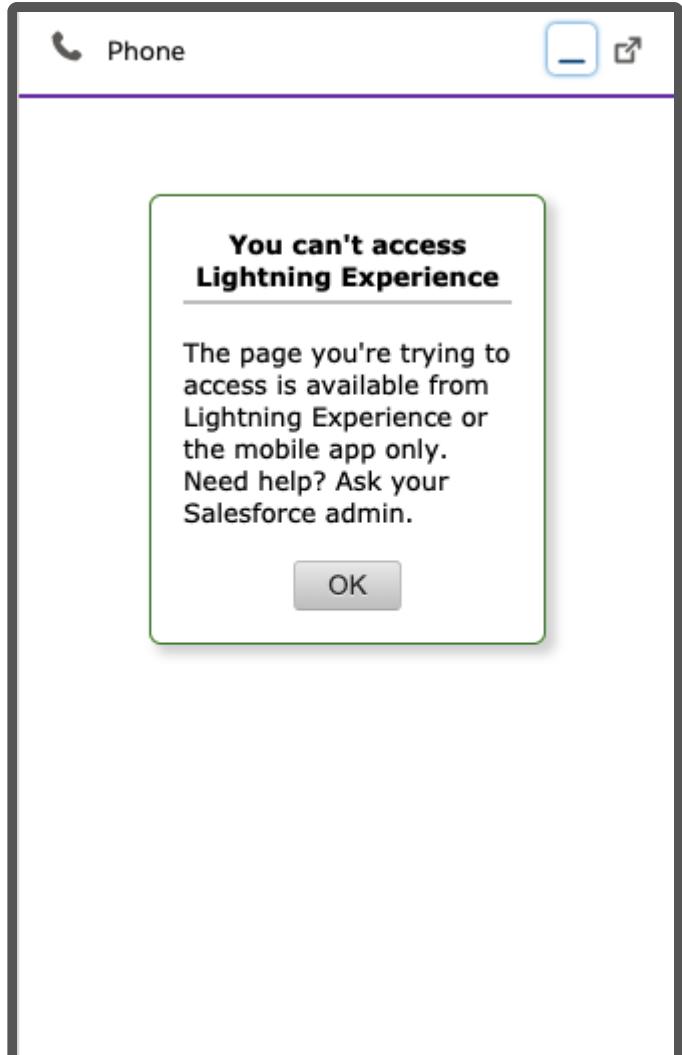
I upgraded my adapter to v5, but I don't see the CTI Flows feature.

See the [Upgrading from an Earlier Version](#) section of the installation guide.

I upgraded my adapter from v3 to v5 and we lost some screenpop functionality.

All screenpop functionality native to v3 now needs to be recreated using CTI Flows. Please review the [CTI Flow Examples](#) for more details, all screenpop functionality from v3 has been recreated.

The CCP doesn't show up in service console and I instead see the following image:



Copy the full url of the lightning adapter visualforce page into the call center.

Certain picklists are missing picklist items.

When upgrading from a version of the package to a higher version of the package in which new picklist items were added to a picklist, those new picklist items won't be installed. This is a [known Salesforce issue](#).

How to remove permissions to Visualforce pages, Apex classes for a desired profile

1. Navigate to **Setup** and search for "Profiles".
2. Select the desired profile.
3. Select either **Visualforce Page Access** or **Apex Class Access**.

The screenshot shows the Salesforce Setup interface under the 'Profiles' section. A search bar at the top contains the text 'profiles'. On the left, a sidebar shows 'Users' and 'Profiles' with 'Profiles' highlighted. A message below says 'Didn't find what you're looking for? Try using Global Search.' The main content area has a blue header 'SETUP Profiles'. Underneath, there are two sections: 'Apex Class Access' (Permissions to execute Apex classes) and 'Visualforce Page Access' (Permissions to execute Visualforce pages).

4. Select **Edit** and remove any desired permissions. All permissions can be removed because permissions are managed through permission sets, not through profiles.

CTI Adapter Configuration

The CTI Adapter configuration begins with the adapter details. These fields provide the basic information needed to relate the Adapter to the call center configuration in Salesforce and, ultimately, to the agents and supervisors that will be using the platform.

The screenshot shows the 'Details' tab of the CTI Adapter configuration page. It lists various settings:

- CTI Adapter Name: ACLightningAdapter
- Amazon Connect Instance: https://sfadAPTERtest.awsapps.com/
- Custom Ringtone: (checkbox)
- Softphone Popout Enabled: (checkbox checked)
- Medialess: (checkbox)
- Audio Device Settings: (checkbox)
- Owner: [REDACTED]
- Amazon Connect Instance Region: us-east-1
- Call Center Definition Name: ACLightningAdapter
- Debug Level: Off
- Presence Sync Enabled: (checkbox checked)
- Phone Type Settings: (checkbox checked)

A section for 'Single SignOn (SSO)' is also present but collapsed.

CTI Adapter Details

- 1. CTI Adapter Name:** provide a unique name for this CTI adapter definition
- 2. Amazon Connect Instance:** This was configured in a previous section. This is the instance url for your Amazon Connect instance.
- 3. Amazon Connect Instance Region:** This is the code for the region that you have deployed your Amazon Connect instance to. This is required for the Amazon Connect chat APIs to work correctly. If you do not use the chat feature of Amazon Connect, this field is not necessary
- 4. Custom Ringtone:** This allows for overriding the built-in ringtone with any browser-supported audio file accessible by the user.
- 5. Call Center Definition Name:** This was configured in a previous section. This is the internal name of the Call Center configured in Salesforce setup. This value links the CTI Adapter to the Call Center, and ultimately to the agents.

6. Softphone Popout Enabled: Salesforce supports softphone pop out in Console and Lightning Experience modes. When the softphone is popped out, it opens in a new browser window external to the Salesforce UI. This is helpful in use cases where the call controls are regularly needed but the agent also needs full access to the entire console.

7. Debug Level: For future use

8. Medialess: Amazon Connect supports running in VDI environments, however best practice is to send the actual audio stream via a separate CCP. Selecting the medialess option will configure the Salesforce CCP to run in medialess mode, which provides the data that Salesforce needs for screenpop while the audio is streamed to a local CCP. [See more information here](#)

9. Presence Sync Enabled: This setting allows the adapter to use the presence rules to sync state from Amazon Connect to Salesforce Omni-Channel.

10. Early Get User Media (GUM): When enabled, the CCP will capture the agent's browser microphone media stream before the contact arrives to reduce the call setup latency. If disabled, CCP will only capture agent media stream after the contact arrives.

Note: Enabling this feature may lead to draining in wireless headset batteries and/or impacted music/video audio quality when the agent is not on the call. [Link to streams documentation](#)

11. Audio Device Settings Turning this setting on allows the Agent to setup a custom audio device for their speaker, microphone and ringer in the adapter (Speaker and Ringer settings not available on Firefox). You may have to add this field to the layout manually. [See troubleshooting](#).

12. Phone Type Settings Turning this setting on allows the Agent to change their Phone Type in the CCP. You may have to add this field to the layout manually. [See troubleshooting](#).

Single Sign On Settings

The Amazon Connect CTI Adapter supports single sign on(SSO) via SAML integration. This allows customers that use a SAML provider for authentication into Amazon Connect. You will need the SSO URL for your provider and the Relay State settings for your Amazon Connect instance.

For general information on configuring SAML for Amazon Connect, please refer to: [Amazon Connect Administrator Guide: Configure SAML for Identity Management in Amazon Connect](#).

If you wish to use **Salesforce** as your identity provider for Single Sign On, please follow the setup instructions in [Appendix B - Configuring Salesforce as Your Identity Provider](#).

For information about configuring specific SAML providers to work with Amazon Connect:

- AWS Single Sign-On
- Okta

Once you have your SAML integration working with Amazon Connect, you will need to create the Amazon Connect Single Sign On URL and validate that it works correctly, then configure the Lightning CTI adapter and login the agent.

Note: With the new Amazon Connect instance urls (*.my.connect.aws) you must put the full URL into the **Amazon Connect Instance** field in the AC CTI Adapter record for SSO to work. Ex: using <https://myinstance.my.connect.aws> instead of `my instance`.

Identify the SSO URL components

In order to authenticate with Amazon Connect, you need your IdP login URL from your SAML provider and a relay state URL that will redirect the authenticated user to your Amazon Connect instance.

Your IdP Login URL will resemble the following (Salesforce is shown):

```
https://m*****run-dev-ed.my.salesforce.com/idp/login?app=0sp0N00000Caid
```

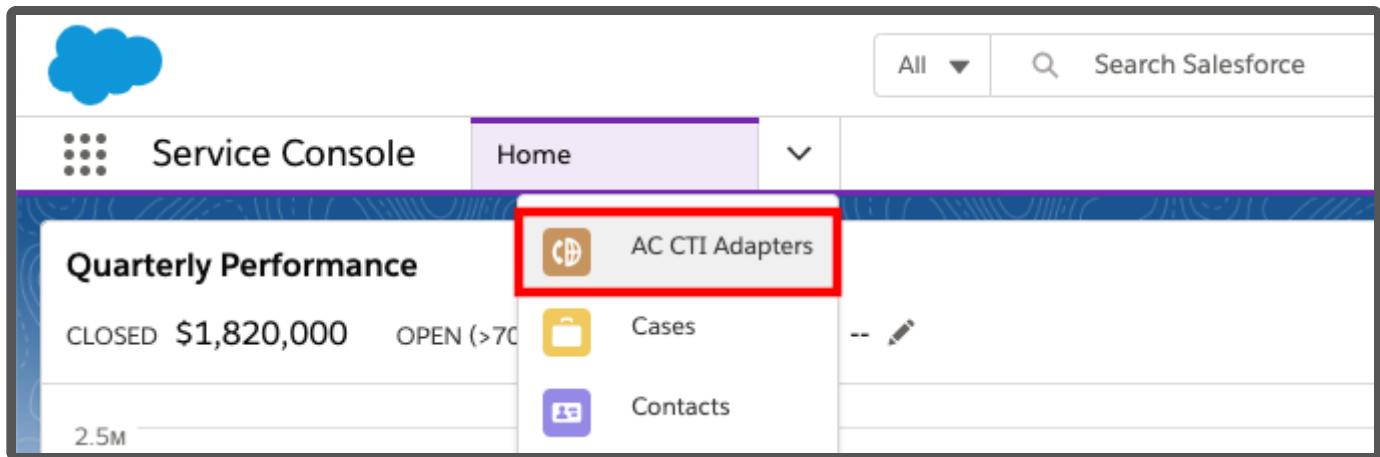
The 'RelayState' will be in the following format (replace `us-west-2` with the region you are using):

```
https://us-west-2.console.aws.amazon.com/connect/federate/InstanceId?destination=%2Fconnect%2Fccp
```

Configure the CTI Lightning Adapter in Salesforce

Now we are ready to complete the last step in the configuration process: Adding the SSO settings to the Lightning Adapter. This will configure the adapter to authenticate via SSO and redirect to the Amazon Connect Contact Control Panel once authentication completes.

1. Log in into your Salesforce org and go to the **Service Console**
2. Expand the **navigation menu** by selecting the down arrow and choose **AC CTI Adapters**.



3. Select **ACLightningAdapter**

4. Scroll down to the Single SignOn (SSO) section and choose the pencil icon of either field to edit

▼ Single SignOn (SSO)

SSO Url

SSO Relay State

5. For the SSO Url, paste your IdP login URL up to the first question mark (if one exists). A couple of examples are provided: Salesforce:

```
https://m*****run-dev-ed.my.salesforce.com/idp/login?  
app=0sp0N000000Caid
```

Microsoft ADFS:

```
https://sts.yourcorp.com/adfs/ls/idpinitiatedsignon.aspx
```

6. Paste this portion of the URL into the **SSO Url** field

▼ Single SignOn (SSO)

SSO Url

https://sample-dev-ed.my.salesforce.com/idp/login

7. For the SSO Relay State: IF you had a question mark in your login URL, paste everything AFTER the question mark into the SSO Relay state field, then add &RelayState= to the end, and append your relay state URL. For example:

```
app=0sp0N00000Caid&RelayState=https://us-west-2.console.aws.amazon.com/connect/federate/InstanceId?destination=%2Fconnect%2Fccp
```

IF you did not have a Question Mark, then enter &RelayState= into the SSO Relay State field and append your relay statue URL to it. For example:

```
&RelayState=https://us-west-2.console.aws.amazon.com/connect/federate/instanceId?destination=%2Fconnect%2Fccp
```

8. Example of a completed SSO section (Salesforce is shown)

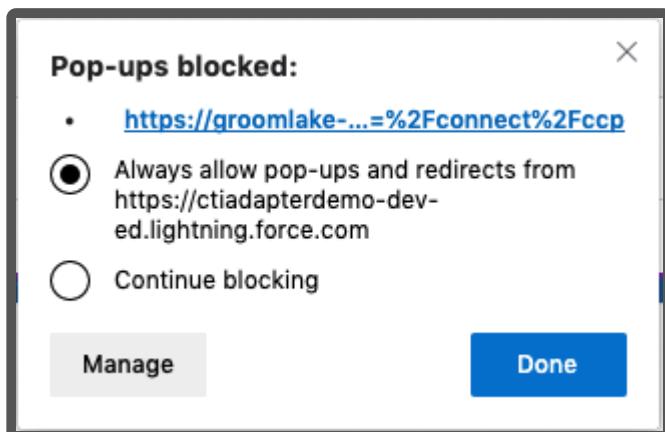
Single SignOn (SSO)

SSO Url	<input type="text" value="https://sample-dev-ed.my.salesforce.com/idp/login"/>
SSO Relay State	<input type="text" value="app=0sp6g000000XZyd&RelayState=https://us-west-2.console.aws.amazon.com/connect/federate/YOUR-INSTANCE-ID?destination=%2Fconnect%2Fccp"/>

9. Choose **Save**

10. Refresh your browser to make the changes take effect

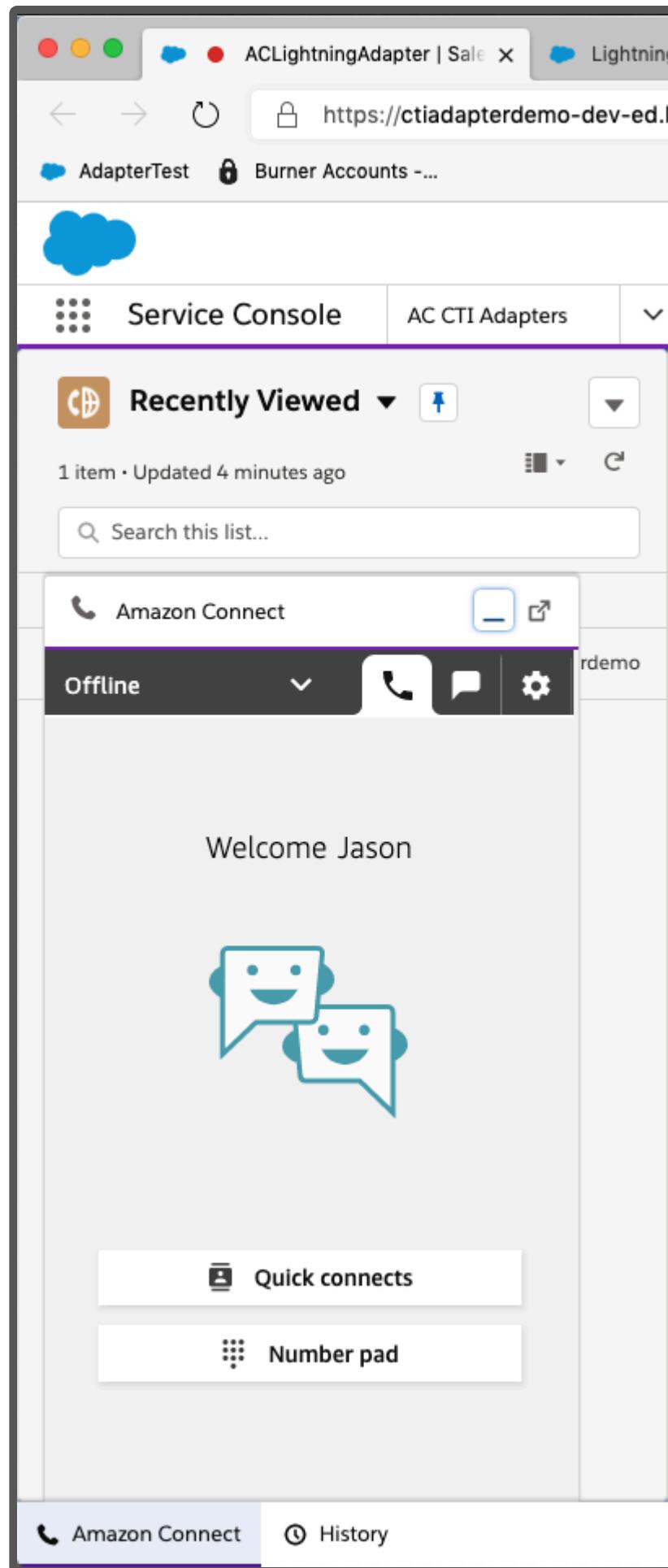
- a. **NOTE:** If you receive a blocked popup warning, select the warning and change the setting to always allow popups from your Salesforce org, then refresh the browser again



11. Select the **phone icon** in the console toolbar to open the CCP Note: You may also receive popups to allow notifications and microphone access. Please accept both.

12. Click the Sign into CCP button

13. You should now see the authenticated and logged in CCP



14. SSO Configuration is complete

CTI Attributes

CTI Attributes provide the ability to reference and display contact attribute data within the Amazon Connect Contact Control Panel (CCP). This allows for easy access to data or URLs that may be necessary for agents to perform tasks external to Salesforce. Adding attributes does not import data directly into Salesforce. Instead, it is simply available in the CCP for the life of the contact.

Attribute Properties

When configuring CTI attributes, you will need to complete the configuration with the following information:

- **CTI Attribute Name:** the user-friendly name that will identify this attribute configuration. This is not the name or key of the attribute itself.

Note: in v5.16 there is a bug where this has to be the same as the contact attribute name/key.

- **Label:** will be displayed in the CCP as the label for the attribute value.
- **Display:** indicates how this attribute should be displayed. Options are:
 - --None--: this attribute will not be displayed, however it will be available for use. Typically, this is used to define attributes that will be used in URLs.
 - Key-Value: the attribute label and value will both be displayed as a key-value pair
 - Key: only the label is displayed. This can be used to create sections in the attribute list. For example, you could have an "Address" label followed by individual attributes for street, city, state, country, postal code, etc
 - Value: only the value is displayed. This can be used when displaying several values under one section or when displaying a URL that needs no label.
- **Type:** indicates if this is a text or URL attribute
- **Style:** allows you to specify a CSS style rule for the display of this attribute. The style will apply to both the label and the value.
- **Format:** the format allows you to define which contact attributes will be used in the value of this CTI attribute. Contact attributes are referenced by their key name enclosed in double curly

braces. For example, an Amazon Connect contact attribute of accountId would be referenced as `accountId`.

- **Active (checkbox):** indicates if this CTI attribute is active
- **Default Value:** value to be displayed if the contact attribute referenced is not found

Once you set the CTI attributes, you access them by choosing the appropriate icon during a connected contact



Phone



+1 2
🕒 0

Attributes

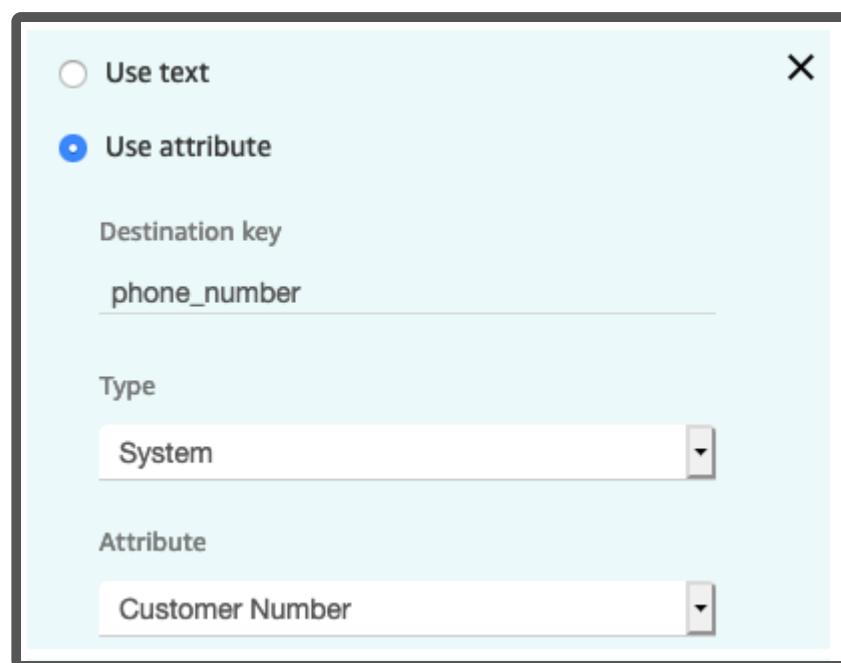
Contact ID 76a33679-... 📁

CTI Attributes Example Walkthrough

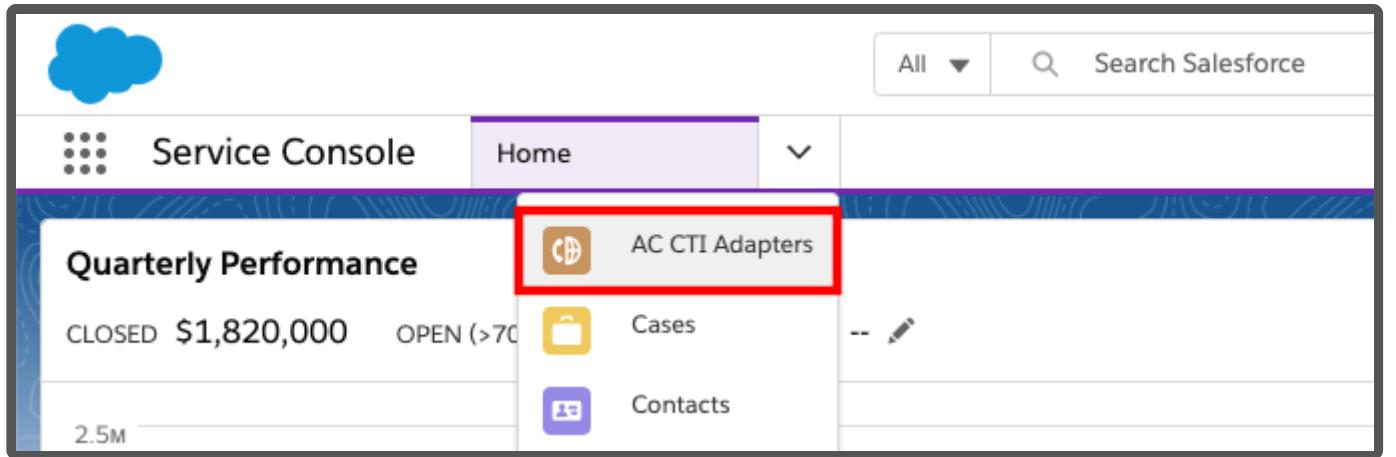
Since there are endless use cases for CTI attributes, this guide will walk through a couple examples that show you how both text and hyperlink based attributes are configured, presented, and used. These examples are not intended to remain in your configuration and are instead designed to provide you with the experience of configuring a functional attribute.

Adding a Text-based CTI Attribute

In this example, we will walk through creating a new CTI Attribute based on a contact attribute named "phone_number" and add it to the CCP. In our scenario, the contact flow has set this attribute using input from the customer to indicate their phone number of record. In order for this example to work, your contact flow must also set a contact attribute named "phone_number"

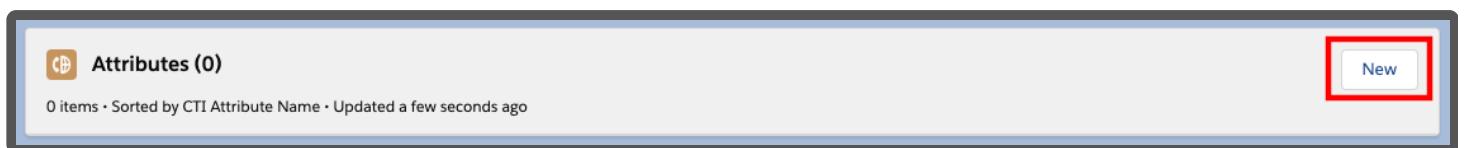


1. Log in into your Salesforce org and go to the **Service Console**
2. Expand the **navigation menu** by selecting the down arrow and choose **AC CTI Adapters**.



3. Select **ACLightningAdapter**

4. Scroll down to the **Attributes** section and select New



5. Provide a **CTI Attribute Name** value, for example: customer_phone

6. Provide the **Label** name, for example: Callback Phone

7. Select the **Display** option, in this case: Key-Value

8. Select Text as the **Type**

9. For **Style**, enter the following: color: red

10. In the **Format** field, enter `phone_number` to reference the incoming contact attribute

11. Set **Default Value** to unk

12. Choose Save

CTI Adapter

ACLightningAdapter

* CTI Attribute Name

customer_phone

* Label

Callback Phone

* Display

Key-Value

* Type

Text

Style

color: red

* Format

{{phone_number}}

Active



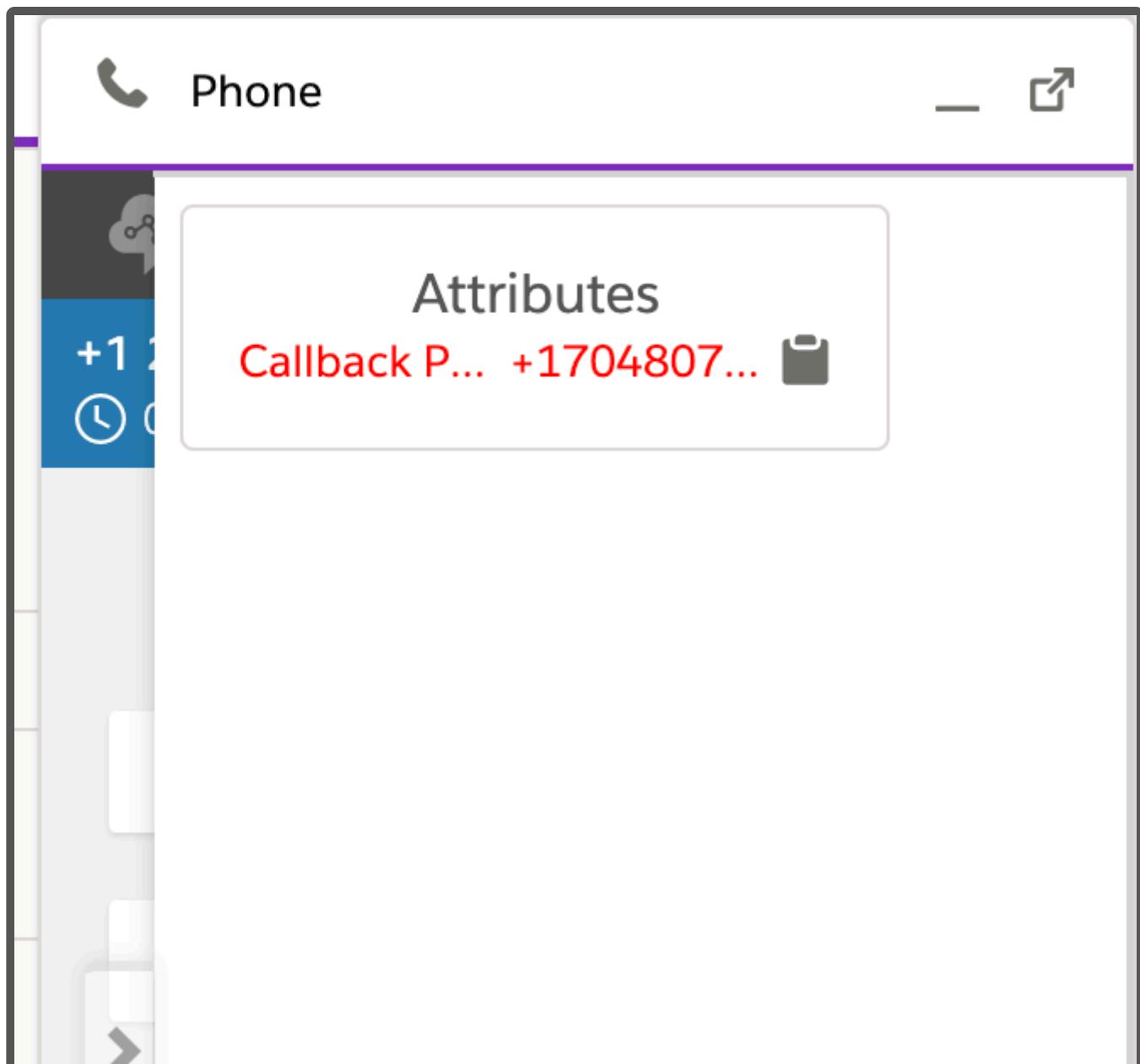
Default Value

unk

13. Refresh your browser

14. Place a new call into your Amazon Connect instance and accept the call as an agent

15. Once the call is connected, select the text attribute icon to expand the CTI Attributes



16. Note the Style formatting. Also note that you can quickly copy the content of the attribute by selecting the clipboard icon.
17. Disconnect the contact.

Adding a Hyperlink-based CTI Attribute

In this example, we will walk through creating a new hyperlink CTI Attribute that incorporates a contact attribute named "postal_code" and add it to the CCP. In our scenario, the contact flow has set this attribute using a data query into Salesforce. In order for this example to work, your contact flow must also set a contact attribute named "postal_code"

Use text

Use attribute

Destination key

postal_code

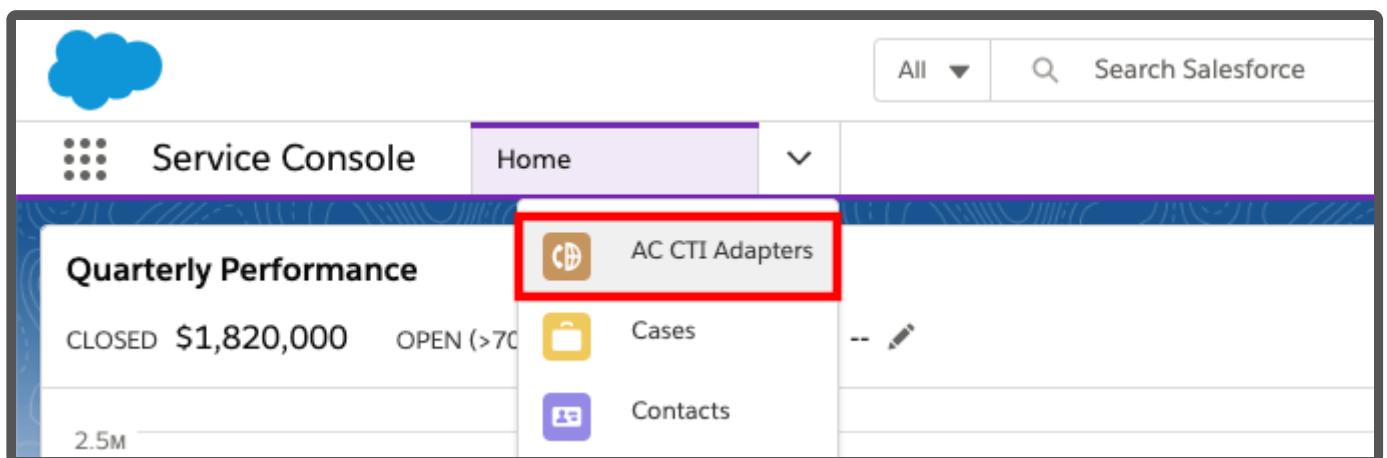
Type

External

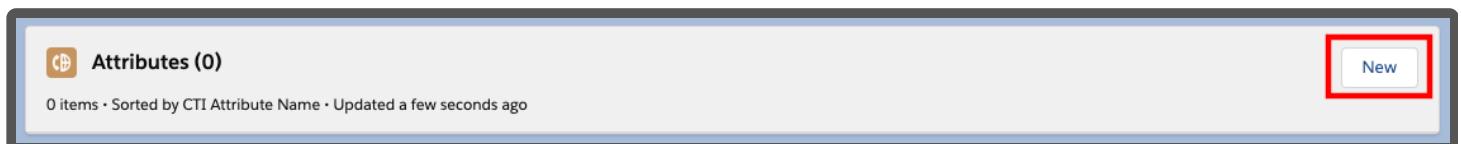
Attribute

postCode

1. Log in into your Salesforce org and go to the **Service Console**
2. Expand the **navigation menu** by selecting the down arrow and choose **AC CTI Adapters**.



3. Select **ACLightningAdapter**
4. Scroll down to the **Attributes** section and select New



5. Provide a **CTI Attribute Name** value, for example: postal_code
6. Provide the **Label** name, for example: MapIt
7. Select the **Display** option, in this case: Key-Value
8. Select Hyperlink as the **Type**
9. Leave **Style** blank

10. In the **Format** field, enter

```
https://www.google.com/maps/search/postal_code
```

to append the incoming contact attribute to the URL

11. Set **Default Value** to unk

12. Choose Save

Information

* CTI Adapter	ACLightningAdapter
* CTI Attribute Name	postal_code
* Label	MapIt
* Type	Hyperlink
* Format	https://www.google.com/maps/search/[object Object]
Default Value	unk
* Display	
Key-Value	
Style	
Active <input checked="" type="checkbox"/>	

13. Refresh your browser

14. Place a new call into your Amazon Connect instance and accept the call as an agent

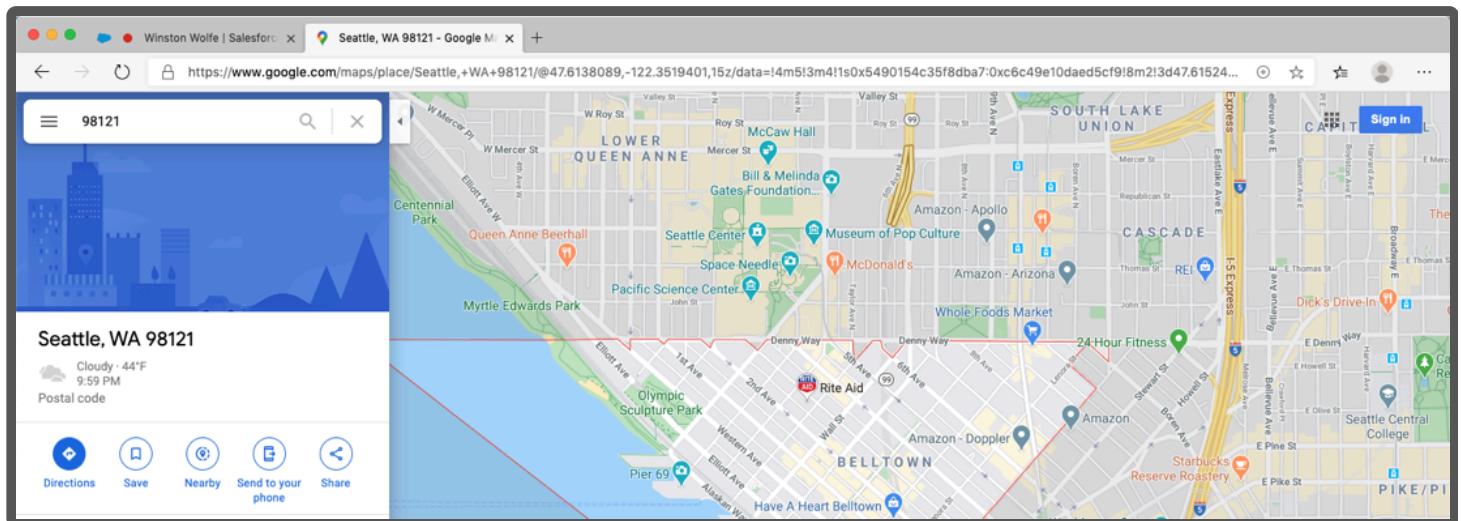
15. Once the call is connected, select the hyperlink attribute icon to expand the CTI Attributes



Phone



16. Select the URL and observe the page load



17. Disconnect the contact.

CTI Attribute Additional Features

Enabling CTI Attribute Additional Features

The additional CTI Attribute features allow you to further customize CTI Attributes.

1. In Service Console, navigate to your CTI Adapter

The screenshot shows the Service Console interface. The top navigation bar has tabs for 'Service Console' and 'AC CTI Adapters'. The main content area is titled 'AC CTI Adapter ACLightningAdapter'. On the left, there's a sidebar titled 'Recently Viewed' with one item listed: 'ACLightningAdapter' (updated 4 minutes ago). The main panel shows the adapter's configuration details, including fields for CTI Adapter Name, Amazon Connect Instance Alias, Softphone Popout Enabled (checked), Medialess (unchecked), and Single SignOn (SSO) settings. At the bottom of the main panel, there are links for Phone, History, Notes, Macros, and Omni-Channel (Offline).

2. Scroll down to the features section of your AC CTI Adapter and select new

The screenshot shows the 'Features' tab selected in the AC Feature configuration screen. Below the tab, it says 'Features (0)'. In the top right corner of this area, there is a 'New' button, which is highlighted with a red box.

3. Set the AC Feature Name to **FEATURE_CTI_ATTRIBUTES**

4. Fill the value text box to contain the following settings:

- ShowAttributesIfEmpty** (Boolean, default true): show attributes text box when contact has no attributes
- ShowAllAttributes** (Boolean, default false): show all attributes, including attributes with no values

* AC Feature Name

FEATURE_CTI_ATTRIBUTES

Value

ShowAttributesIfEmpty: true
ShowAllAttributes: true

Active



CTI Adapter

ACLightningAdapter

5. Select **Save**

CTI Flow

The CTI Adapter provides a mechanism to customize the behavior of the adapter based on your business needs without needing to edit the underlying Visualforce pages, which could negatively impact overall adapter function. This is accomplished through CTI Flows.

A CTI Flow consist of "actions" that represent an API call to parts of Salesforce or Amazon Connect API. Like a JavaScript function, each action can take inputs and provide outputs, or returns values, that you can use from other actions.

Create CTI Flow

To create a new CTI Flow, log in into your Salesforce org and go to the **Service Console**. Expand the **navigation menu** by selecting the down arrow and choose **AC CTI Adapters**.

The screenshot shows the Salesforce Service Console interface. At the top, there's a blue cloud icon, a search bar with the placeholder 'Search Salesforce', and a 'Service Console' button. Below the header, a navigation menu is expanded, showing options like 'Home', 'AC CTI Adapters' (which is highlighted with a red box), 'Cases', and 'Contacts'. On the left, there's a 'Quarterly Performance' dashboard with metrics like 'CLOSED \$1,820,000' and 'OPEN (>70)'. The overall theme is purple and blue.

Select **ACLightningAdapter**. Scroll down to the **CTI Flows** section and select New to create a new CTI Script.

The screenshot shows a list of 'Scripts (3)' with a header bar containing a gear icon, a refresh icon, and a 'New' button. The 'New' button is highlighted with a red box.

Provide a user-friendly name in the **CTI Flow Name** field. And click **Save**.

The screenshot shows the 'New CTI Script' configuration form. It includes fields for 'Information' (CTI Script Name: 'Set Agent Offline on Login', CTI Adapter: 'ACLightningAdapter'), 'Active' (checked), 'Debugger Breakpoint' (unchecked), 'Source' (Amazon Connect Agent), 'Event' (onInit), and a 'Description' (Script to set agent to Offline when first logging in).

This will take you to a form where you can fill in name and adapter of the CTI Flow. There are a couple of fields that you may be unfamiliar with: **Source** and **Event**.

Let's look at **Source** field first.

* Source

Amazon Connect Voice Contact

--None--

Initialization

Amazon Connect Agent

✓ Amazon Connect Voice Contact

Amazon Connect Queue Callback Contact

Amazon Connect Chat Contact

Salesforce Agent

You can think of Source as the "origin" of the CTI Flow. There are currently 7 sources: Initialization, an Agent on Connect, Voice Contact on Connect, Queue Callback Contact on Connect, Chat on Connect, Salesforce Agent or Salesforce UI.

Each source comes with a set of events that you can hook into, i.e. your CTI Flow will be executed when one of these events fire. Typically, you will have only one flow for a combination of a source and an event. (You can find out more about sources and events in [Appendix C - CTI Flow Sources and Events](#).)

For the purposes of this example, we selected **Amazon Connect Voice Contact** source and **onConnecting** event. Now click Save and on the next page scroll down till you find the **CTI Flow** section.

Details

Information

CTI Flow Name

Create Screenpop

Source

Amazon Connect Voice Contact

Description

Created By



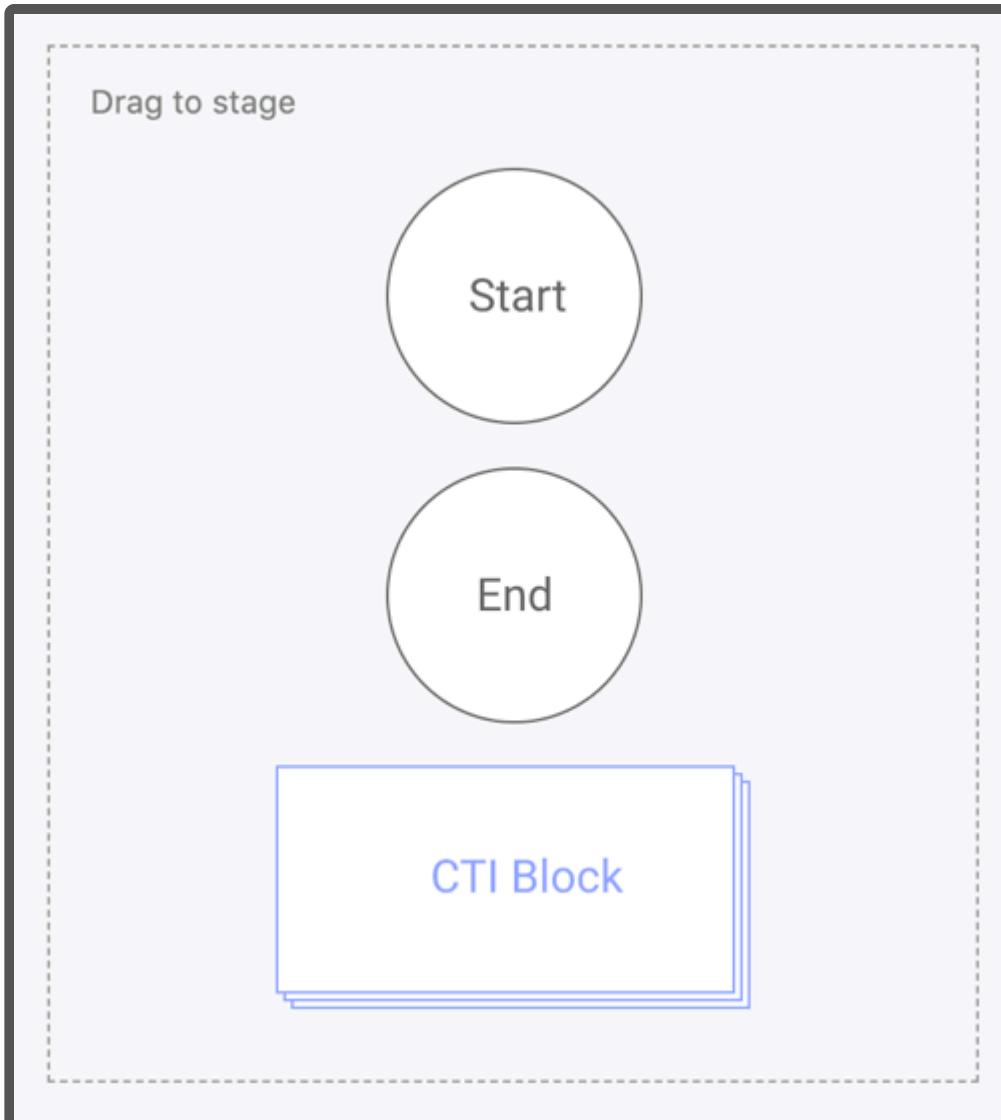
Amazon Connect, 7/23/2020 9:10 AM

CTI Flow

Main Menu

Save

Let's build a CTI Flow that opens a screenpop in Salesforce when a voice call comes.



You can start using by dragging the item called **CTI Block** from the sidebar in the Main Menu over the stage, which is marked by a grid pattern.

When you drop the block, you will see a modal titled **Explorer**. This modal contains a list of actions you can choose from.

Explorer

Search

phone

Categories

Filter by category

Tags

Filter by tag

Showing 13 actions

Save search

Searches (Clear)

phone

date

Format Phone Number	Format Phone Number (E164)
Formats a phone number for a country code. Parameters > What it calls: ac.Utils.Common.formatPhoneNumber(...) Select	Formats a phone number for a country code in E164 format. Parameters > What it calls: ac.Utils.Common.formatPhoneNumberE164(...) Select
Get Softphone Layout	Show Softphone Panel
The query to get softphone layout. What it calls: ac.Utils.Salesforce.getSoftphoneLayout() Select	The command to show softphone panel. What it calls: ac.Utils.Salesforce.showSoftphonePanel() Select

In the **Search** field, search for **Phone** and Select the action called **Get Customer Phone Number** from the results on the right.

Change type ▾

Get Customer Phone Number

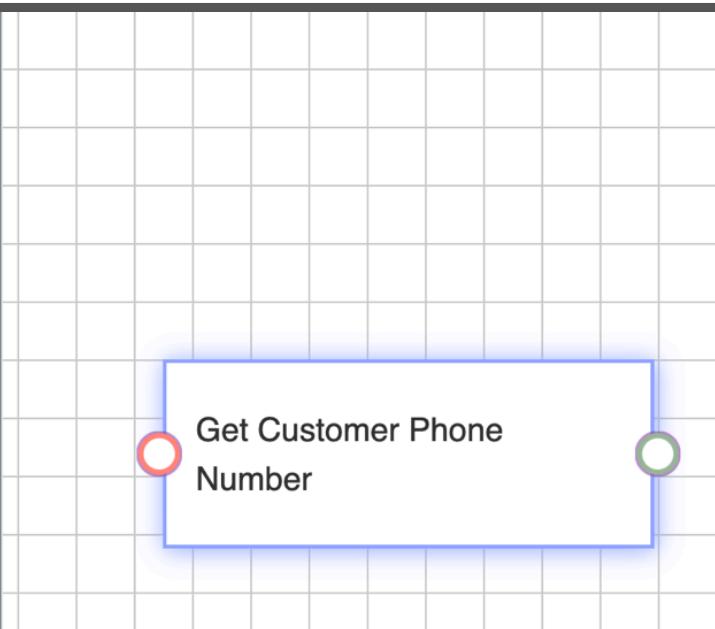
ID: uid-0

Remove About this action

Return Values

This action has a return value. It returns the following fields. You may use these fields in the input fields of connected actions.

phone	Phone number of the caller.
country	Country of the phone number.



You should now see a block on the stage for the action you selected, and the sidebar will display some information about this action, including its return value.

(Note: If you'd like to change the label of the action, doubleclick on it. This will open a text editor. Make your changes and when you're finished click outside the node to save your label.)

Some actions can be configured using input fields to provide arguments to function calls, as well. This action does not have any input fields, and returns two values ---- **phone** and **country**.

Now let's drag another CTI Block over the stage and find an action called **Search and Screenpop**.

Change type ▾

Search And Screenpop

ID: uid-9

Remove About this action

Arguments

searchParams ⓘ

Enter a value

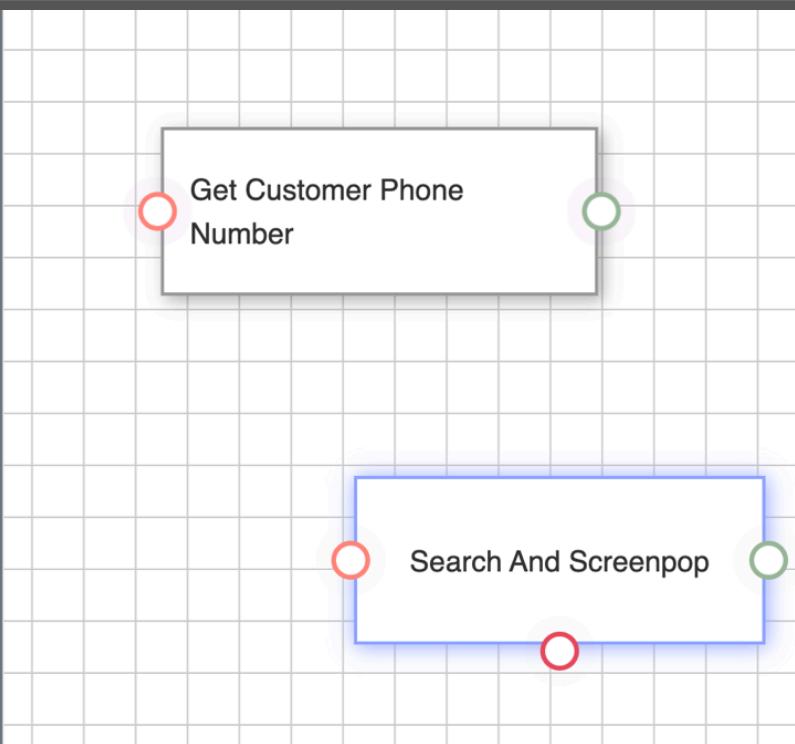
queryParams ⓘ

Enter a value

defaultFieldValues ⓘ

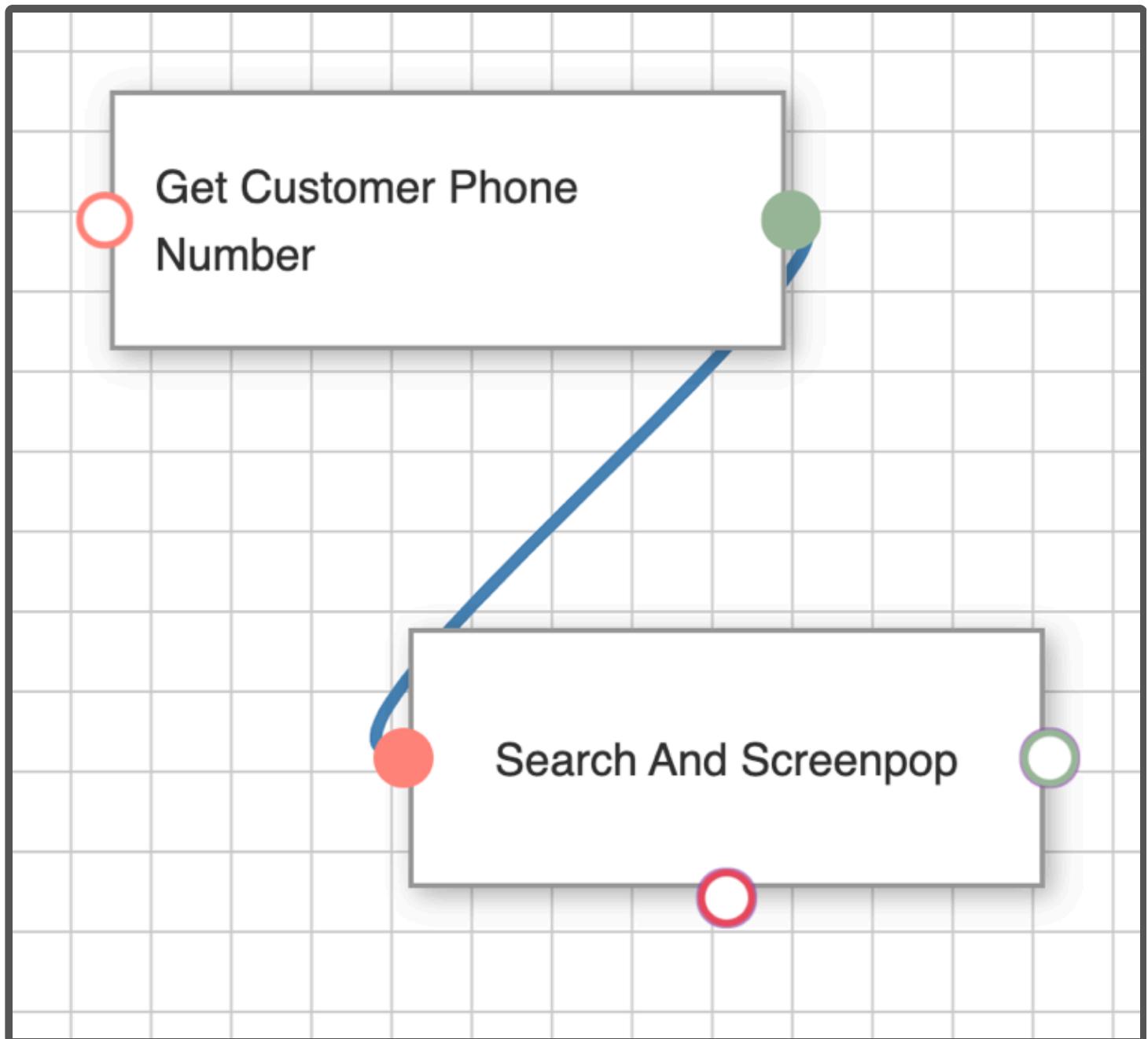
Add a field

deferred ⓘ



Connect these blocks by clicking the green socket (green means "done") on **Get Customer Phone**, which will display a blue line that tracks your mouse cursor around the stage.

Now, click on the pink socket, i.e. the **input** socket, which is to the left of the **Search and Screenpop** block. If the connection is successful, the sockets will turn into a solid color and the blue line will connect them. (There are some restrictions on which sockets you can connect together. For example, you cannot connect output of an action to its own input socket or connect two inputs.) If you are not happy with this connection, you can hover over it and double click to remove.



Now we'd like to get the phone number of the customer and use it in **Search and Screenpop**. Here is a tip: if two actions are connected, you can use the return values of the first action in the input fields of the next action. (You can even use the return values of actions connected to the last action, and the ones connected to that, and so on.)

This action has only two options, and we want to use the one called "phone" for this field.

Change type ▾

Search And Screenpop

ID: uid-2

Remove

About this action

Arguments

searchParams

GET CUSTOMER PHONE NUMBER (UID-0)

phone

country

Add a field

deferred

callType

Get Customer Phone Number

Search And Screenpop

Search And Screenpop

ID: uid-9

Remove

About this action

Arguments

searchParams

ValueOf

queryParams

Add New Value

Add a field

deferred

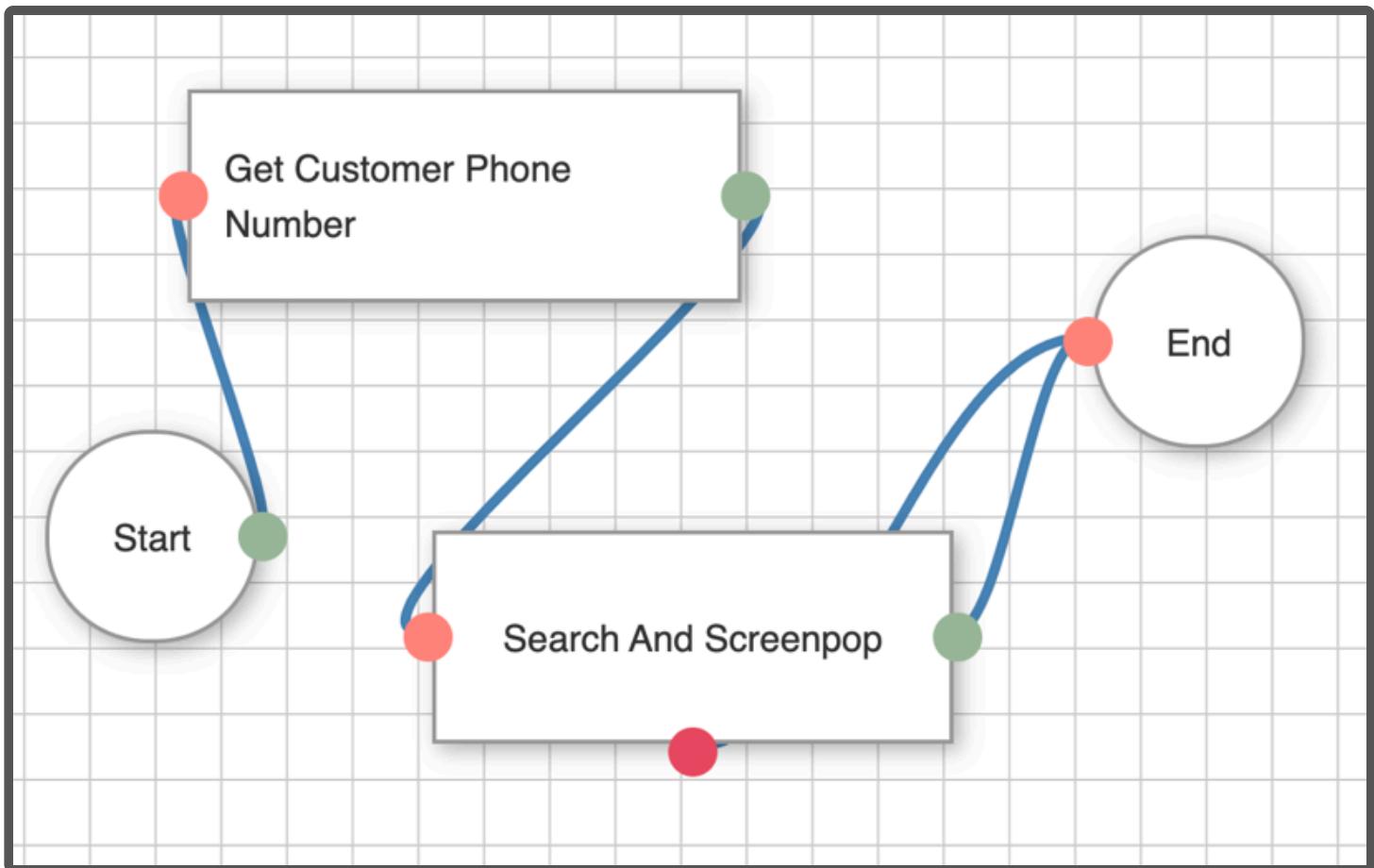
callType

Get Customer Phone Number

Search And Screenpop

If you want to enter a custom input value, you can type that, and select **Add New Value** from the dropdown.

And make sure to set **callType** to "inbound." Finally, add the **Start** and **End** nodes and connect everything together.



When you're finished, click **Save** in the sidebar. That's it. You created your first CTI Flow.

To test your flow, go to your **Service Console**, and make a call from a number that is in the profile of a Contact. As the call is displayed in your CCP dashboard, Salesforce will pop open the contact of the caller in a separate tab.

Accessing CTI Flow Block Values

Internally, CTI flows are organized as JavaScript Object Notation (JSON) objects, and access to them is facilitated through JSON Paths.

JSON Paths

JSON paths help you find specific data within a JSON structure, similar to giving directions to locate hidden treasure on a map.

Basic Syntax:

- Use dot notation (.) to traverse through object properties.
- Use square brackets ([]) to index into arrays.

Accessing Object Properties

Consider this JSON object below:

```
{  
  "name": "John",  
  "age": 25,  
  "address": {  
    "city": "New York",  
    "zip": "10001"  
  }  
}
```

To retrieve the value of the "name" field in this JSON object, you would utilize `$.name`. The same approach applies to accessing the value of "age" through `$.age`. If you wish to access the "city" value, you would use `$.address.city`.

Consider this next JSON Object:

```
{  
  "fruits": ["apple", "orange", "banana"]  
}
```

To access the different fruits, you should use the following format: `$.fruits[0]` or `$.fruits[2]`. It's important to note that the first element (apple) is accessed with "0" rather than "1" due to arrays starting their indexing from 0.

Accessing CTI Flow Object Properties

Having covered the fundamental concepts of accessing JSON objects, here is an illustration of a CTI flow JSON:

```
LogUtils.ts:41 [CTI ADAPTER]: [FLOW] [AC_clickToDialFlow]: actions: {"uid-0": {"success":true,"results":{"value":{"number":"+12345678900"}}, "exception":null}, {"success":true,"results":{"value":"+12345678900"}, "exception":null}, "uid-54": {"endpointARN":null,"endpointId":null,"type":"phone_number","name":null,"phone":44}: {"success":true,"results":null,"exception":null}, "uid-18": {"success":true, {"type": "Task", "Id":null, "CustomField__c": "RandomMessage"}, "uid-20": {"success":true,
```

It is presented in this format typically, but for this demo, it will be more convenient to conceptualize it like this (condensed for brevity):

```
{
  "actions": {
    "uid-39": {
      "success": true,
      "results": {
        "value": {
          "number": "+12345678900"
        }
      },
      "exception": null
    },
    "uid-29": {
      "success": true,
      "results": true,
      "exception": null
    },
    "uid-54": {
      "success": true,
      "results": {
        "endpointARN": null,
        "endpointId": null,
        "type": "phone_number",
        "name": null,
        "phoneNumber": "+12345678900",
        "agentLogin": null,
        "queue": null
      },
      "exception": null
    },
    "uid-17": {
      "success": true,
      "results": {
        "value": {
          "type": "Task",
          "Id": null,
          "CustomField__c": "RandomMessage"
        }
      }
    }
  }
}
```

Similar to the approach used for accessing values in smaller JSON objects, you can apply the same methodology here. To retrieve the phone number in the CTI flow block with "uid-54," you can use the following syntax: `$.actions.uid-54.results.phoneNumber`.

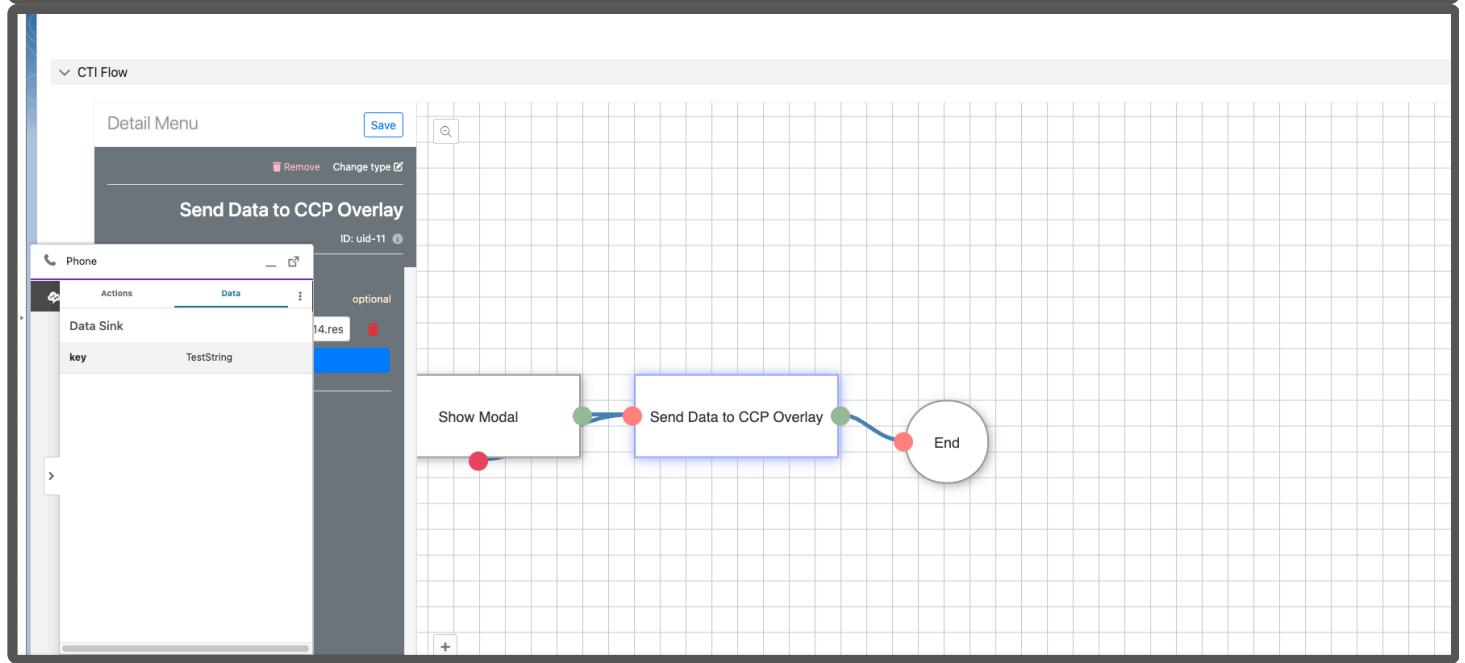
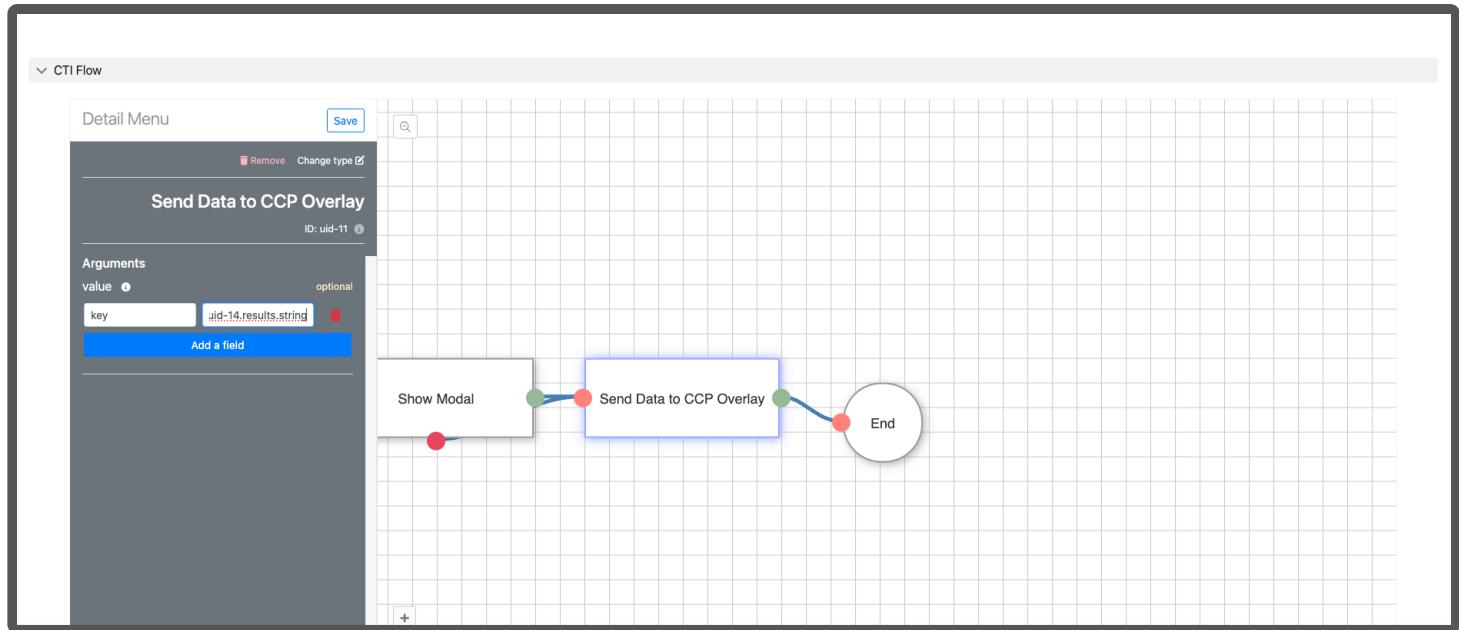
For certain CTI flow blocks, the return values can be more intricate. Take, for instance, the CTI flow block with "uid-17," which generates a Task record in Salesforce. To access the values `type`, `Id`, or `CustomField__c`, you need to use the format: `$.actions.uid-17.results.value.[0].CustomField__c`. The use of "[0]" is essential in this case, as the particular CTI flow block can return multiple Task objects stored as an array.

Note: It's crucial to verify the return values of a CTI flow object before attempting to access its value, as not every CTI flow block returns a value. Otherwise, you may receive `undefined`, indicating the requested value does not exist.

Why Would I Use This?

In most instances, direct access to CTI flow values is unnecessary, as return values are selectable through a dropdown menu in the CTI flow block.

However, for scenarios where the dropdown menu is inaccessible, such as with CTI flow blocks like "Send Data to CCP Overlay," accessing the value directly becomes more practical.



Presence Sync Rules

The CTI Adapter supports bidirectional synchronization of agent state between Amazon Connect and Salesforce Omnichannel. This allows you to tightly control agent availability for different contact/media types dependent on current agent state. This section of the guide assumes that you have Omnichannel configured appropriately. If you do not and wish to test this function, please refer to the section [Configure Salesforce Omnichannel for Testing](#).

NOTE: In order for Presence Sync to work, the CTI Adapter must be configured to allow it. See [CTI Adapter Details](#) for more information. This feature is not turned on by default.

NOTE: After Salesforce Winter '22 Release, users need to have View Setup and Configuration OR View DeveloperName permission via a profile or permission set to use this feature. See [New Permission Requirements for DeveloperName Field](#) for more information.

Presence Sync Rules are evaluated based on specific events. The available events are:

- **Connect Agent State Change:** The Connect agent's state has changed.
- **Salesforce Agent State Change:** The Salesforce agent's state has changed.
 - If a rule is set up with this event and the new state is set to "Offline", this will not trigger Salesforce Agent Logout
- **Salesforce Agent Logout:** The Salesforce agent has logged out
 - Logging out of Omnichannel does not automatically log you out of Connect or set CCP to offline. If you want this functionality, you will need to set up a Presence Sync rule.
 - Rules triggered by Salesforce Agent Logout will only work if the rule is set to trigger when Salesforce New Agent Status is equal to the exact value "Offline" (case sensitive without quotes)
- **Salesforce Work Accepted:** The Salesforce agent has accepted work.
- **Salesforce Workload Changed:** The Salesforce agent's workload has changed.

Once the event is triggered, the CTI adapter will evaluate the provided criteria. The criteria is established by comparing Operand A, using standard comparator options, against Operand B. Possible options for Operand A and B are:

- **Connect Agent New State:** The Connect agent's new state value
- **Connect Agent Old State:** The Connect agent's old (previous) state value
- **Salesforce Agent New State:** The Salesforce agent's new state value
- **Salesforce Service Channel:** The service channel upon which the Salesforce agent has accepted work
- **Salesforce Previous Workload:** The Salesforce agent's previous workload
- **Salesforce Previous Workload Pct:** The Salesforce agent's previous workload expressed as a percent of configured capacity
- **Salesforce New Workload:** The Salesforce agent's new workload
- **Salesforce New Workload Pct:** The Salesforce agent's new workload expressed as a percent of configured capacity
- **Salesforce Configured Capacity:** The Salesforce agent's configured capacity

- **Static Value:** The user may provide a value. For example, a custom agent state name or other alphanumeric value. When Static Value is selected a "Value" field becomes visible to accept the users static value input.

Available comparators are:

- **Equal to:** Are Operand A and Operand B equal
- **Not equal to:** Are Operand A and Operand B not equal
- **Greater than:** Is Operand A greater than Operand B
- **Greater than or equal to:** Is Operand A greater than or equal to Operand B
- **Less than:** Is Operand A less than Operand B
- **Less than or equal to:** Is Operand A less than or equal to Operand B

Configuring Statuses

Presence Sync Rules require statuses in both Amazon Connect and Salesforce. In this example, we will add two additional statuses to each side of the configuration and prepare rules that sync both clients to the same state regardless of which agent sets the status. Essentially, you will configure the status sync similar to the following example:

When a sets status to b	Set x to y
Amazon Connect sets status to Available	Omnichannel to Available
Omnichannel sets status to Available	Amazon Connect to Available
Amazon Connect sets status to Working -- Phone	Omnichannel to Working -- Phone
Omnichannel sets status to Working -- Media	Amazon Connect to Working - Media

Amazon Connect System Statuses

The following Amazon Connect CCP statuses are system statuses that can be used in presence sync. Please note however that these statuses are restricted and you cannot set the Amazon Connect status to the below.

- Busy - agent is in a call
- Pending - agent is receiving a request for a queue callback

- PendingBusy - agent is receiving call
- CallingCustomer - agent is calling customer
- AfterCallWork - agent is in the after call work screen

Create Presence Statuses in Amazon Connect

Agents are responsible for setting their status in the Contact Control Panel (CCP). Typically, the only time an agent's status changes is when they manually change it in the CCP however Presence Sync Rules can automate the process when conditions are met.

Amazon Connect provides two default status values:

- Available
- Offline

You can change the name of these values, and you can add new ones. For example, you might add a status for Lunch, and another for Training. These and the default status values will be used for reporting, metrics, and resource management.

Note: When you add a new status, it will always be **Custom**, not routable.

Create an Amazon Connect status

1. Login to your Amazon Connect instance as an Administrator
2. From the left navigation, choose **Users**, then select **Agent status**



Dashboard

Configuration guide

Users

User management

Routing profiles

Agent status

Security profiles

Agent hierarchy

With Amazon Connect setup, it's easy to

control your channels of communication

and manage agents by phone number in order to re

route calls to the best agents in interactions

and manage the hierarchy of operations

3. Select **Add new agent status**

4. Provide a Status name and Description. Leave the Enabled checkbox selected.

Status name	Description	Type	Enabled for use in CCP
Lunch	Lunch	Custom	<input checked="" type="checkbox"/>

5. Select Save. Repeat as desired for the remaining statuses that you wish to add.

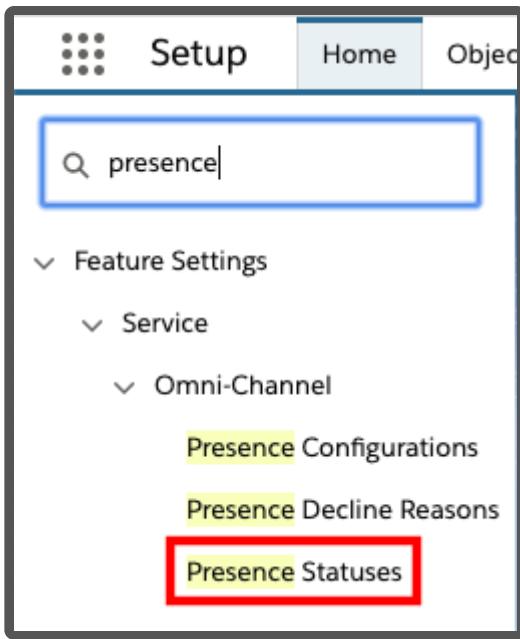
Create Presence Statuses in Salesforce

You will need to configure presence statuses to reflect the different presence states that you wish your Omni-Channel agents to enter. These do not need to match agent statuses in Amazon Connect exactly, but it does make it easier to track what you are doing.

Create a Salesforce presence status

1. Log in into your Salesforce org and go to **Setup**

2. In the **Quick Find** field, enter presence and choose **Presence Statuses** from the results



3. In the Presence Statuses page, choose New

4. Provide a status name, for example Lunch

5. Set the Status options appropriately, for example, Busy

a. For Online statuses, you will need to provide a channel. Please reference the [Omni-Channel documentation](#) for details

6. Choose Save

Presence Statuses

Let agents indicate when they're online and available to receive work items from a specific service channel, or whether they're away or offline.

Basic Information

Status Name	<input type="text" value="Lunch"/>
Developer Name	<input type="text" value="Lunch"/>

Status Options

Choose whether agents are online or busy when they use this status. Online statuses let agents receive new work items. Busy statuses make agents unavailable for new work items.

Online
 Busy

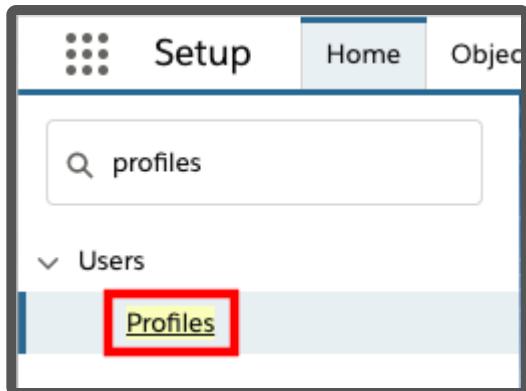
Buttons: Save | Cancel

7. Repeat as necessary for all desired statuses

Configure Enabled Service Presences Status Access in Salesforce

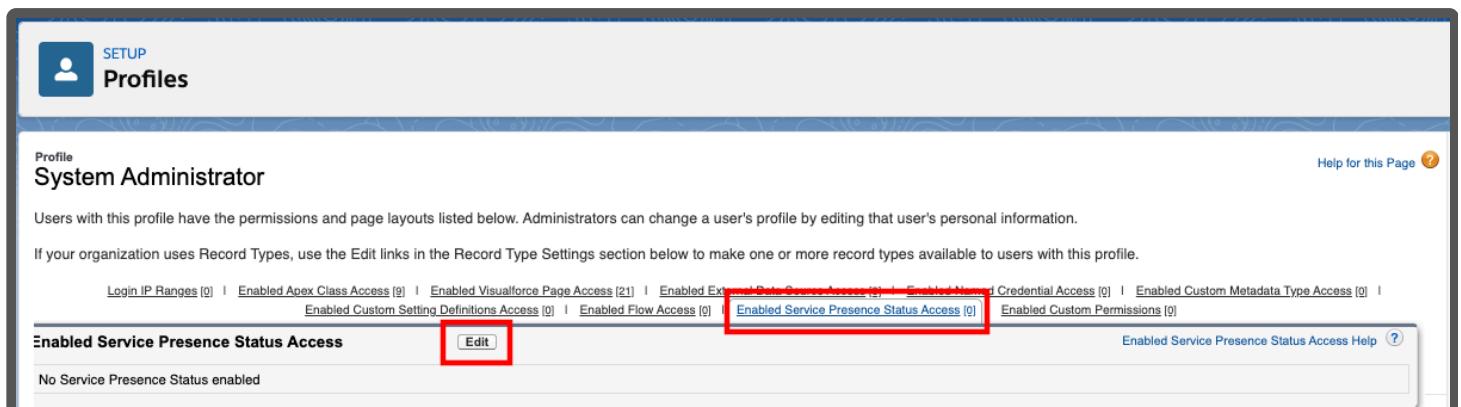
1. Log in into your Salesforce org and go to **Setup**

2. In the **Quick Find** field, enter profiles and choose **Profiles** from the results

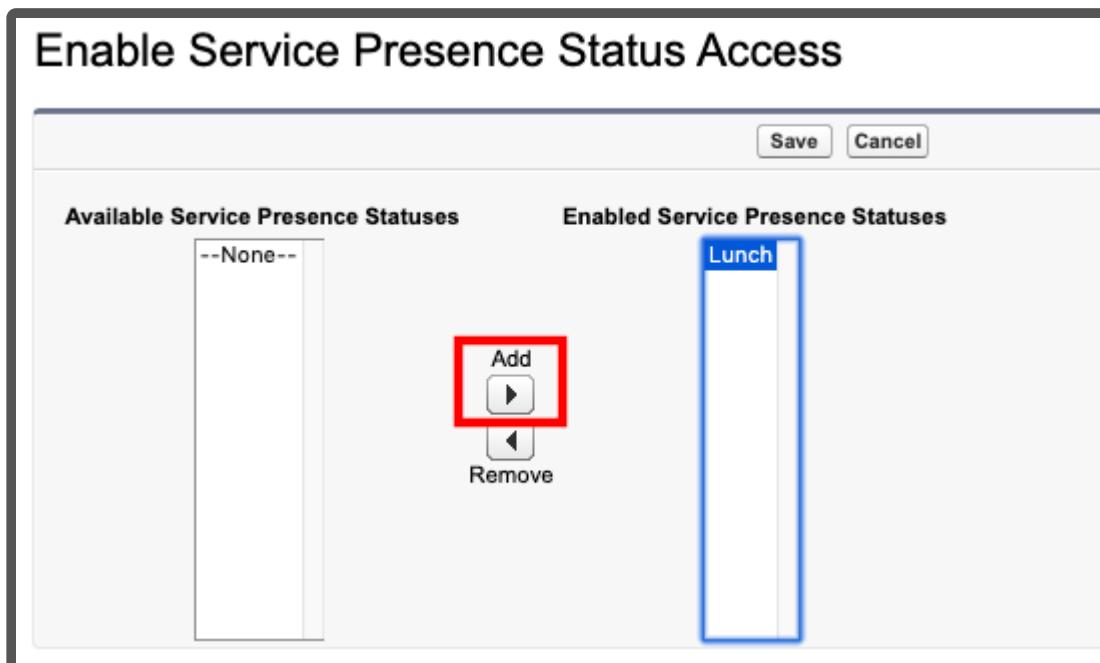


3. Select the profile assigned to your users

4. Hover over the Enabled Service Presence Status link and choose Edit



5. Select the available status from the left, then choose the Add button to add it to the Enabled Service Presence Statuses field



6. Select Save

7. Repeat as necessary for other statuses or profiles.

Configure Presence Sync Rules

The CTI Adapter provides a rules-based presence status synchronization system allowing for flexibility in mapping agent states between Amazon Connect and Salesforce Omni-Channel.

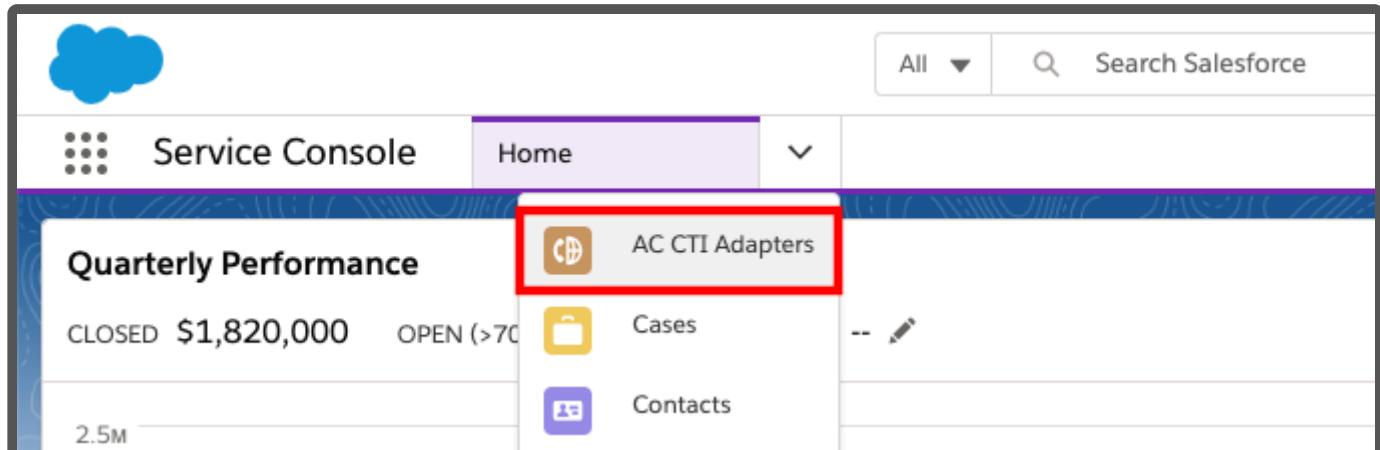
Presence synchronization actions may be configured based upon manual agent state changes (agent goes on break), system agent state changes (answering a call), omnichannel agent work (agent accepts an email), and omnichannel workload changes (agent completes an email) as examples.

As the scope of presence sync rules can vary wildly, this example will show you how to change state based on Amazon Connect agent state change and Salesforce agent state change.

Create a Presence Sync Rule

1. Log in into your Salesforce org and go to the **Service Console**

2. Expand the **navigation menu** by selecting the down arrow and choose **AC CTI Adapters**.



3. Select **ACLightningAdapter**

4. Scroll down to the **Presence Sync Rules** section

5. Select **New** to create a new presence sync rule

6. Provide a **Presence Sync Rule Name** to identify the use case of this rule. For example:
Connect agent switches to Lunch

Provide a user friendly name for this presence sync rule and specify if this rule is currently active.

* Presence Sync Rule Name

Connect agent switches to Lunch

Active



7. Select **Next**

8. For Source, select **Connect Agent State Change**, and select **Next**

9. For Operand A, choose **Connect Agent New State**

10. Set the Comparator to **Equal to**

11. Set Operand B to **Static Value**

12. For Operand B Value, enter **Lunch** (Or whatever state you have created in Amazon Connect)**

Configure the criteria that is evaluated to determine if the rule's action should be applied.

If the expressions configured here evaluates to 'true', the rule's action is applied. If the expression configured here evaluates to 'false', the rule's action is not applied.

* Operand A

Connect Agent New State



* Comparator

Equal to



* Operand B

Static Value



* Operand B Value

Lunch

13. Select **Next**

14. For Destination, choose **Salesforce Agent State**

15. Set the Value to **Lunch** (Or whatever state you have configured in Salesforce) **NOTE:** the static value for Salesforce Omni-Channel status is the Developer Name, not the Status Name

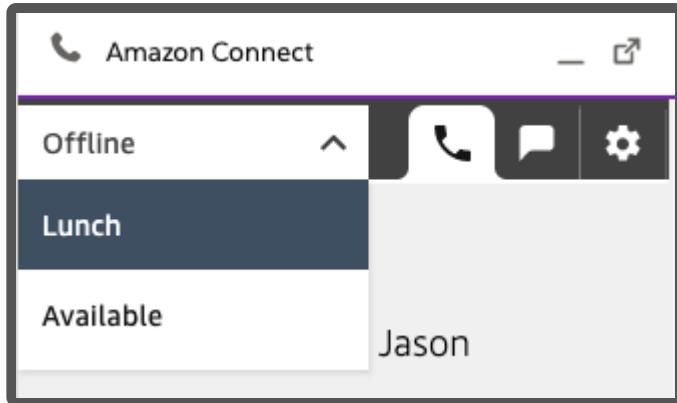
16. Select **Save**.

17. Refresh your browser

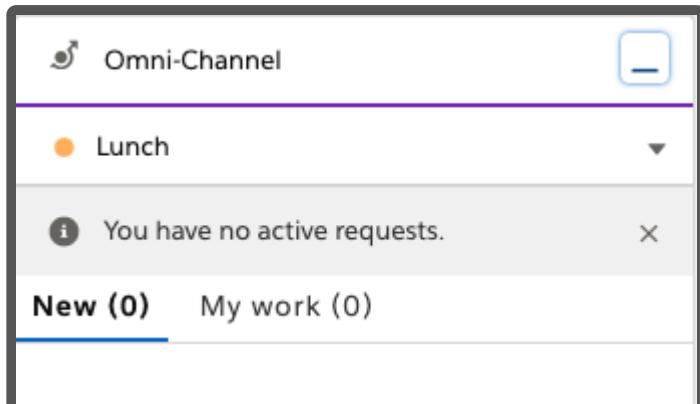
18. In the bottom left corner of the Service Console, select the CTI Softphone icon



19. Set your Amazon Connect agent status to Lunch



20. Observe that the Omni-Channel status switches to Lunch



21. Repeat this process as desired to configure your presence sync rules.

Localization

Prerequisites

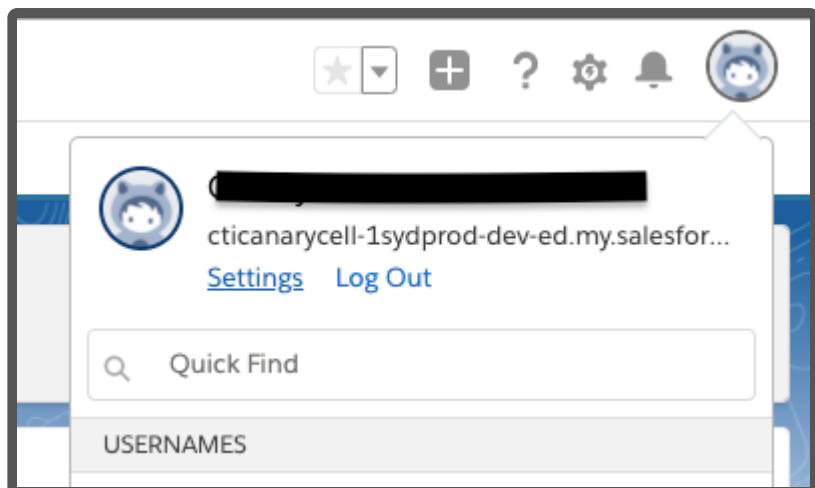
CTI Adapter will use Translation Workbench to maintain translated values for metadata and data labels in your Salesforce org. In order for that to work, you need to enable Translation Workbench in your org.

1. From Setup, in the Quick Find box, enter Translation Language Settings, and then select Translation Language Settings.
2. On the welcome page, click Enable.

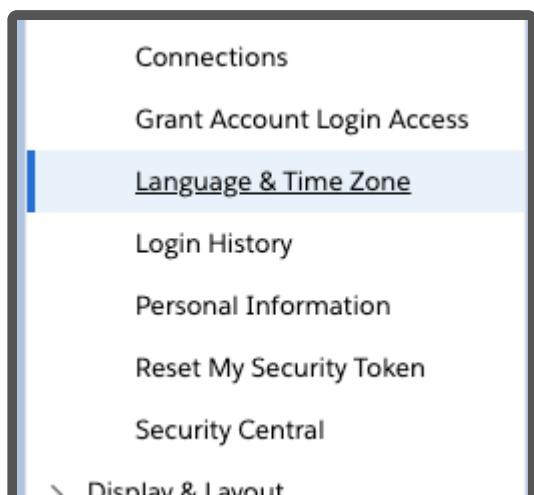
Setting your preferred language

Starting from v5.6, Amazon Connect Salesforce CTI adapter is localized in nine new languages: Spanish, French, Brazilian Portuguese, Korean, Italian, German, (Simplified/Traditional) Chinese, and Japanese.

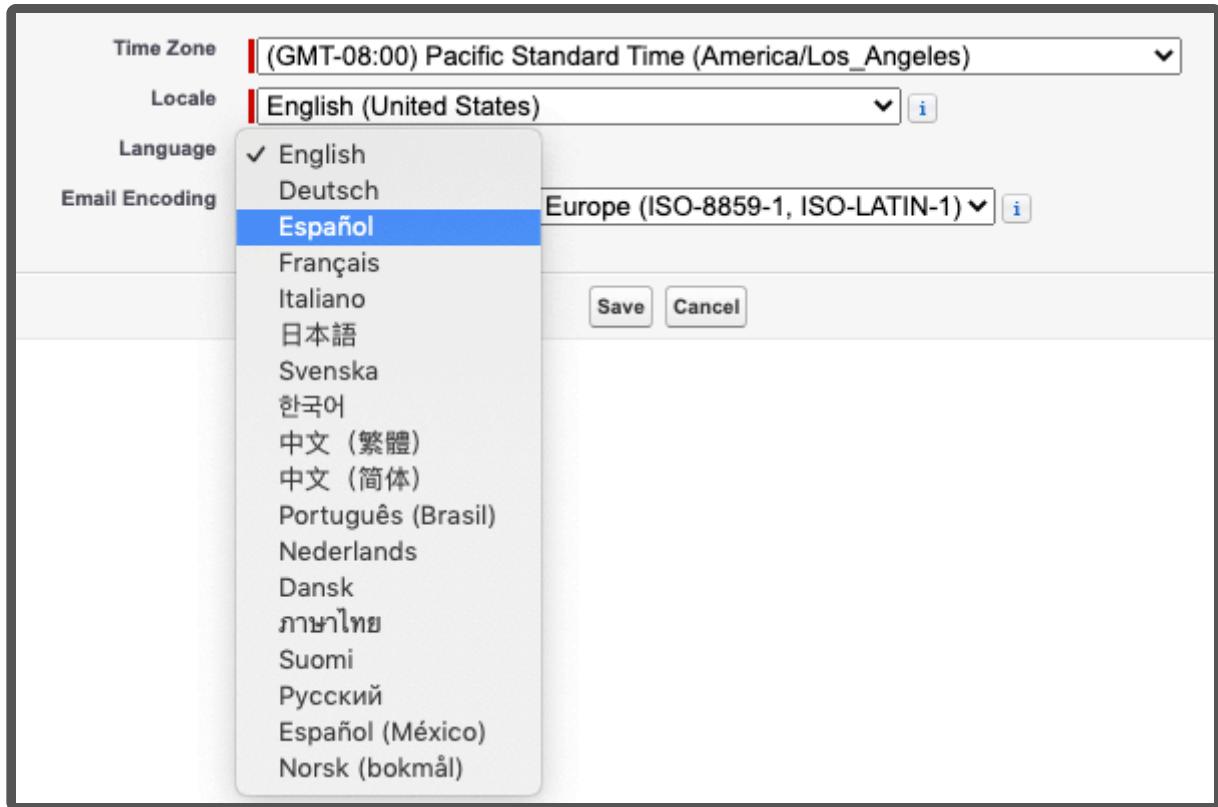
Change the language by selecting the username in the top right corner, then click on "My Settings".



On the setting page on the left panel go to "Personal" and then select "Language & Time Zone".



You can then select your preferred language. Note that CTI adapter only have nine languages built within the package.



Click save and the page will reload. That's it. You can check in other pages to see if it actually applies your change. For example here is a screenshot of CTI Flow Editor in Spanish.



Explorer



Buscar

Buscar por nombre

Categorías

Filtrar por categoría

Etiquetas

Filtrar por etiqueta

Mostrar 100 acciones

[Guardar búsqueda](#)

If-else

Cambie el flujo del script en función del valor de los campos que obtenga o almacene. Se trata de una utilidad "if-else" sencilla para el flujo.

[Parámetros >](#)

Qué llama:

`ac.Utils.Common.decision(..
.)`

[Seleccionar](#)

CoreCast

Cast an input value to a Javascript type, such as Number or String.

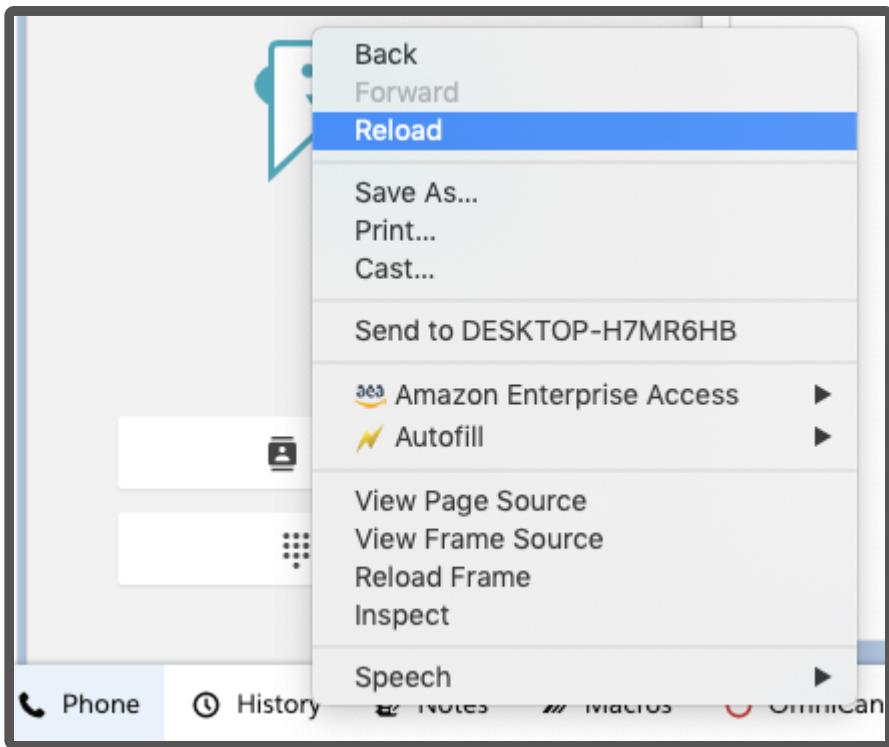
[Parámetros >](#)

[Seleccionar](#)

Solicitud HTTP

Obtener la propiedad

Click on Phone pannel on the bottom to see if CCP has been localized. If not right click on CCP and reload.



Additional Notes

Please note that not all fields can be localized to different languages due to a couple reasons. Here are places that cannot be localized:

- Dashboard. Salesforce dashboards do not support localization.
- Flexipages. This means the page with tabs that you can find in AC CTI Adapter page in lightning.

A screenshot of the AC CTI Adapter feature panel. The top navigation bar has tabs: Attributes (which is underlined in blue), CTI Flows, Presence Sync Rules, and Features. Below the navigation bar, there is a section titled "Attributes (0)" with a small orange icon containing a white telephone handset. The rest of the panel is currently empty.

- Reports. This is a missing functionality in Salesforce.

Set Agent Status on Session End

This feature automatically sets the status of the agent to "Offline" -- or to any status you choose -- when the agent closes all his Salesforce tabs. **Disclaimer:** This feature will popup a window to perform the logout functionality. This window must stay open for the feature to work, but it does not have to be visible (i.e. can be put in the background).

You can configure this feature by heading to the feature panel on your CTI Adapter and clicking new.

 Features (0)

New

Then for "AC Feature Name", enter: SetAgentStatusOnSessionEnd

New AC Feature

Information

* AC Feature Name

SetAgentStatusOnSessionEnd

Value

Active



* CTI Adapter



ACLightningAdapter



You can optionally specify which status the agents should be changed to when they end the session. By default, this is "Offline," but you can configure it using the Status setting of the feature.

* AC Feature Name

SetAgentStatusOnSessionEnd

Value

Status:Away

When turned on, the feature will apply to all agents. If you'd rather have it apply to a small subset, you can configure `IfProfileNameIncludes` setting.

* AC Feature Name

SetAgentStatusOnSessionEnd

Value

Status:Away

IfProfileNameIncludes:On-Call

Now only the agents that have "On-Call" in their Connect routing profile name will be shown as "Offline" when they end their session. This setting can accept multiple, comma-separated profile names, as well.

If you would also like to control this feature based on the current state of the agent, you can add the `IfCurrentAgentState` tag to the feature, and assign a comma separated list of statuses that you would like the feature to execute on.

* AC Feature Name	SetAgentStatusOnSessionEnd
Value	Status:Offline SessionEndTimer:20 IfCurrentAgentState:Available, At Lunch

From this example, only agents who have a current status of "Available" or "At Lunch" will be moved to a state of "Offline" when they end their session.

The example above also utilizes the `SessionEndTimer` feature as well. This delays the state change for the desired amount of time (default of 5 seconds). In the example above it sets the delay to 20 seconds. This feature is useful to account for agent's with slow internet refreshing their page - with 5 seconds, it may change the state of the agent before the refresh loads all of the assets again, while 20 seconds could be enough time for the assets to load, and stop the state change.

You can also have the Status set to `Logout`, which will append the functionality of the logout feature mentioned [here](#) - logging the agent out of the CCP upon session ending. It will not log the user out if a call is ongoing.

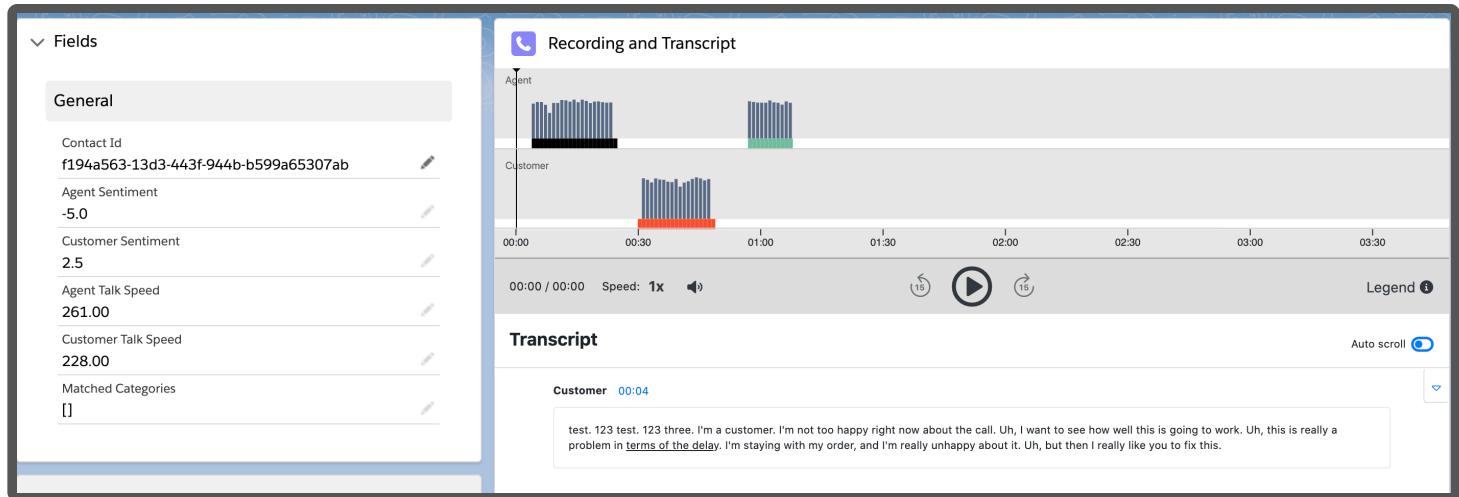
When your agents log back in, they will be shown as "Available" by default. If you'd like to control which status to set your agents, you can configure it with `InitialAgentState` setting.

Note that this feature does not work with Salesforce Pop-Out utilities. This means that it won't be working if CCP is popped out from utility bar. This is because the pop-out window is a different window managed by Salesforce and we are not able to track any session on that window.

Contact Lens

CTI Adapter now gives you access to your post-call Contact Lens data on your Salesforce instance. To configure this feature, you must have installed and configured the AWS Serverless application.

Three or four minutes after the call, a new Contact Channel Analytics record is created with the recording url with only the call recording. In another three minutes, this record is updated with Contact Lens recording, transcript and other metadata.



The new record is also associated automatically with a Case and Contact through their Amazon Connect contact id. This means that you will be able to configure your case record page with a related list that lists all the calls related to a case.

Please note: to have access to the recording, the user must have an active session with Amazon Connect. This can be achieved by either logging in to the CCP softphone, or by logging in to Amazon Connect outside of Salesforce. After the session is established, a page refresh should allow the user to play the audio.

Prerequisites

In order to set up Contact Lens you must first follow the steps detailed in the below sections:

1. [Postcall Contact Lens Import](#)
2. [Set up Contact Channel Analytics](#)

CTI Actions

Customers can now extend their Contact Control Panel (CCP) with customizable buttons called CTI Actions. These buttons can be configured in Salesforce and used to simplify common agent actions. For example, you can add a button that starts and stops recordings, automate case creation, or start a customer refund process. CTI Actions are configured in the CTI Adapter's Actions Admin panel to execute [CTI Flows](#) which are process blocks that enable you to easily design agent workflows within our Salesforce integration.

You can configure a CTI Action in the CCP Element Editor page.

**Actions****Step 1:** Name and Flow**Save**

Quick Save

Delete

Cancel

Step 2: Payload

(optional)

Step 3: Additional Data

(optional)

This section asks you for some required information about your action. It is the only required section you need to fill to create an action.

Action Name
Leave Voicemail

The name agents will see.

CTI Flow
Leave a VoicemailIn this field, you will see all CTI Flows in this account whose source field is [CCP Overlay](#).**Order**
0

Position of the action in the overlay.

Make sure that you have created a CTI Flow and it uses the source "CTI Action." Only these CTI Flows will be displayed in the dropdown field.

You can optionally specify a payload to pass to the CTI Flow. This allows your agents to enter additional data about the customer or information about the call to pass into the CTI Flow. The CCP Element Editor gives you the ability to add input fields into your form. These fields can be accessed in the CTI Flow through `$.payload.fieldKey`.

Actions

Step 1: Name and Flow	
Step 2: Payload	(optional)
Step 3: Additional Data	(optional)

Save Quick Save **Delete** Cancel

In this section, you will build a form that will be displayed to the agents prior to triggering the CTI Flow. The form data will be passed as a payload to the executed flow.

[Overview](#) [Form fields](#)

New field +

This section collects some basic information about the form, such as title and instructions. Both fields are optional.

(optional)

Title

Enter a short title for the form.

(optional)

Instructions

Enter a few lines about how to fill out this form.

[Form fields ▶](#)

Actions

Step 1: Name and Flow	
Step 2: Payload	(optional)
Step 3: Additional Data	(optional)

Save Quick Save **Delete** Cancel

In this section, you will build a form that will be displayed to the agents prior to triggering the CTI Flow. The form data will be passed as a payload to the executed flow.

[Overview](#) [Form fields](#)

New field +

Field Name

Label



This is the name of the field in your payload. It should be a [camelCased](#) word.

The label is a human readable text shown to the agent next to the input field.

Field Type

Text



Order

0

You have the option to select a text input or a dropdown.

Field Required

Cancel

Finish

[◀ Overview](#)

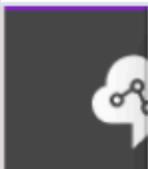
CCP Overlay

The **Actions** panel in the CCP overlay drawer displays the CTI Action buttons where your agents have easy access to them as they are interacting with customers.

The screenshots below are showcasing the CTI Actions and their behavior in the CCP Overlay panel, not the individual CTI Flows shown.



Phone



Attributes

Actions



Send Customer Giftcard



Activate Customer Account

Execute

Transfer to Manager

Execute

Give customer refund

Execute

Open a Case

Execute



Find Cases for Customer

Execute

Create Task and Contact and
Screenpop

Execute

VIP

Execute

Transfer to Manager

Execute

Transfer to Peer

Execute

If a CTI Action requires additional input by the agent, its name will be followed by an arrow and when the agent clicks on this button, it will open the configured form. Otherwise, it will be shown with an "Execute" button next to its name.



Phone



Attributes

Actions



Go back

Customer Gift Card

Please fill in these details about the user.

First name*

John



Last name*

Doe



Telephone

Submit

Example

In this section we demonstrate how to use CTI Actions and how they interact with CTI Flows through an example.

Here we setup a CTI Action and Flow to create a Salesforce Task to callback a customer and pop it. The end goal is to have a Task with the subject *Callback - FirstName - LastName* and the number to callback in the comments section of the Task. If a contact exists for that number, we will also link it in the Task. We use a CTI Action to do this to let the agent enter the customer's first and last name and callback number if it is different from the number used to call in. This action looks like this in the CCP Overlay.



Go back

Customer Callback Information

If the callback number is the different from the number used to dial in enter it in the form, otherwise keep it empty.

First Name*

- is a required property

Last Name*

- is a required property

Callback Number

Submit

To achieve this, we need to setup a CTI Action then a CTI Flow.

First, we setup the CTI Action. To do that we need to have created a CTI Flow with the **CTI Actions** as source. For now we create an empty Flow, which we will build later, just to reference it in the Action.

The first step is to name and link the Action to a Flow.

This screenshot shows the 'Actions' interface. At the top, there are three tabs: 'Step 1: Name and Flow', 'Step 2: Payload' (optional), and 'Step 3: Additional Data' (optional). Below these tabs are four buttons: 'Save' (highlighted in dark grey), 'Quick Save' (light grey), 'Delete' (red border), and 'Cancel'. A note below the tabs states: 'This section asks you for some required information about your action. It is the only required section you need to fill to create an action.' The main form area contains fields for 'Action Name' (set to 'Create Callback Task'), 'The name agents will see.' (set to 'CTI Flow Create Callback Task'), and 'Position of the action in the overlay.' (set to 'Order 0').

The second step is to add hardcoded fields to the payload, if desired. In this example we add part of the Task subject as hardcoded fields to demonstrate the functionality.

This screenshot shows the 'Actions' interface with the 'Step 2: Payload' tab selected. A note says: 'The payload allows you to pass hardcoded values to the CTI Flow. Your payload may include values that are specific to this action and are not already available through a CTI Flow block.' Below this, under the heading 'Payload' (optional), there is a table with two columns: 'Key' and 'Value'. The key 'SubjectPrepend' has the value 'Callback'. A blue 'New key' button is located at the bottom left of the payload table.

Finally, as shown previously, the action is a form, that means it has additional data that the agent can provide. Below are images showing how they are setup for this example.

Actions

Step 1: Name and Flow	(optional)
Step 2: Payload	(optional)
Step 3: Additional Data	(optional)

Save Quick Save **Delete** Cancel

In this section, you will build a form that will be displayed to the agents prior to triggering the CTI Flow. The form data will be passed as a payload to the executed flow.

Overview Form fields **New field +**

This section collects some basic information about the form, such as title and instructions. Both fields are optional.

(optional)

Title
Customer Callback Information

Enter a short title for the form.

(optional)

Instructions

If the callback number is the different from the number used to dial in enter it in the form, otherwise keep it empty.

Enter a few lines about how to fill out this form.

Form fields ▶

Actions

Step 1: Name and Flow	(optional)
Step 2: Payload	(optional)
Step 3: Additional Data	(optional)

Save Quick Save **Delete** Cancel

In this section, you will build a form that will be displayed to the agents prior to triggering the CTI Flow. The form data will be passed as a payload to the executed flow.

Overview Form fields **New field +**

This is a list of fields that will appear in your form. They are shown in the order they will appear.

First Name

Last Name

Callback Number

◀ Overview

Actions

Step 1: Name and Flow	(optional)
Step 2: Payload	(optional)
Step 3: Additional Data	(optional)

Save Quick Save **Delete** Cancel

In this section, you will build a form that will be displayed to the agents prior to triggering the CTI Flow. The form data will be passed as a payload to the executed flow.

Overview Form fields **New field +**

Field Name
FirstName

This is the name of the field in your payload. It should be a [camelCased](#) word.

Label
First Name

The label is a human readable text shown to the agent next to the input field.

Field Type

Text

You have the option to select a text input or a dropdown.

▼

Order

0

Field Required

Cancel **Finish**

◀ Overview

Actions

Step 1: Name and Flow	(optional)
Step 2: Payload	(optional)
Step 3: Additional Data	(optional)

Save
Quick Save
Delete
Cancel

In this section, you will build a form that will be displayed to the agents prior to triggering the CTI Flow. The form data will be passed as a payload to the executed flow.

Overview
Form fields
New field +

Field Name
LastName

Label
Last Name

This is the name of the field in your payload. It should be a **camelCased** word.

Field Type
Text

Order
1

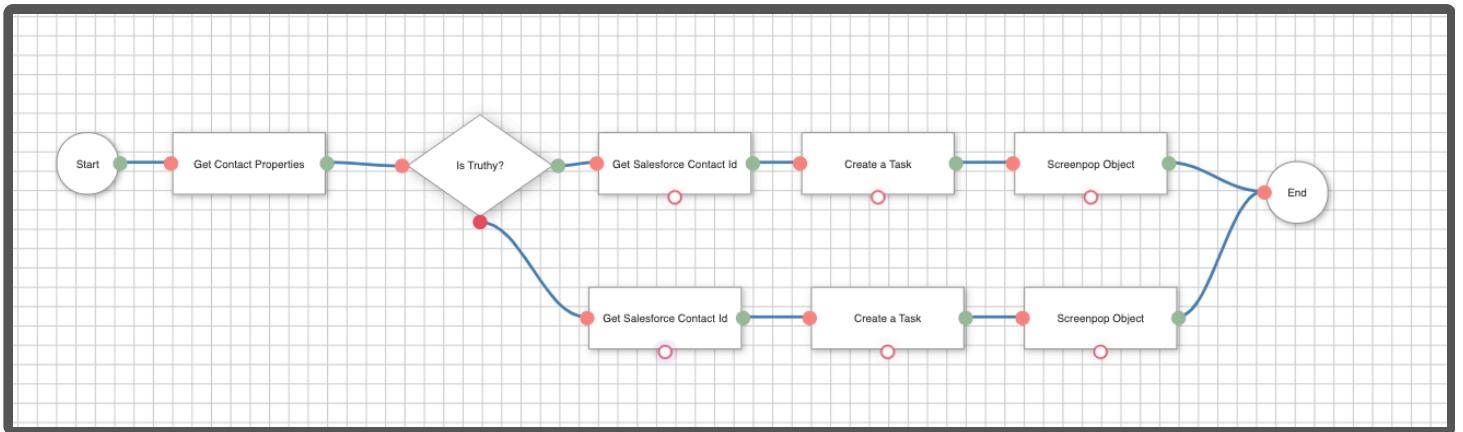
You have the option to select a text input or a dropdown.

Field Required

Cancel
Finish

New field +
Last Name

Then, we setup the CTI Flow. As mentioned above, it's possible to have the callback number different from the number used to call in, or it could be the same. If it's the same, we don't want the agent to enter the number again, in fact we can get that number in the CTI Flow. In the flow we use the **Get Contact Properties** block to get the phone number of the contact. Then using the **Is Truthy?** block, we check if the agent entered a callback number in the form or not. Depending on whether they did or not, we get the Salesforce Contact and create a Task using the correct callback number. In the Flow we reference the CTI Action fields by using `$.payload.fieldKey` for both the hardcoded payload and the fields in the additional data form (Take a look at the **Create a Task** blocks in the flow below).



[Download Flow](#)

To test this action, you can place or accept a call from the CCP, open the overlay, fill in the form then submit it. If everything is setup correctly, a Task should pop up with the desired information.

Receiving Data from CTI Flows

In addition to agents sending data to the CTI Flow, they can also receive data from a CTI Flow.. When a CTI Flow sends some information to the CCP overlay, it will be displayed in the Data panel.



Phone



Send Data to CTI



Attributes

Data



+1 3

Data Sink

foo

bar



Here is how you would configure your CTI Flow to send data back to the CCP overlay.

Send Data to CCP Overlay

ID: uid-9 ⓘ

Arguments

value ⓘ optional

✖

Add a field

```

graph TD
    Start((Start)) -- Blue Curve --> Decision{Is Contact Inbound?}
    Decision -- Red Line --> Green1(( ))
    Decision -- Blue Line --> Action[Send Data to CCP Overlay]
    Action -- Blue Curve --> Start
  
```

Upgrading from an earlier version

If you are upgrading the Salesforce package from an earlier version of CTI Adapter, there are a few additional steps to follow:

1. Go to Setup
2. In "Quick Find," search for "Picklist Value Sets" and click on the result.
3. Select "AC_CtiScriptSource" on "Picklist Value Sets" page.
4. Scroll down to "Values" section
5. Click "New" to add a new value.
6. In the textarea, enter "ctiAction" and save
7. Scroll down to the new field you added, "ctiAction," and click "Edit."
8. Update the label to "CTI Action" and save.

Recording Controls

Recording Controls panel in the CCP Overlay allows your agents to control the recording behavior of the call.



Phone



Attributes

Recording Controls



Start recording

Pause recording



This panel integrates to Amazon Connect [call recording API](#). To use it, make sure to add [Set recording behavior block](#) in your Contact Flow. The controls will be activated during a call.

This can be useful when you don't want to record every call, and give the agent the ability to pause and resume a recording.

Note that once a recording is stopped, it cannot be restarted. After starting a recording, you should use pause/resume button to control it.

This panel is disabled by default. You can enable it by adding `FEATURE_RECORDING_PANEL` feature flag to your CTI Adapter, with the setting `Enabled:true`.

Setup

First, create an IAM user and give it the managed policy `AmazonConnect_FullAccess`.

The screenshot shows the AWS IAM Permissions Policies page. At the top, there are tabs for **Permissions**, **Groups**, **Tags**, and **Security credentials**. The **Permissions** tab is selected. Below the tabs, a section titled **▼ Permissions policies (1 policy applied)** is shown. A blue button labeled **Add permissions** is visible. In the main area, there is a table with one row. The first column contains a dropdown arrow icon. The second column contains the **Policy name** field, which has **AmazonConnect_FullAccess** selected. Below the table, the text **Attached directly** is displayed, followed by a dropdown arrow icon and the **AmazonConnect_FullAccess** label.

Copy the access key and secret of this user (from the "Security credentials" tab.) Next, go to your Salesforce instance Setup section. Search for Named Credentials in the left sidebar, and create a new credential named `AmazonConnectAPI`. (The name and the label should be identical.)

Named Credential Edit: AmazonConnectAPI

Specify the callout endpoint's URL and the authentication settings that are required for

Save Cancel

Label:

Name:

URL:

▼ Authentication

Certificate:

Identity Type:

Authentication Protocol:

AWS Access Key ID:

AWS Secret Access Key:

AWS Region:

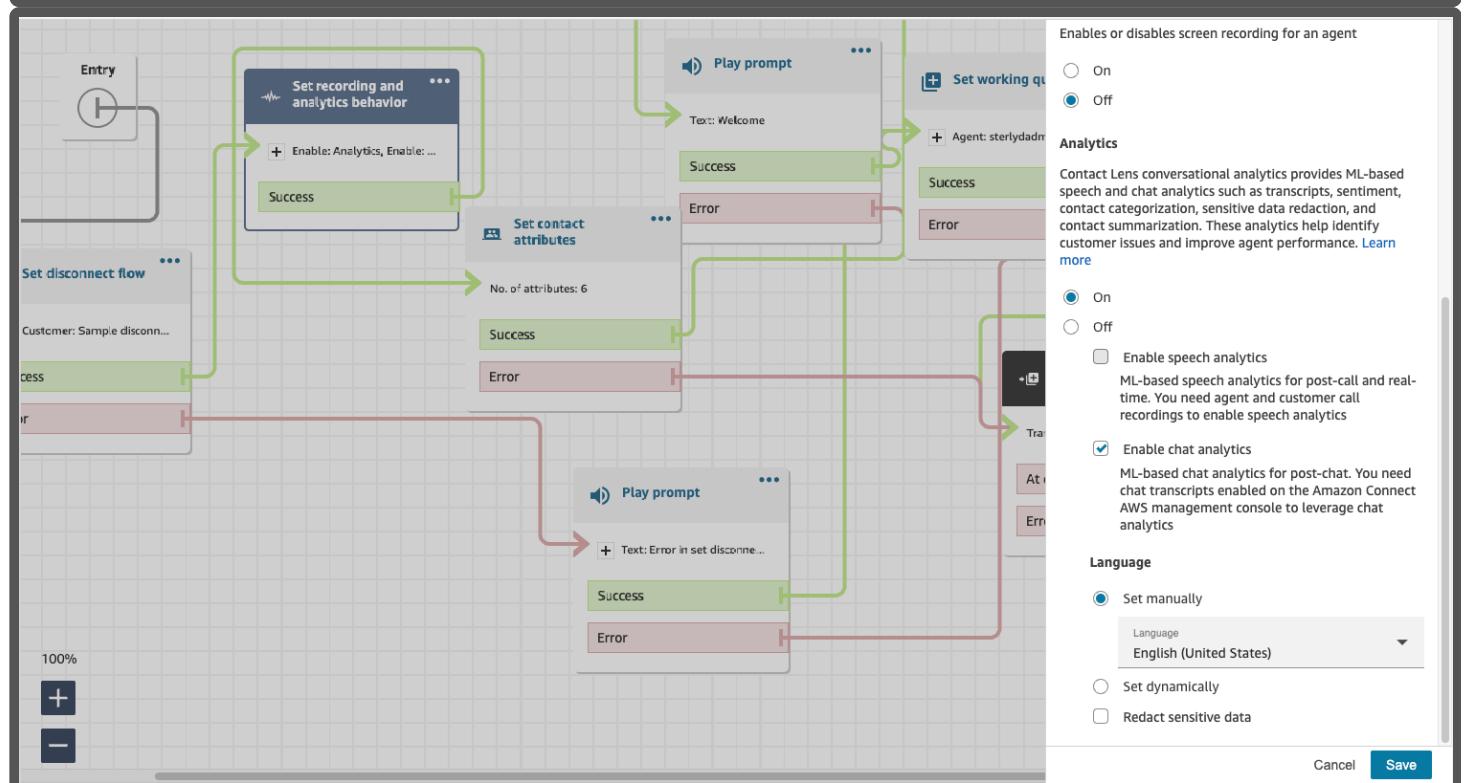
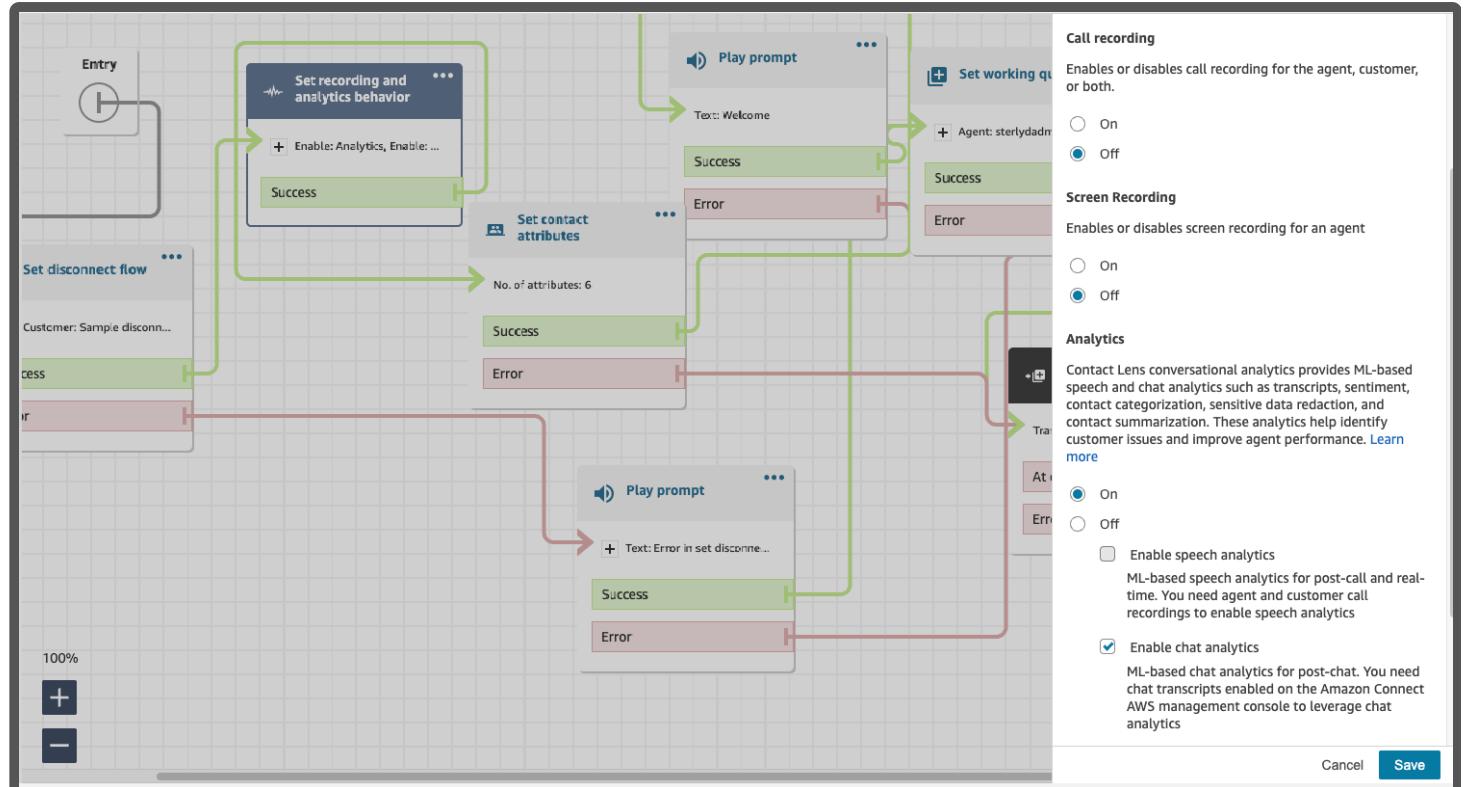
AWS Service:

Fill in `https://connect.{region}.amazonaws.com` as the url. The region in the url needs to match the region for your connect instance / CTI Adapter, so if the CTI Adapter region is `us-west-2`, your url should be: `https://connect.us-west-2.amazonaws.com`

For Identity Type, select "Named Principal" and for "Authentication Protocol" select "AWS Signature Version 4." Then fill in the "AWS Access Key Id" and "AWS Access Secret" fields with your IAM user credentials. And for AWS Region, use the region of your Connect instance. And for the AWS Service, fill in `connect`.

Using Recording Controls with Contact Flows

This feature can sometimes cause issues if an existing contact flow manages call recording, which is common with Contact Lens. If you're using this feature with Contact Channel Analytics or Contact Trace Records enabled, make sure your amazon connect contact flow does not enable call recording by default. If you still wish to generate Contact Channel Analytics / Contact Trace Records, then you should disable call recording in the contact flow block and leave the other settings the same. For reference, here is a valid configuration with Contact Lens and CTR enabled:



Recording Named Credential

Starting with version v5.22, you now have an area in the CTI Adapter to specify the recording Named Credential you wish to use with the adapter. Follow the instructions in the "Setup" section above to create the Named Credentials. Once you have the names for the Named Credentials, add them directly to the Adapters you wish to use.

CTI Adapter Named Credential Location:

The screenshot shows a configuration page for a CTI Adapter named 'ACLightningAdapter'. The 'Recording Named Credential' field is highlighted with a blue box and a tooltip. The tooltip states: 'Optional: This is the Named Credential used for the "Recording Controls" feature. If this is empty, a default value of "AmazonConnectAPI" will be assumed.' Other fields visible include 'Owner' (lilax-starlight), 'Amazon Connect Instance', 'Custom Ringtone', 'Custom Chat Ringtone', 'Softphone Popout Enabled' (checked), 'Medialess' (unchecked), 'Audio Device Settings' (unchecked), 'Call Center Definition Name' (ACLightningAdapter), 'Debug Level' (Off), 'Presence Sync Enabled' (checked), 'Early Get User Media (GUM)' (unchecked), and 'Phone Type Settings' (checked).

Note: In v5.22+, if you have "Recording Controls" enabled and you do not explicitly set a Named Credential on the CTI Adapter, it will assume "AmazonConnectAPI" by default.

Synchronizing Recording State with Contact Attributes

The Connect API does not provide a way for us to check that the recording has already been started when a call is answered. This may result in the UI panel falling out of sync with the actual state of the contact. If you have configured your contacts to be recorded automatically, using the Contact Flow, you must take care to add a contact attribute to indicate that:

Attribute Name: RECORDING_STARTED Attribute Value: true

If you have configured this attribute, then the recording controls will be in sync with the recording state.

Voicemail Drops

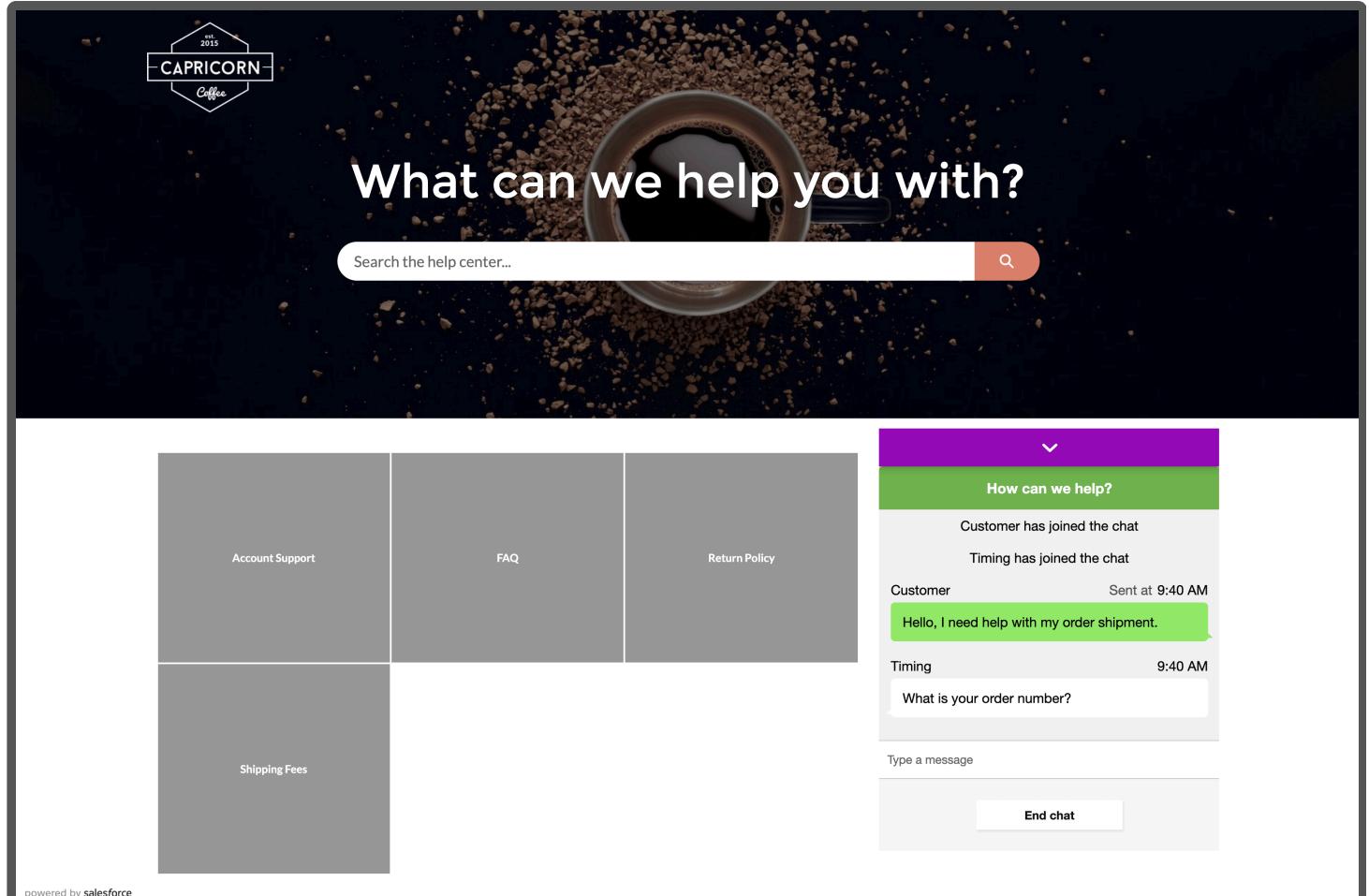
You can find the complete documentation for this feature [in this pdf](#).

Chat Widget Integration

SalesForce Experience Cloud allows you to setup a website for your customers easily, with the included template, you can setup a help center, or a customer service website with just a few clicks.

Amazon Connect CTI Adapter now provides you a chat-widget component, and you can use it in the Experience Cloud Builder App to add the Amazon Connect Chat Widget to any page you want.

The screenshot below shows an example of having the chat widget added to a help center website. Please note that this feature does not support **Build Your Own(LWR)** and **Salesforce Tabs + Visualforce** template.



To start using this feature, you can either follow the steps below to setup an Experience Cloud Site for testing purpose, or you can skip to the next section if you are already familiar with SalesForce Experience Cloud.

Setup Experience Cloud Site

- Go to Setup
- Search for Digital Experience

- Enable Digital Experience

The screenshot shows the Salesforce Setup interface with the following details:

- Search Bar:** digital
- Left Navigation:** Feature Settings > Digital Experiences > Settings
- Message:** Didn't find what you're looking for? Try using Global Search.
- Section:** Experiences
- Text:** Build pixel-perfect websites, portals, communities, and forums with Experience Cloud. [Learn More](#)
- Text:** To get started with digital experiences, you must first enable it and select a domain. If enhanced domains are enabled, your org's My Domain name is the subdomain for any site you create.
- Form:**
 - Enable Digital Experiences:** A checkbox labeled "Enable Digital Experiences" is checked.
 - Select a domain name:** An important note states: "Important: The domain name is used in all of your digital experiences and can't be changed after you save it." Below this, a "Sample Domain Name" is listed as "MyCompany.na162.force.com".
 - Sample Experience URLs:** Three URLs are shown: "MyCompany.na162.force.com/customers", "MyCompany.na162.force.com/developers", and "MyCompany.na162.force.com/partners".
 - Domain Name:** A text input field contains "-developer-edition.na162.force.com".
 - Buttons:** "Check Availability" and "Save".

- Create a new Site by clicking New button

The screenshot shows the Salesforce Setup interface with the following details:

- Section:** Digital Experiences
- Message:** Success! You can now create new Experience Cloud sites.
- Text:** The list shows Experience Cloud sites in your org. Clicking on the URL takes you directly to the site. If you're not a member, the URL isn't linked.
- Text:** Maximum number of sites (including active, inactive, and preview): 100
- Table:**

All Sites	New
No Sites	

- Choose Help center template to create a new site

[← Back to Setup](#)

Choose the Experience You Love

BROWSE BY:

All Sales Service Commerce Installed

Build Your Own (LWR)
by Salesforce

Unparalleled Performance • Standards-Based Customization •

Develop blazing fast digital experiences, such as websites, microsites, and portals, using the Lightning...

B2C Commerce
by Salesforce

Live search • Product filtering • Einstein Product Recommendations •

Create a responsive ecommerce store that provides easy customization of store layout and template, configure...

Help Center
by Salesforce

Self-Service • Curated Knowledge • Case Deflection • Guest Case Creation

Give your customers the answers they're looking for. Customers can search for and read articles and contact...

Customer Account Portal
by Salesforce

Customer Service
by Salesforce

Build Your Own
by Salesforce

- Go to Builder of the new site

See your favorite metrics right here by getting the latest AppExchange package.

My Workspaces

Builder

Build, brand, and customize your site's pages.

Moderation

Monitor posts and comments, create rules.

Content Management

Organize, manage, and build collections for your Experience Cloud site.

Gamification

Keep your members engaged with recognition badges.

Dashboards

Examine the health of your site with reports and dashboards and engage with members.

Administration

Configure settings and properties for your experience.

Guided Setup

Configure features and integrations with step-by-step instructions.

- This will be the place to setup chat widget feature in the following sections. You can get yourself familiar with this Builder before moving to the next section.

Setup Chat Widget in Amazon Connect

- Follow instructions [here](#) to setup your Chat Widget and copy the script to a text editor.
- Example of Script:

```
<script type="text/javascript">
  (function(w, d, x, id){
    s=d.createElement('script');
    s.src='https://dg9yx063wihht.cloudfront.net/amazon-connect-chat-
interface-client.js';
    s.async=1;
    s.id=id;
    d.getElementsByTagName('head')[0].appendChild(s);
    w[x] = w[x] || function() { (w[x].ac = w[x].ac || []).push(arguments)
  };
  })(window, document, 'amazon_connect', '5338d219-92c7-427e-8b10-
26a8f4dfb3d1');
  amazon_connect('styles', { openChat: { color: 'white', backgroundColor:
'#826359' }, closeChat: { color: 'white', backgroundColor: '#940eb9' } });
  amazon_connect('snippetId',
'QVFJREFIaUpTVGJkNWhNc0Q1wHpHYnFQTkJyYXN0.....=');
  amazon_connect('supportedMessagingContentTypes', [ 'text/plain',
'text/markdown' ]);
</script>
```

- Example Call back function for JWT

```
amazon_connect('authenticate', function(callback) {
  window.fetch('https://www.yourdomain.com/yourAuthEndpoint').then(res => {
    res.json().then(data => {
      callback(data.data);
    });
  });
});
```

Create Required Visualforce Pages

- Navigate to the Salesforce Setup by clicking on the gear icon in the top-right corner of the page.
- In the Setup menu, search for "Visualforce Pages" in the quick find box and click on that.
- On the "Visualforce Pages" page, click on the "New" button.
- According to Security selected above in Amazon Connect Chat Widget website:
 - If Enabled: Provide name like "AC_ChatWidgetWithJWT" in the "Label" field & "Name" field for your Visualforce page.
 - If Disabled: Provide name like "AC_ChatWidget" in the "Label" field & "Name" field for your Visualforce page.
 - *Note: Going forward in documentation, Use the same name which you mention here in place of "AC_ChatWidgetWithJWT" or "AC_ChatWidget".*
- Check the box front of "Available for Lightning Experience, Experience Builder sites, and the mobile app" field.
- Copy the below snippet in text editor and replace comments with mentioned script copied from [here](/amazon-connect-salesforce-cti/docs/classic/cti-adapter/12-chat-widget-integration#Setup Chat Widget in Amazon Connect).
 - For "AC_ChatWidgetWithJWT" Visual force page:

```
<apex:page id="AC_ChatWidgetWithJWT" showHeader="false" sideBar="false" docType="html-5.0">
  <html xmlns="http://www.w3.org/2000/svg"
        xmlns:xlink="http://www.w3.org/1999/xlink" lang="en">

    <head>
        <apex:slds />
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge" />
        <meta name="viewport" content="width=device-width, initial-scale=1" />
        <script type="text/javascript">

            <!-- Add Chat widget script here -->
            <!-- Add Call back function for JWT here -->
        </script>
    </head>
    </html>
</apex:page>
```

Example:

```
<apex:page id="AC_ChatWidgetWithJWT" showHeader="false" sideBar="false" docType="html-5.0">
  <html xmlns="http://www.w3.org/2000/svg"
```

```

xmlns:xlink="http://www.w3.org/1999/xlink" lang="en">
<head>
    <apex:slds />
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <script type="text/javascript">
        <!-- Add Chat widget script here -->
        (function(w, d, x, id){
            s=d.createElement('script');
            s.src='https://dg9yx063wiiht.cloudfront.net/amazon-connect-chat-interface-client.js';
            s.async=1;
            s.id=id;
            d.getElementsByTagName('head')[0].appendChild(s);
            w[x] = w[x] || function() { (w[x].ac = w[x].ac || []).push(arguments) };
        })(window, document, 'amazon_connect', '5338d219-92c7-427e-8b10-26a8f4dfb3d1');
        amazon_connect('styles', { openChat: { color: 'white', backgroundColor: '#826359' }, closeChat: { color: 'white', backgroundColor: '#940eb9' } });
        amazon_connect('snippetId', 'QVFJREFIaUpTVGJkNWhNc0Q1WHpHYnFQTkJyYXN0.....=');
        amazon_connect('supportedMessagingContentTypes', [ 'text/plain', 'text/markdown' ]);
        <!-- Add Call back function for JWT here -->
        amazon_connect('authenticate', function(callback) {
            window.fetch('https://www.yourdomain.com/yourAuthEndpoint').then(res => {
                res.json().then(data => {
                    callback(data.data);
                });
            });
        });
    </script>
</head>
</html>
</apex:page>

```

- For "AC_ChatWidget" Visual force page:

```

<apex:page id="AC_ChatWidget" showHeader="false" sideBar="false" docType="html-5.0">
    <html xmlns="http://www.w3.org/2000/svg"
        xmlns:xlink="http://www.w3.org/1999/xlink" lang="en">

```

```

<head>
    <apex:slds />
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <script type="text/javascript">
        <!-- Add Chat widget script here -->
    </script>
</head>
</html>
</apex:page>

```

Example:

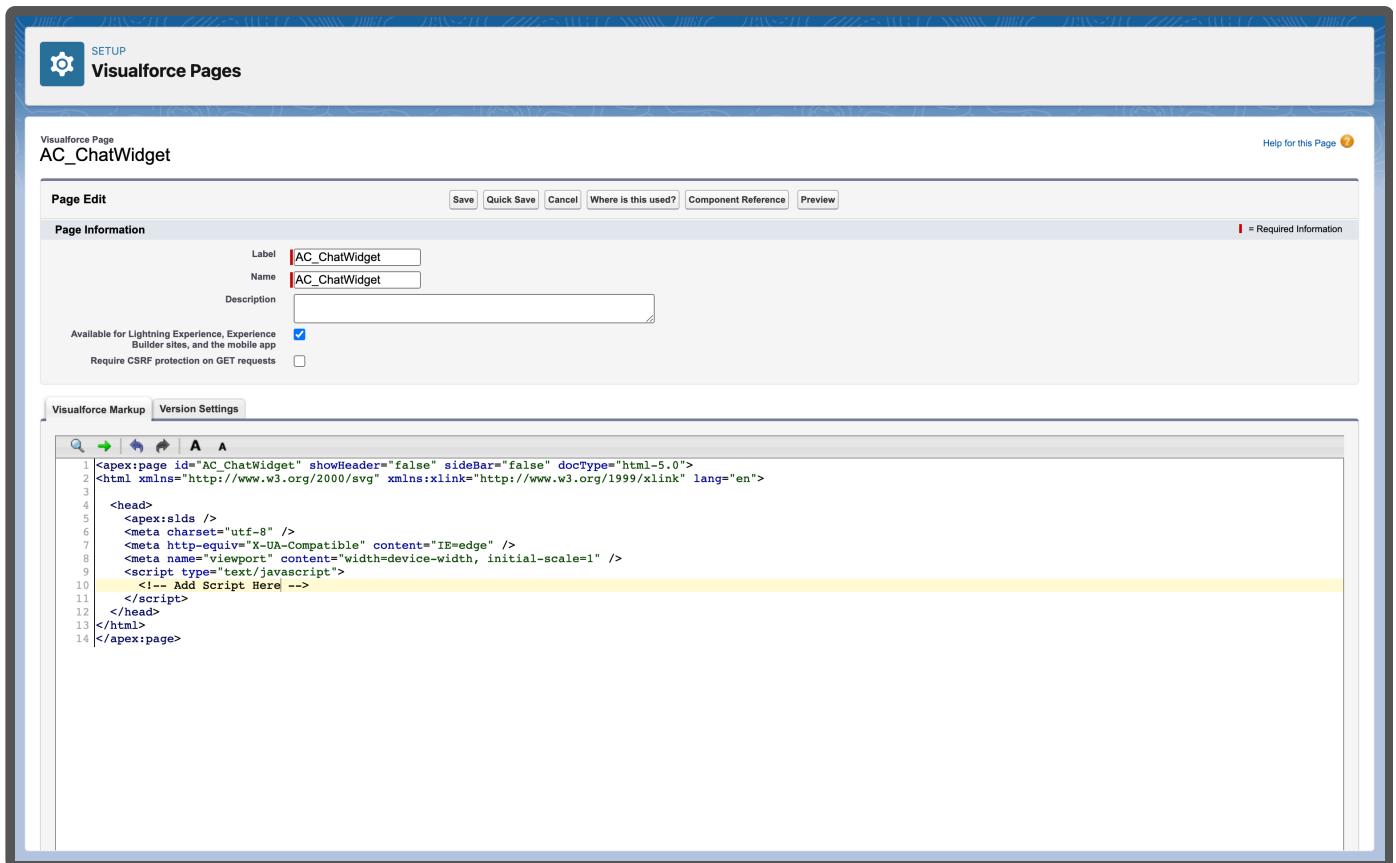
```

<apex:page id="AC_ChatWidget" showHeader="false" sideBar="false"
docType="html-5.0">
    <html xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink" lang="en">
        <head>
            <apex:slds />
            <meta charset="utf-8" />
            <meta http-equiv="X-UA-Compatible" content="IE=edge" />
            <meta name="viewport" content="width=device-width, initial-scale=1" />
            <script type="text/javascript">
                <!-- Add Chat widget script here -->
                (function(w, d, x, id){
                    s=d.createElement('script');
                    s.src='https://dg9yx063wiiht.cloudfront.net/amazon-connect-chat-
interface-client.js';
                    s.async=1;
                    s.id=id;
                    d.getElementsByTagName('head')[0].appendChild(s);
                    w[x] = w[x] || function() { (w[x].ac = w[x].ac || [])
                        .push(arguments) };
                })(window, document, 'amazon_connect', '5338d219-92c7-427e-8b10-
26a8f4dfb3d1');
                amazon_connect('styles', { openChat: { color: 'white',
backgroundColor: '#826359' }, closeChat: { color: 'white', backgroundColor:
'#940eb9' } });
                amazon_connect('snippetId',
'QVFJREFIaUpTVGJkNWhNc0Q1WHpHYnFQTkJyYXN0.....=');
                amazon_connect('supportedMessagingContentTypes', [ 'text/plain',
'text/markdown' ]);
            </script>
        </head>

```

```
</html>  
</apex:page>
```

- Final page should look like below image. Click on Save button.



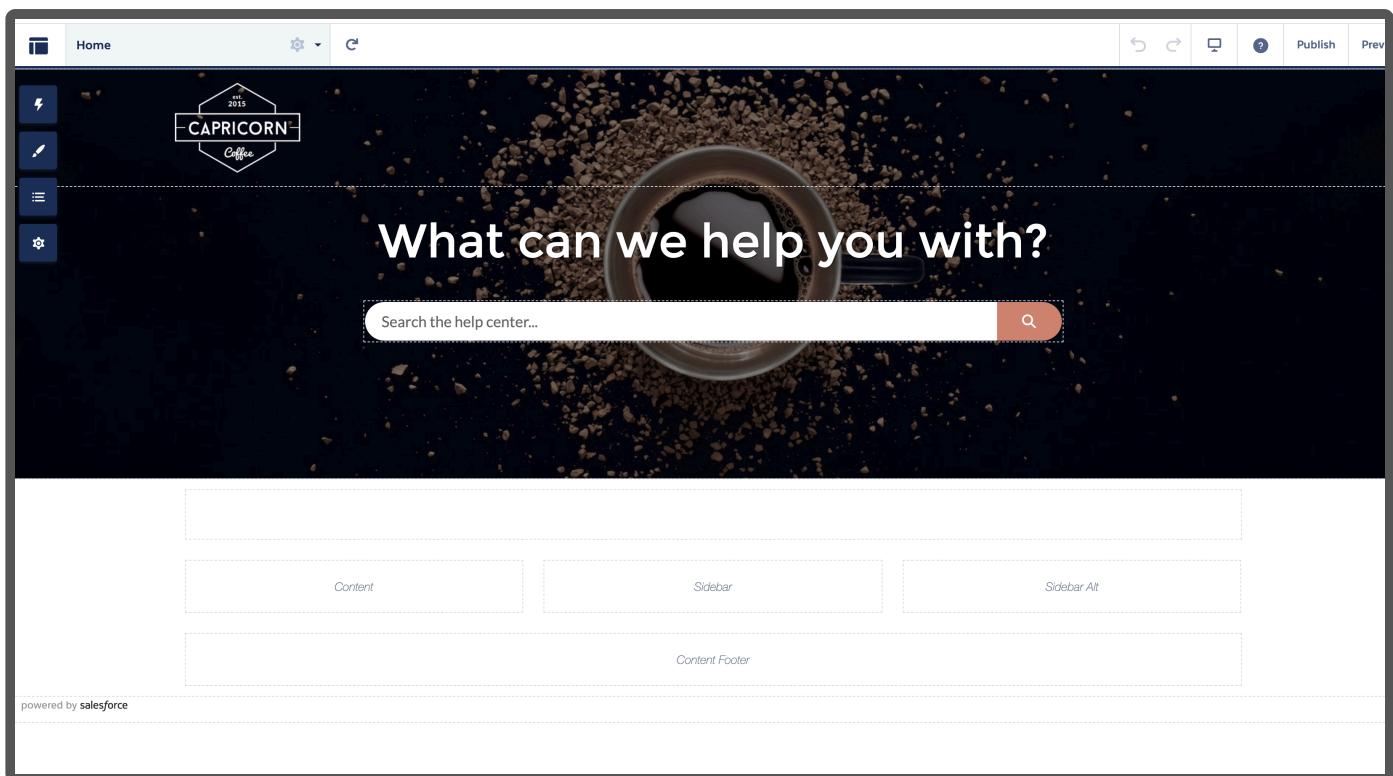
Setup Chat Widget for your Experience Cloud Sites

- Option 1: Setting up using out-of-box VisualForce page. Choose this if you need the chat widget only on one specific page.
- Option 2: Setting up using Lightning Component based on VisualForce page. Choose this if you need the chat widget only on one specific page but you don't have the license for the VisualForce page component in the experience cloud builder. It is a workaround for Option1.
- Option 3: Setting up using custom header. Choose this if you want the chat widget exists across all pages.

Option 1: Setting up using VisualForce page.

- Go to Setup
- Go to VisualForce page
- Select AC_ChatWidget
- Click Preview

- You should see a chat icon on the right bottom corner. If not, check browser console for error messages
- Copy the AC_ChatWidget visualforce page URL.
- Go to your Experience Cloud Builder



- Open Components



Home



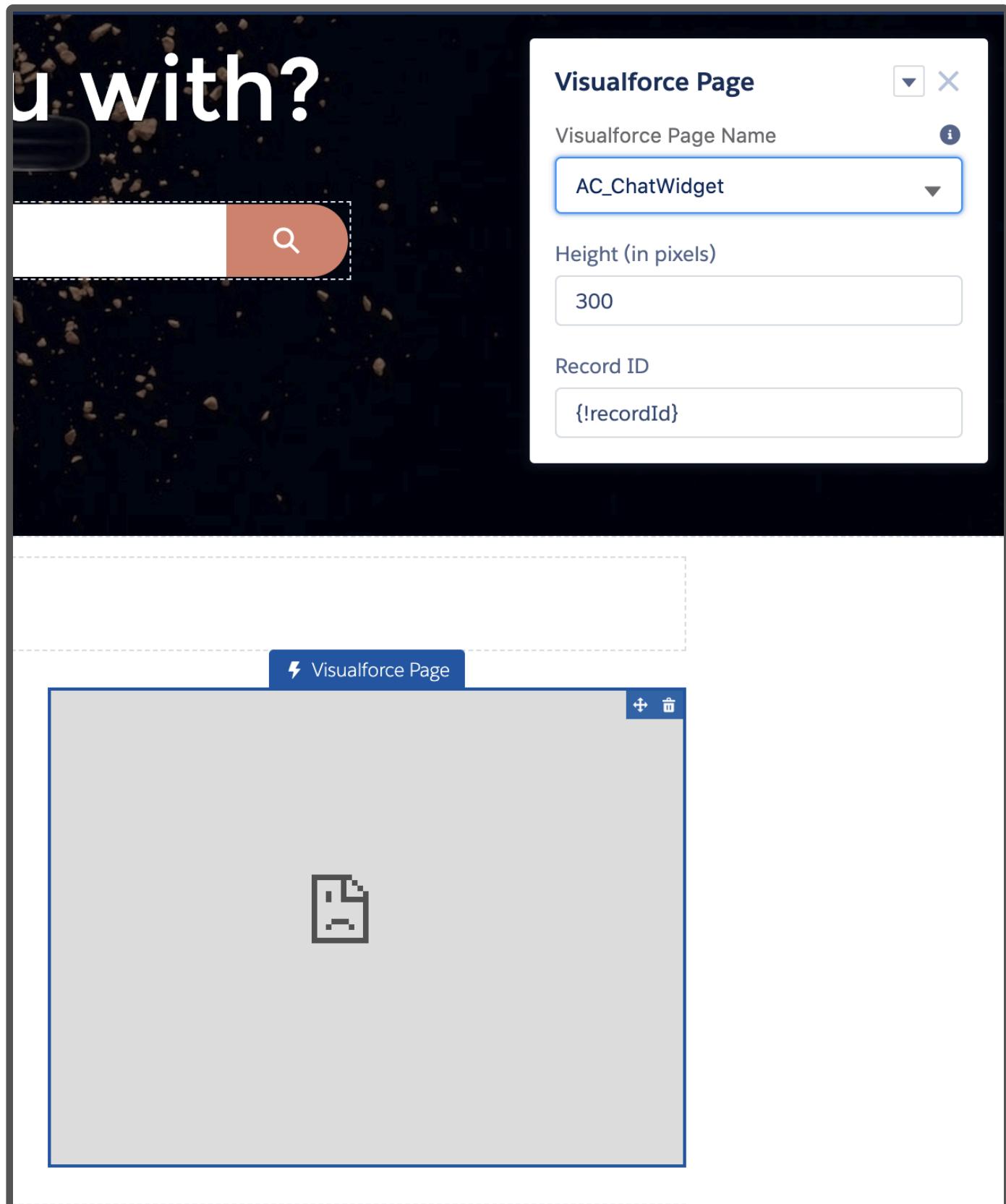
Components



▼ CONTENT (12)

-  CMS Collection
-  CMS Connect (HTML)
-  CMS Connect (JSON)
-  CMS Single Item
-  Headline
-  HTML Editor
-  Language Selector
-  Recommendations Carousel
-  Rich Content Editor
-  Tabs
-  Tile Menu
-  Visualforce Page

- Drag and drop Visualforce Page to your page. If you didn't enable chat widget security, you need to change the Visualforce Page Name to AC_ChatWidget. If you enabled security for ChatWidget, change it to AC_ChatWidgetWithJWT



- Go to Settings→General→Guest User Profile and click in to the Guest User Profile

Guest User Profile

Configure access for guest or unauthenticated users. [Learn More](#)
[dev3test Profile](#)

- Inside Guest user profile, go to Enabled Visualforce Page Access
- Add AC_ChatWidget(or AC_ChatWidgetWithJWT if you have enabled security for chat widget)

Enable Visualforce Page Access

Select the Visualforce pages that you want to make accessible at this Salesforce site.

Available Visualforce Pages	Enabled Visualforce Pages
AnswersHome ChangePassword IdeasHome MyProfilePage SiteTemplate StdExceptionTemplate Unauthorized amazonconnect.ACSFCCP_CallLogging_View amazonconnect.ACSFCCP_CallRecordingCase amazonconnect.ACSFCCP_CallRecordingTask amazonconnect.ACSFCCP_CallTask amazonconnect.ACSFCCP_ObjectType amazonconnect.ACSFCCP_PostCallUpdateTask amazonconnect.AC_AgentStatusSessionEnd	AC_ChatWidget BandwidthExceeded CommunitiesLanding CommunitiesLogin CommunitiesSelfReg CommunitiesSelfRegConfirm CommunitiesTemplate Exception FileNotFoundException ForgotPassword ForgotPasswordConfirm InMaintenance MicrobatchSelfReg SiteLogin

Available Visualforce Pages Enabled Visualforce Pages

Add Remove

- Click Save

- Click Publish button on the top right to publish the website

The screenshot shows the Visualforce page settings interface. On the left, there's a sidebar with icons for Home, Settings, Themes, Languages, Navigation, Mobile Publisher, and SEO. The 'Settings' icon is highlighted. The main content area has a title 'General' and a sub-section 'Site Details'. It includes links for 'Template' and 'Help Center'. Under 'Public Access', there's a checkbox labeled 'Public can access the site' which is checked. In the top right corner, there's a blue 'Publish' button.

- Copy the published website URL in Settings→Published Status
- Go back to Amazon Connect Chat Widget website, add following url to the allow-list Domains:
 - The AC_ChatWidget visualforce page URL, remove everything after .com
 - The published website URL to chat widget allow-list origin, remove everything after .com
- Go to Setup→Sharing Settings. Search for AC CTI Adapter Sharing Rules. Create a new Rule for Guest user so that they have the object access. Make sure in Step2 the Rule Type is Guest user access, the Steps 3 you put a proper criteria, for testing purpose you can put CTI Adapter Name not equal to 1. In Step 4 Share with the Guest user profile of the community website you are

working on, and change the Access level to Read Only

SETUP

Sharing Settings

Setup Help for this Page ?

AC CTI Adapter Sharing Rule

Use sharing rules to make automatic exceptions to your organization-wide sharing settings for defined sets of users.

Note: "Roles and subordinates" includes all users in a role, and the roles below that role. This includes portal roles that may give access to users outside the organization.

You can use sharing rules only to grant wider access to data, not to restrict access.

Step 1: Rule Name ■ = Required Information

Label	<input type="text" value="test"/>
Rule Name	<input type="text" value="test"/> i
Description	<input type="text"/>

Step 2: Select your rule type

Rule Type Based on record owner Based on criteria Guest user access, based on criteria

Step 3: Select which records to be shared

This sharing rule grants access to guest users without login credentials. By modifying the default settings in accordance with these criteria, you're allowing immediate and unlimited access to all records matching these criteria to anyone accessing the site, even without logging in. To secure your site and its data from guest users, consider all the use cases and implications, and implement security controls that you think are appropriate for the sensitivity of your data. Salesforce isn't responsible for any exposure of your data to guest users related to this change from default settings.

Criteria	Field	Operator	Value	
	--None--	--None--		AND
	--None--	--None--		AND
	--None--	--None--		AND
	--None--	--None--		AND
	--None--	--None--		

[Add Filter Logic...](#)

Additional Options Include records owned by high-volume users [i](#)

Step 4: Select the users to share with

Share with

Step 5: Select the level of access for the users

Access Level

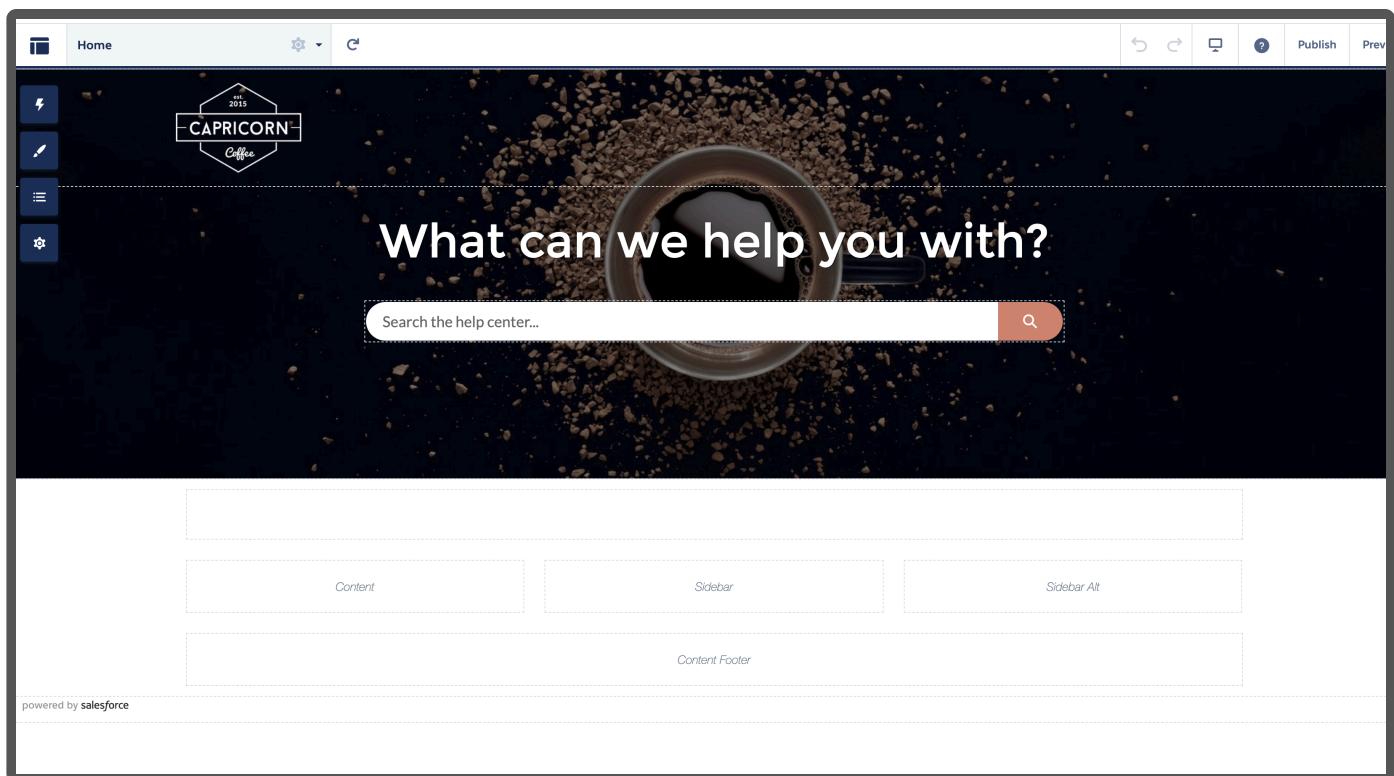
[Save](#) [Cancel](#)

Verify the change: Open your published website in a incognito window, you should be able to use chat widget to chat as a customer and chat to your agent without login Note: If you want to setup chat widget for authorized user group only, you could change the settings to the guest profile to the authorized user profile.

Option 2: Setting up using out-of-box Lightning Component.

- Go to Setup
- Go to VisualForce page
- Select AC_ChatWidget
- Click Preview

- You should see a chat icon on the right bottom corner. If not, check browser console for error messages
- Copy the AC_ChatWidget visualforce page URL.
- Go to your Experience Cloud Builder



- Open Components



Home



Components



▼ CONTENT (12)

-  CMS Collection
-  CMS Connect (HTML)
-  CMS Connect (JSON)
-  CMS Single Item
-  Headline
-  HTML Editor
-  Language Selector
-  Recommendations Carousel
-  Rich Content Editor
-  Tabs
-  Tile Menu
-  Visualforce Page

- Drag and drop iFrame Component to your page



Components

Search...



Record Detail



Related Record List

▼ SALES (1)



Campaign Marketplace

▼ SUPPORT (6)



Case Deflection



Channel Menu



Contact Request Button & F...



Contact Support Button



Contact Support Form



Embedded Service Appoint...

▼ TOPICS (3)



Featured Topics



Topic Catalog



Trending Topics

▼ CUSTOM COMPONENTS (1)



Some components in this section are blocked due to the site's security level setting. [More Details](#)



iFrame Component

Get more on the AppExchange

- Change Chat Widget URL to <your-website-domain>/AC_ChatWidget if you did not enable the security for the chat widget. If you have enabled security, change it to <your-website-domain>/AC_ChatWidgetWithJWT
 - You will have the website domain once it is published. The URL is in Settings→General→Published Status, and the part from https to .com is your website domain. If you haven't published it yet, you can update it once it is published and re-publish the website.
 - If you have site name, you need to append /<site-name> after your domain name. For example if the published website is demo-developer-edition.na111.force.com/testing/s/, your Chat Widget URL should be:
 - If security disabled --> demo-developer-edition.na111.force.com/testing/AC_ChatWidget
 - If security enabled --> demo-developer-edition.na111.force.com/testing/AC_ChatWidgetWithJWT
- Go to Settings→General→Guest User Profile and click in to the Guest User Profile

Guest User Profile

Configure access for guest or unauthenticated users. [Learn More](#)
[dev3test Profile](#)

- Inside Guest user profile, go to Enabled Visualforce Page Access

- Add AC_ChatWidget(or AC_ChatWidgetWithJWT if you have enabled security for chat widget)

Enable Visualforce Page Access

Select the Visualforce pages that you want to make accessible at this Salesforce site.

Available Visualforce Pages	Enabled Visualforce Pages
AnswersHome	AC_ChatWidget
ChangePassword	BandwidthExceeded
IdeasHome	CommunitiesLanding
MyProfilePage	CommunitiesLogin
SiteTemplate	CommunitiesSelfReg
StdExceptionTemplate	CommunitiesSelfRegConfirm
Unauthorized	CommunitiesTemplate
amazonconnect.ACSFCCP_CallLogging_View	Exception
amazonconnect.ACSFCCP_CallRecordingCase	FileNotFoundException
amazonconnect.ACSFCCP_CallRecordingTask	ForgotPassword
amazonconnect.ACSFCCP_CallTask	ForgotPasswordConfirm
amazonconnect.ACSFCCP_ObjectType	InMaintenance
amazonconnect.ACSFCCP_PostCallUpdateTask	MicrobatchSelfReg
amazonconnect.AC_AgentStatusSessionEnd	SiteLogin

- Click Save
- Click Publish button on the top right to publish the website

- Copy the published website URL in Settings→Published Status
- Go back to Amazon Connect Chat Widget website, add following url to the allow-list Domains:

- * The AC_ChatWidget visualforce page URL, remove everything after .com
- * The published website URL to chat widget allow-list origin, remove everything after .com

Verify the change: Open your published website in a incognito window, you should be able to use chat widget to chat as a customer and chat to your agent without login

MultiChat Management

The Amazon Connect CTI Adapter enables Agents concurrently managing multiple Chat contacts efficiently. In the process of handling multiple Chat contacts, Agents need to switch between these chat contacts. With help of the MultiChat Management feature, agents will be able to easily identify the contact while switching.

The Amazon Connect CTI Adapter provides a CTI Flow Event called "onViewContact" specifically designed for the "Amazon Connect Chat Contact" CTI Flow Source. It is available in versions v5.22+. With this event, when agents navigate between multiple chat contacts, the associated CTI Flow can be triggered. For example, a CTI Flow attached to Source : "Amazon Connect Chat Contact" | Event : "onViewContact" can be enabled to execute a ScreenPop action, revealing a related Salesforce object linked to the active Chat contact. Consequently, as agents switch between Chat contacts, the respective object for the ongoing chat will automatically open in the background within the Salesforce window. This functionality serves as a valuable identifier for the currently active Chat contact, enhancing the agent's workflow and efficiency.

Recommendations

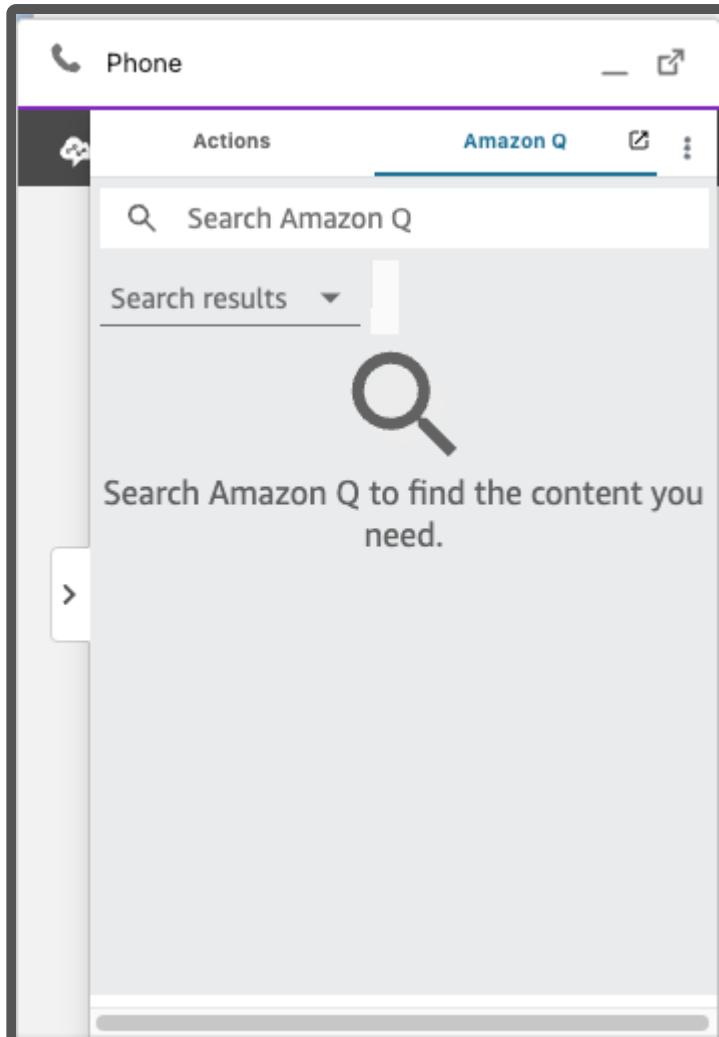
- It's essential to acknowledge that the "onViewContact" event can be triggered multiple times during the lifecycle of a single Chat contact.
- The advisable practice is to utilize only ScreenPop CTI Flow blocks within the CTI Flow associated with the "onViewContact" event.
 - Examples of CTI Flow blocks include: Screenpop Object, Screenpop Object Home, Screenpop Search, Search And Screenpop
- Additionally, it is best to avoid using any CTI Flow block that creates Salesforce objects, as it can result in the creation of multiple objects for a single contact during its lifecycle.
- The CTI Flow connected to the "onViewContact" CTI Flow Event is only activated after the Chat contact is in "Connected" Contact state. As a result, it is preferable to create and screen-pop the Salesforce object during the initial connection phase using the "onConnecting" or "onConnected" CTI Flow events. Subsequently, leverage the "onViewContact" CTI Flow Event to screen-pop Salesforce objects when switching between multiple connected Chat contacts. Please find the example attached in below Section.

Example Use

- Log into your Salesforce instance and open the relevant AC CTI Adapter.
- Configure CTI Flow:
 - Source: "Amazon Connect Chat Contact" | Event: "onConnected" - SalesforceContactCreation.json
 - Source: "Amazon Connect Chat Contact" | Event: "onViewContact" -ScreenPopContact-MultiChat.json
- Agent Experience:
 - Agents receives a new Chat contact.
 - Agent clicks on "Accept Chat" and contact is in "Connected" State.
 - CTI Flow attached to "onConnected" CTI Flow events is triggered and it creates Salesforce contact record with contactId as name and ScreenPop it.
 - Agent switches from one Chat contact to another, CTI Flow attached to "onViewContact" is triggered and it screenPops the relevant Salesforce Contact record of that visible Chat Contact.

Amazon Q Integration

The Amazon Connect CTI Adapter allows for integration with Amazon Connect Amazon Q. We still support reference to Amazon Q's old name "Wisdom" for now, but we will not support it in CTI Adapter version 5.23 and onwards.



The integration between Amazon Q and the CTI Adapter first requires that Amazon Q is set up in the Amazon Connect instance that the CTI Adapter is integrated with. See [here](#) for full instructions.

Before proceeding with the below, please ensure that Amazon articles are properly showing up in your instance with Amazon Q for the specific user you are testing.

Amazon Connect Amazon Q Permission Sets:

Salesforce users accessing Amazon Q in Salesforce must belong to either the *AC_AmazonQ* permission set, or the *AC_Administrator* permission set.

1. In *setup*, search for and select *permission sets*.
2. Select either the *AC_AmazonQ* or the *AC_Administrator* permission set
3. Select *Manage Assignments*, and add all relevant users to the permission set of choice.

Setting up Amazon Connect Amazon Q in the CCP Overlay:

1. Navigate to your CTI Adapter
2. Scroll down to the Features section and create a new feature

Attributes CTI Flows Presence Sync Rules **Features**

Features (0) **New**

3. Create a new feature with the following values:

- AC Feature Name - FEATURE_AMAZONQ_PANEL
- Value - Enabled: true

**AC Feature
FEATURE_AMAZONQ_PANEL**

AC Feature Name
FEATURE_AMAZONQ_PANEL

Value
Enabled:true

Active

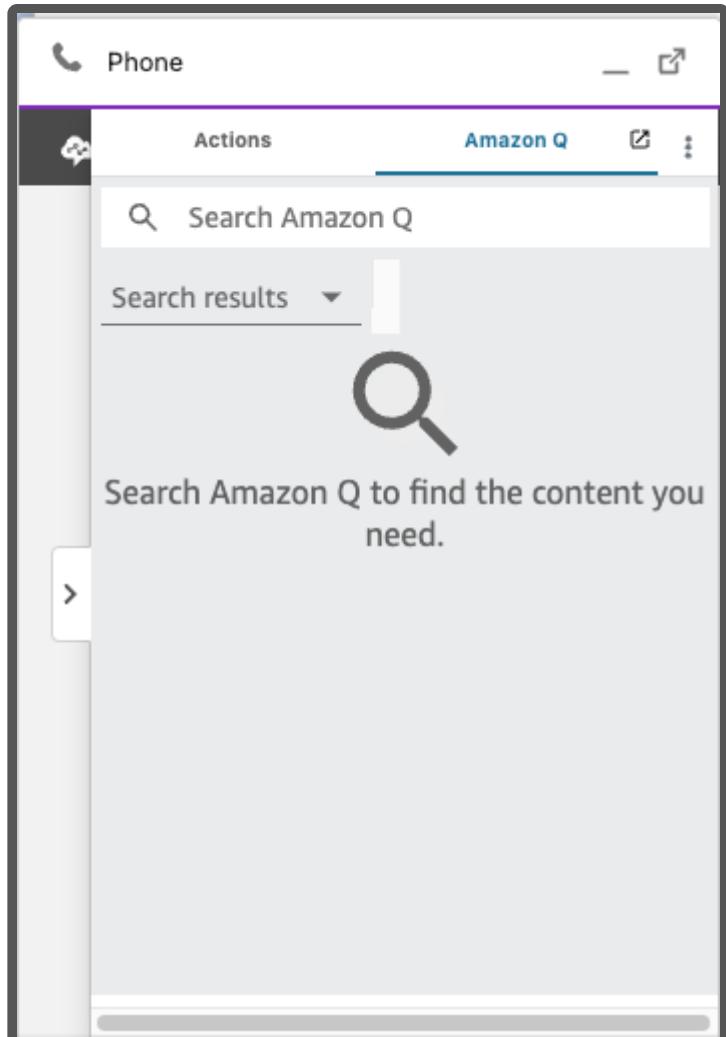
4. In addition, you can also include the `IgnorePermissionSet` setting to the value of the feature on a new line. This setting will show AmazonQ if it is enabled regardless of whether the logged in user belongs to the `AC_AmazonQ` or the `AC_Administrator` permission set. This setting is required if the logged in user has the `View Setup and Configuration` profile setting set to false.

- `IgnorePermissionSet: true`

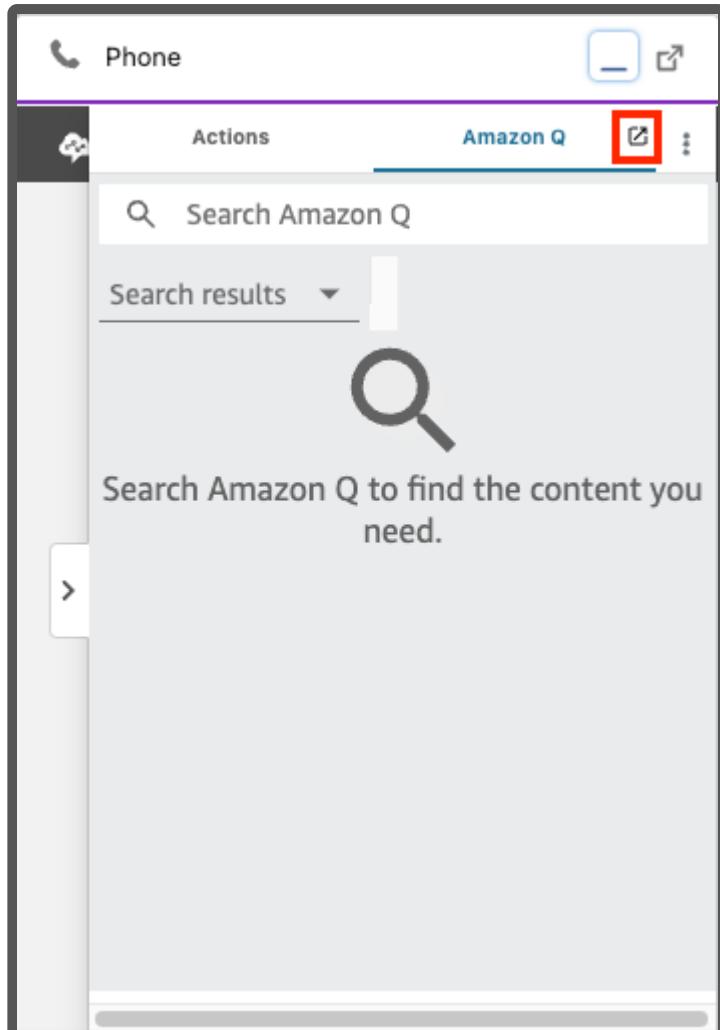
AC Feature Name
FEATURE_AMAZONQ_PANEL

Value
Enabled:true
IgnorePermissionSet: true

5. Open the ccp, observe that there is a tab with Amazon Q in the CCP Overlay.



Amazon Q can be popped out into a new window by pressing pop out button.



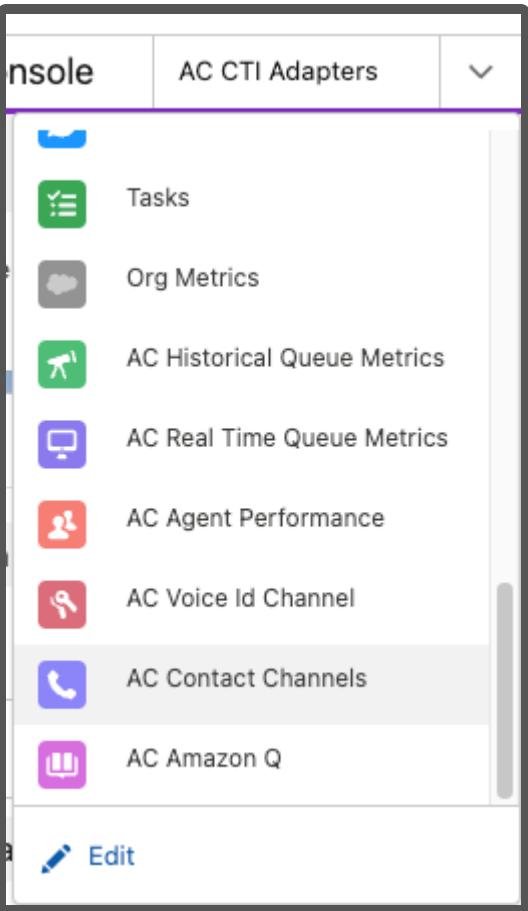
In addition, articles that originated in Salesforce Knowledge have a button that pops out the article into Salesforce Knowledge.

The screenshot shows a mobile application interface. At the top, there is a navigation bar with a phone icon and the word "Phone". Below the navigation bar is a header with "Actions" and "Amazon Q" tabs. A search bar contains the text "test". Underneath the search bar, a section titled "Search results" shows a single result named "Test1". To the right of "Test1" is a red-bordered square highlighting a small icon. Below "Test1", the content area displays the text "Test" and "Test". On the far left, there is a vertical sidebar with a back arrow icon.

The screenshot shows a desktop browser window displaying a knowledge base article. The title of the article is "Test1". The top right corner has "Edit as Draft" and "Archive" buttons. Below the title, there is a summary table with columns: Article Record Type (FAQ), Article Number (000001003), Publication Status (Published), Last Modified Date (1/9/2024, 9:00 AM), and Version Number (1). The main content area is divided into tabs: "Details", "Related", and "Versions". The "Details" tab is active and shows sections for "Information" and "Article Details". The "Information" section contains fields for Title (Test1) and URL Name (Test1). The "Article Details" section contains fields for Question and Answer. To the right of the main content, there is a sidebar with "Was this article helpful?" buttons (like and upvote counts both at 0) and a "Categories (0)" section with an "Expand All" button.

Accessing the Tabbed Version of Amazon Q:

Amazon Q is also accessible in Tabbed form.



Service Console AC Amazon Q v X (AC Lig... v X Tony S... v X Hudson... v X a00DS... v X CCA 0... v X CCA 0... v X CTR 0... v X Test1 v X Test3 v X More v

Q Search Amazon Q

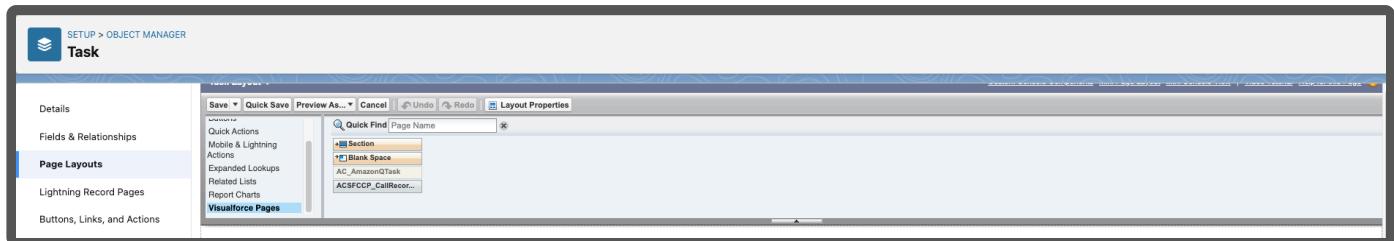
Search results

This screenshot shows the main content area of the application. It features a toolbar at the top with various icons and labels corresponding to the sidebar items. Below the toolbar is a search bar with the placeholder 'Search Amazon Q' and a dropdown menu labeled 'Search results'. The main content area is a large, light-grey rectangular space that appears to be empty or loading.

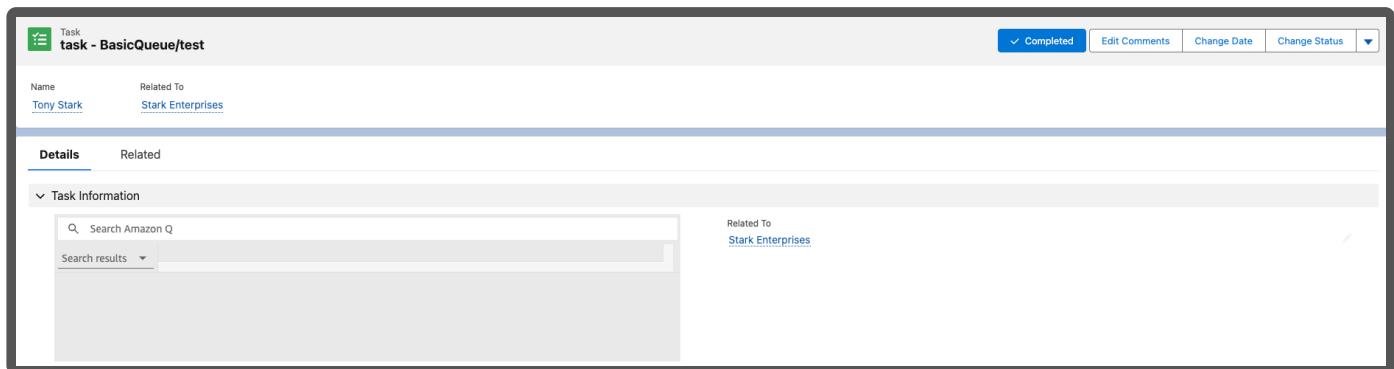
Accessing the Component Version of Amazon Q:

The final method of accessing Amazon Q in Salesforce is through the Amazon Q component.

1. Navigate to Object Manager in Setup
2. Select either Task or Case (note: the Amazon Q component is embeddable in other pages as well, but you may need to write custom classes in order to do so.)
3. Select *Page Layouts*
4. Select the appropriate layout
5. Select *Visualforce Pages* in the top component



6. Click and drag the appropriate Amazon Q visualforce page into the desired location
7. Save the layout
8. Navigate to a task page



Voice Id

The Amazon Connect CTI Adapter allows for integration with Amazon Connect Voice Id.

The integration between Voice Id and the CTI Adapter first requires that Voice Id is set up in the Amazon Connect instance that the CTI Adapter is integrated with. See [here](#) for full instructions.

Before proceeding with the below, please ensure that Voice Id works as expected in a standalone CCP.

Enabling the Voice Id Trigger:

1. In Setup, search for Custom Settings.

2. Click on Custom Settings, and click Manage on the row with the **Toolkit for Amazon Connect** setting

3. Click into your setting (or create one if it doesn't exist)

The screenshot shows the 'Custom Setting' section for 'Toolkit for Amazon Connect'. It includes a note about adding new data for country codes or hierarchies. Below is a table with a single row labeled 'Default Organization Level Value'. The 'View' dropdown is set to 'All'. A red box highlights the 'New' button in the top right corner of the table header.

4. Search and assign the toolkit for either your profile or user, and then uncheck Disable the Voice Id Channel Trigger

The screenshot shows the 'Edit Toolkit for Amazon Connect' page. It displays 'Toolkit for Amazon Connect Information' with fields for 'Location' (set to 'Profile'), 'Disable the CCA Case Trigger', 'Disable the CCA Contact Trigger', 'Disable the Case Contact CCA Trigger', 'Disable the Task Trigger', and 'Disable the Voice Id Channel Trigger'. The 'Url' field is empty. A red box highlights the 'Profile' dropdown in the 'Location' field.

5. Enter the domain of Amazon Connect instance in the Url field (if it doesn't exist already).

6. Click save.

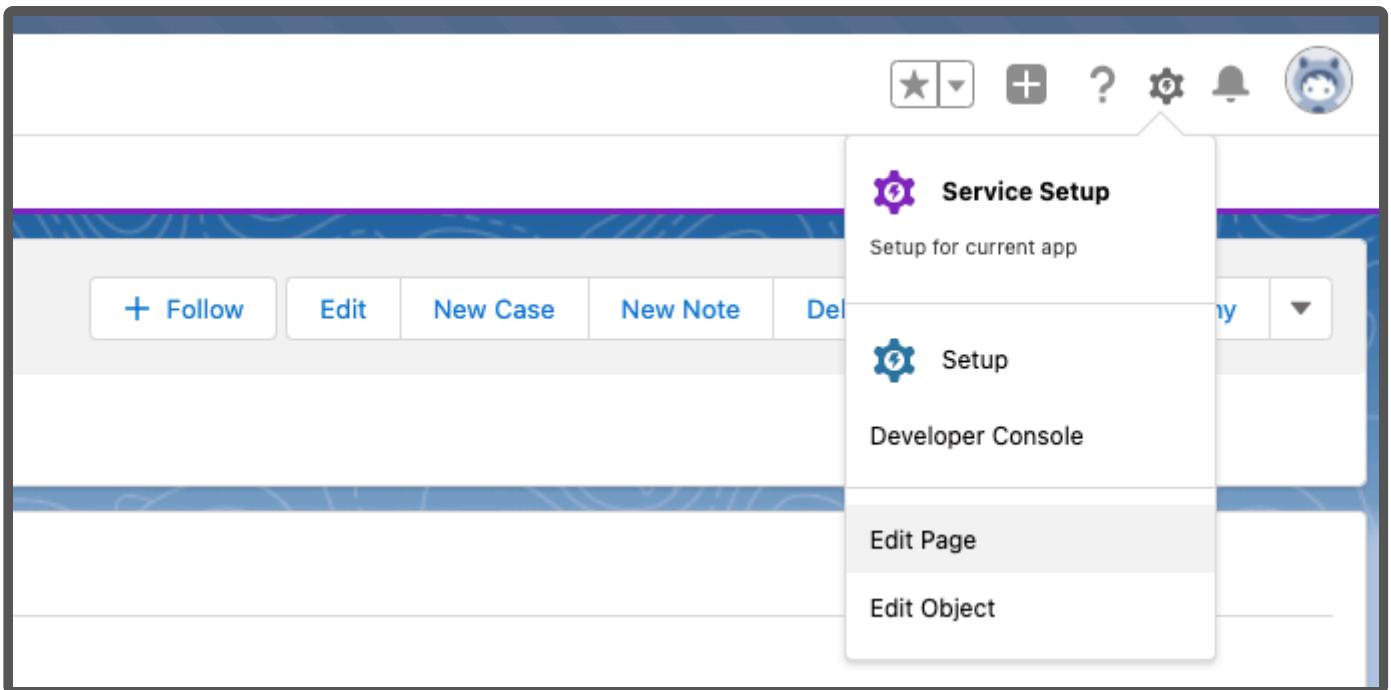
After following the above steps, **AC_VoiceIdChannel__c** records will start to be created on calls where Voice Id is active. These records can be viewed in the AC Voice Id Channel tab:

The screenshot shows a Service Console interface with a purple header bar. On the left is a blue cloud icon, followed by the text "Service Console". To its right is a search bar with a magnifying glass icon. Below the header, there's a toolbar with a red key icon labeled "All", a gear icon, and a dropdown arrow. A message "50+ items • Updated a few seconds ago" is displayed. Below this is a search bar with the placeholder "Search this list...". The main area is a list titled "AC Voice Id Channel Name" with a downward arrow. The list contains eight items, each with a checkbox and a channel name:

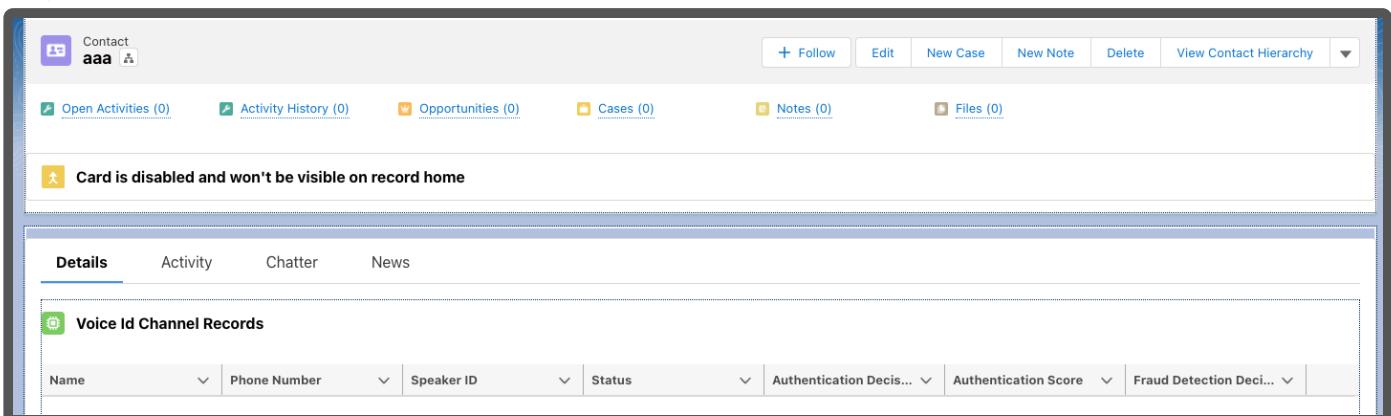
- Voice Id Channel 000000109
- Voice Id Channel 000000108
- Voice Id Channel 000000107
- Voice Id Channel 000000106
- Voice Id Channel 000000105
- Voice Id Channel 000000104
- Voice Id Channel 000000103

Adding Voice Id Components: Add the Voice Id component to the contacts page:

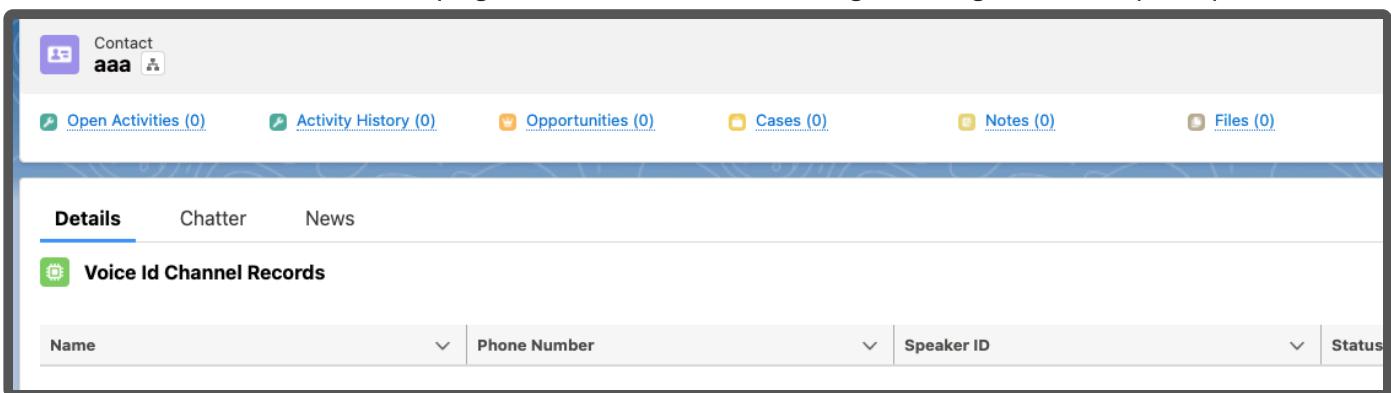
1. Navigate to Contacts list, and create a contact with the phone number you'll use for testing.
2. Click into the created Contact page, on the right-top corner, click the Setup icon and then click Edit Page.



3. Find `ac_VoiceIdChannelListView` in the custom components list, drag and drop it into the page.



4. Save and return to the record page. Click activate and assign as Org Default if prompted.



Add the Voice Id component to the Task/Cases page:

1. Open the task record page, and Edit Page (same steps as Contacts).
2. Find `ac_VoiceIdChannelDetailView` in the custom components list, drag and drop it into the page.

3. Save and return to the record page. Click activate and assign as Org Default if prompted.

The screenshot shows a Salesforce Case feed page. At the top, there's a header with tabs for 'Call' and 'Case'. Below the header, a section titled 'Voice Id Channel Record' displays the message 'No Voice Id Record found.' On the left, there are tabs for 'Feed' and 'Details', with 'Feed' being the active tab. Under the feed tab, there are buttons for 'Post', 'Log a Call', 'Change Priority', and 'Close the Case'. A text input field labeled 'Share an update...' has a 'Share' button next to it. Below these buttons, there's a dropdown menu set to 'Most Recent Activity'. To the right, there's a search bar with the placeholder 'Search this feed...' and two small icons. At the bottom of the feed area, there are tabs for 'All Updates', 'Emails', 'Call Logs', 'Text Posts', and 'Status Changes', with 'All Updates' being the active tab.

Setting up Medialess

Medialess

The Amazon Connect CTI Adapter enables the operation of a cloud contact center in Salesforce within Virtual Desktop Infrastructure (VDI) environments through the utilization of the Medialess feature. The Medialess feature offers advantages for agents using VDI setups, ensuring that audio is accessible on the agent's local system for an enhanced experience. Enabling Medialess mode configures the Salesforce CCP to operate without media, delivering the necessary data for screen pops, etc. while streaming audio to the local system, dependent on your VDI platform.

Prerequisites

1. Install Amazon Connect CTI Adapter version v5.16 or higher in your salesforce instance. For more information, see [the guide here](#)
2. Required AC CTI Adapter feature for all VDI Platforms
 - i. Log In Salesforce instance
 - ii. Open the relevant AC CTI Adapter
 - iii. In the bottom tabs, select the **Features** section and click **New**.
 - iv. Set the **AC Feature Name** to be **EnableMedialessPopout**
 - v. Set the **Value** to be **Enabled:true**

vi. Ensure that the **Active** checkbox is checked, then hit Save.

Set Up for CITRIX VDI Platform

CTI Adapter enables agents to leverage Citrix remote desktop applications to offload audio processing to their local device and to automatically redirect audio to CTI Adapter opened in remote application. In order to know about audio improvement in CCP using VDI, refer to [Amazon Connect audio optimization for Citrix cloud desktops](#). Additionally, refer to [System Requirements](#) for using the Citrix Unified Communications SDK with Amazon Connect

2. Log In Salesforce instance

3. Log in to Salesforce instance

4. Open the relevant AC CTI Adapter.

a. In the bottom tabs, select the **Features** section and click **New**.

5. Set the **AC Feature Name** to be VDIPlatform

6. Set the **Value** to be Name:CITRIX

a. Ensure that the **Active** checkbox is checked, then hit Save.

7. Launch the SoftPhone and log in to CCP.

8. Verify the configuration by initiating a Voice contact.

Note that once this feature is active in CTI Adapter, the CCP can be only used in a CITRIX environment, otherwise it will show an error as shown below.



The Device Settings for the CCP which is opened in a Citrix environment, cannot be managed directly from the CCP level itself. In order to change the device settings for the CCP, for example changing the device input device, it has to be done from the OS level settings

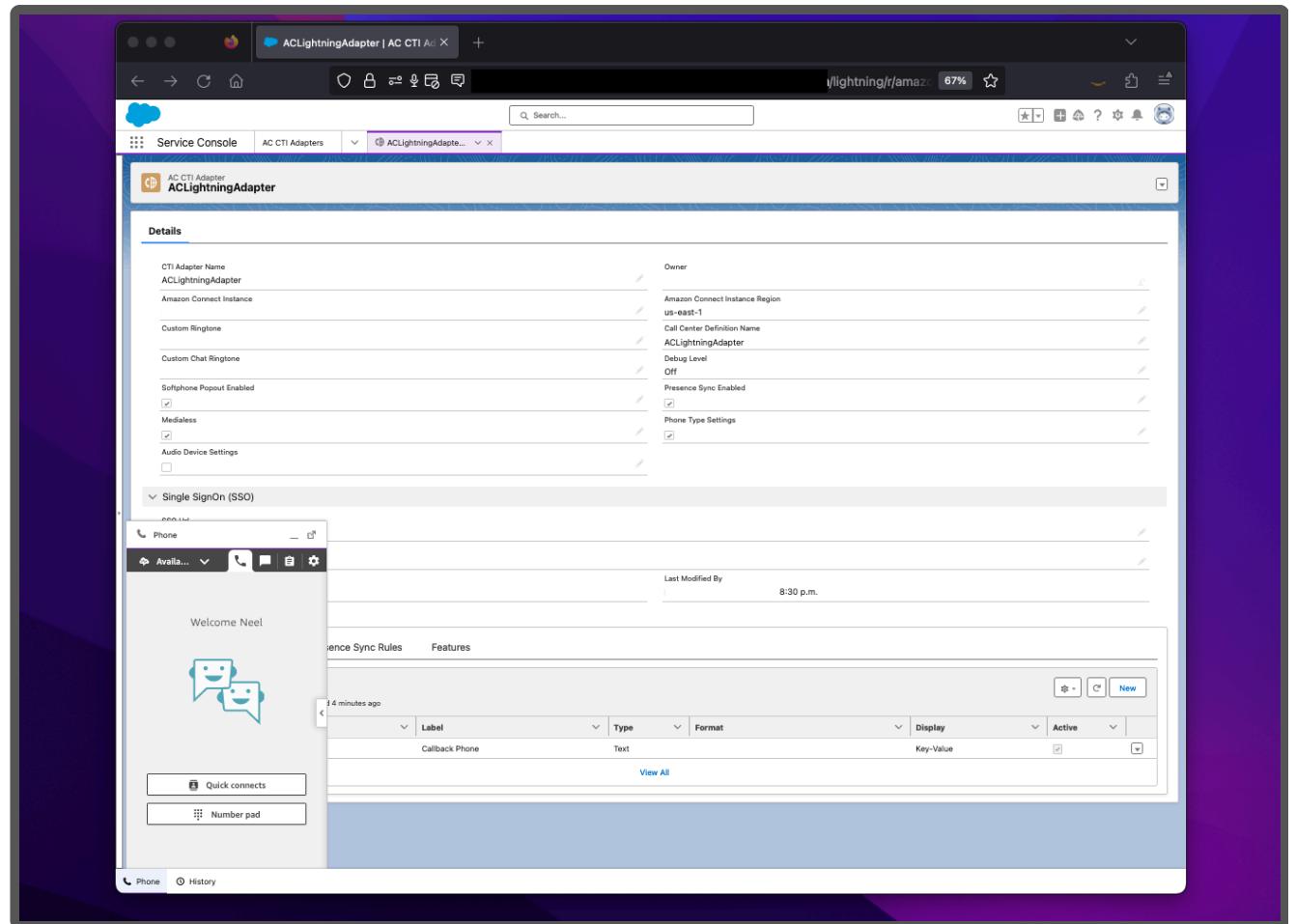
A VDI Platform should be set only if the Medialess settings are disabled. Therefore, if you want to set any VDI Platform, then disable Medialess from CTI Adapter. Similarly, if you want to use Medialess Settings, then first disable VDI Platform Settings from Features

Set Up for Other VDI Platforms

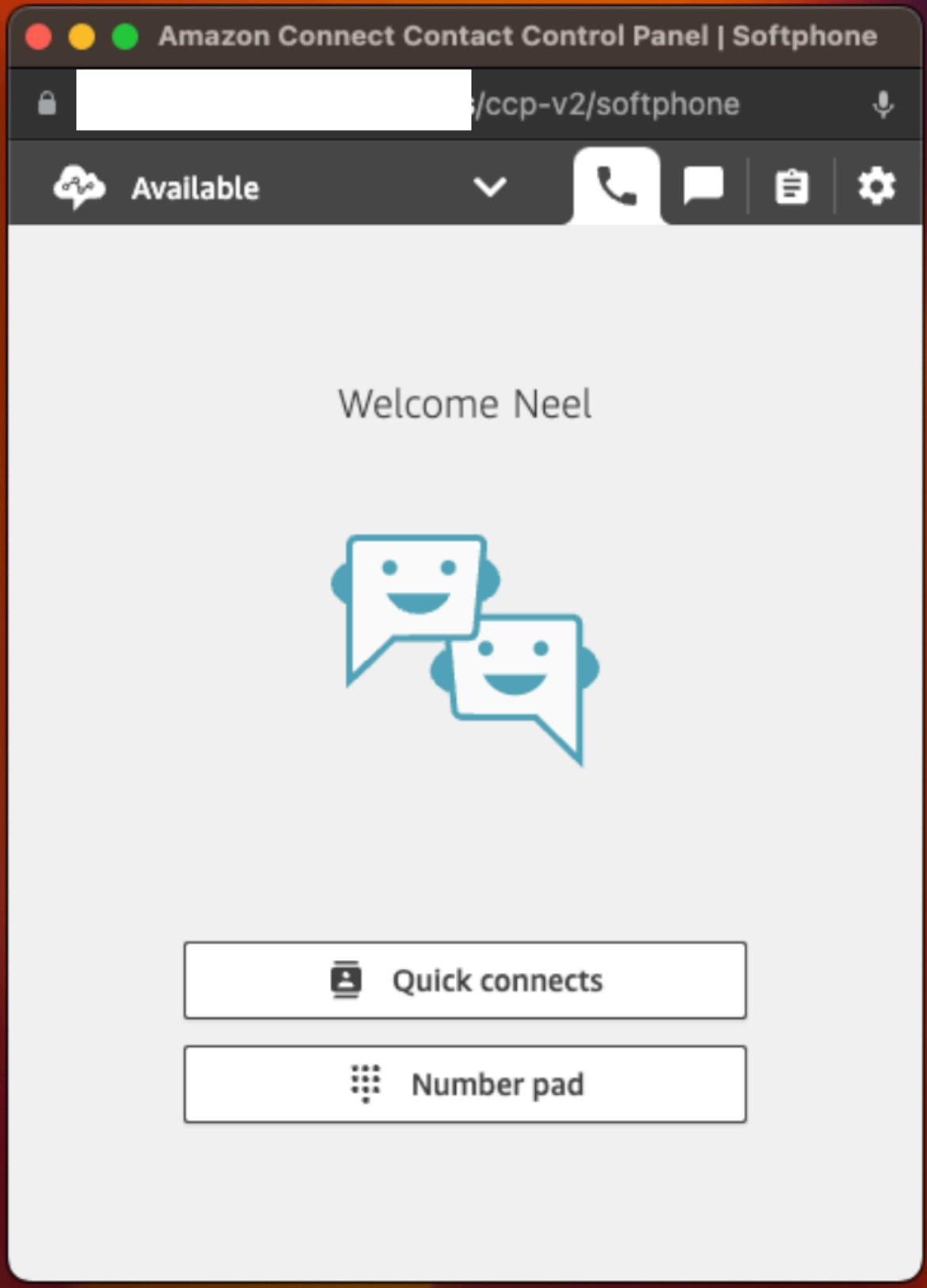
1. Login into your VDI environment.

2. Log In Salesforce instance

3. Open the relevant AC CTI Adapter.
4. In the details section, activate the "Medialess" option by marking the checkbox.
5. Refresh the browser. Launch the SoftPhone and log in to CCP.
6. Upon signing in, click the Toggle Embedded CCP button located within the Softphone's CCP. Close all CCP instances except the one within the Salesforce CTI Adapter. Ensure that your virtual environment mirrors the configuration shown in the following image.



7. Go to your Local System and login to Amazon Connect (e.g. login in connect <https://youraccount.my.connect.aws/>) and open Native CCP by clicking on Contact Control Panel.



8. Verify the configuration by initiating a Voice contact. All media, including audio, will be transmitted through the Native CCP on your local desktop. The CCP within the CTI Adapter of the Virtual Environment can be employed for contact management.

9. Important: Ensure that both CCP instances are open when handling contacts. One CCP should be within the SoftPhone in the Salesforce CTI Adapter of the virtual environment, and the second CCP should be the native CCP on your local system.

Accessing the Salesforce API from Amazon Connect Contact Flows Using AWS Lambda

The most commonly used feature of the AWS Serverless Application Repository for Salesforce is accessing/updating Salesforce data using the `sflnvokeAPI` Lambda function. This function allows an Amazon Connect contact flow to perform the following operations against your Salesforce org:

- **Lookup:** queries Salesforce for objects based on the parameters passed to it
- **Create:** creates a Salesforce object based on the parameters passed to it
- **Update:** updates a Salesforce object based on the parameters passed to it
- **Phone Lookup:** uses Salesforce Object Search Language (SOSL) to construct text-based search queries against the search index, which gives significant performance improvement when searching phone number fields.
- **Delete:** deletes a Salesforce object based on the parameters passed to it
- **Query:** executes a Salesforce Object Query Language (SOQL) query on the Salesforce instance. Can return multiple entries.
- **QueryOne:** executes a Salesforce Object Query Language (SOQL) query on the Salesforce instance. Returns result only when one entry is returned from the query.
- **CreateChatterPost:** creates a chatter post.
- **CreateChatterComment:** creates a chatter comment.
- **Search:** performs a search against the Salesforce instance, returning all results.
- **SearchOne:** performs a search against the Salesforce instance, returning at most one result.

NOTE: naming of the Lambda function will vary based on template data, but `sflnvokeAPI` will always be a part of the name.

When you invoke this Lambda function from your contact flows, you will need to pass along parameters that inform the function as to which Salesforce operation you wish to execute, as well as pass along any required parameters. Depending on your use case, this can require reference to the [Salesforce REST API](#) or the [Salesforce Connect REST API](#) documentation. The core parameters are:

- **sf_operation:** specifies which operation to run. Options are lookup, create, update, phoneLookup, query, queryOne, createChatterPost, createChatterComment
- **sf_object:** defines what type of object you are referencing. Examples include Case, Contact, Task, etc.
- **sf_fields:** the fields you want to receive back from Salesforce when an operation completes successfully
- **sf_id:** the unique identifier for a Salesforce object. Typically used in update operations
- **sf_phone:** contains the phone number used to search when performing a phone lookup

Salesforce Lookup

This operation is invoked by setting **sf_operation** to **lookup**. In this case, the Lambda function queries Salesforce for objects based on the parameters passed to it. For lookup, the following parameters are required:

- sf_object
- sf_fields

Any additional parameters passed will be evaluated as conditional arguments for the lookup.

Note that this operation only returns the first item of the query results. If you want to have all results returned from Salesforce, set **sf_operation** to **lookup_all**.

In the contact flow example below, we are looking for a specific case based on customer input.

Invoke AWS Lambda function

Makes a call to AWS Lambda, and optionally returns key / value pairs.

Function input parameters

Use text

Use attribute

Destination key

CaseNumber

Type

Lex slots

Attribute

case_id

Use text

Destination key

sf_operation

Value

lookup

Use attribute

Use text

Destination key

sf_object

Value

Case

Use attribute

Use text

Destination key

sf_fields

Value

Id

Use attribute

This operation returns a response of:

```
{  
  "Id": "5006g00000AaIs7AAF",  
  "sf_count": 1  
}
```

For **lookup_all** the operation returns a response of:

```
{  
  "sf_records_0_Id": "5006g00000AaIs7AAF",  
  "sf_records_1_Id": "5006g00000AaIs7AAE",  
  "sf_count": 2  
}
```

Note that `sf_count` is the count of records matched and not the count of fields in the response.

This means all fields that start with `sf_records_i_` count as one record. If the query above returned the Name as well as the Id the response will be:

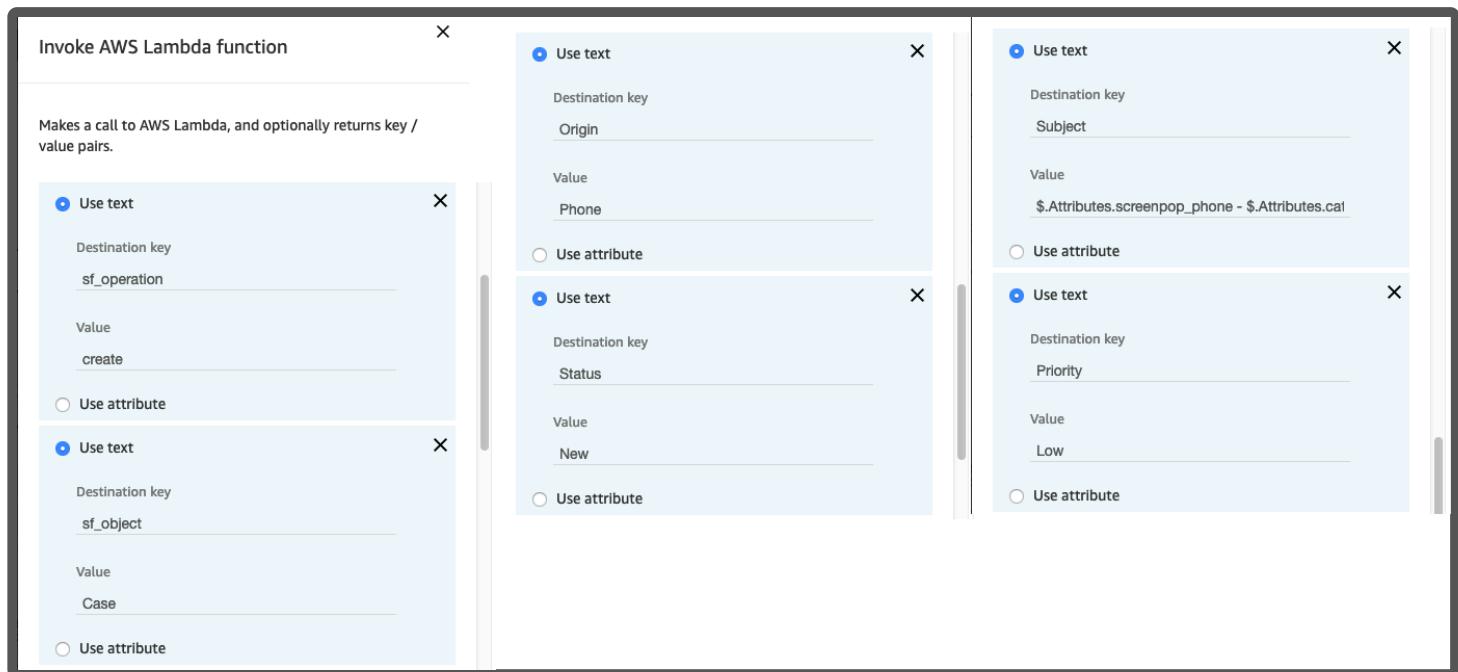
```
{
  "sf_records_0_Id": "5006g00000AaIs7AAF",
  "sf_records_0_Name": "Name0",
  "sf_records_1_Id": "5006g00000AaIs7AAE",
  "sf_records_1_Name": "Name1",
  "sf_count": 2
}
```

Salesforce Create

This operation is invoked by setting **sf_operation** to **create**. In this case, the Lambda function creates a Salesforce object based on the parameters passed to it. For create, the following parameters are required:

- **sf_object**
- Specify additional parameters for the Salesforce object to be created. Please be sure to include all parameters required to create the Salesforce object.

In the contact flow example below, we creating a new case based on customer input.



This operation returns a response of:

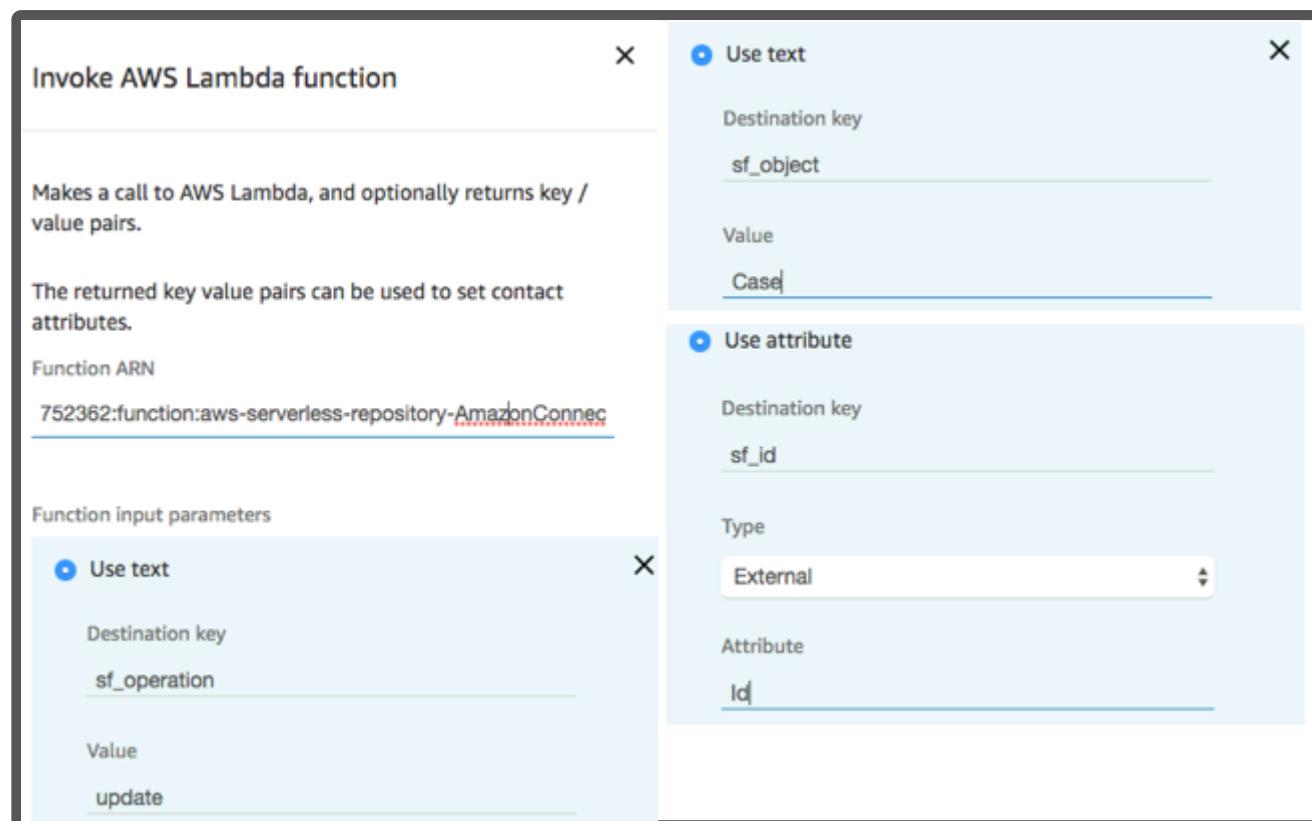
```
{
  "Id": "5006g00000BLqurAAD"
}
```

Salesforce Update

This operation is invoked by setting **sf_operation** to **update**. In this case, the Lambda function updates a Salesforce object based on the parameters passed to it. For update, the following parameters are required:

- sf_object
- sf_id
- Specify additional parameters for the Salesforce object to be created. Please be sure to include all parameters required to create the Salesforce object.

In the contact flow example below, we are updating a specific case.



This operation returns a response of:

```
{  
  "Status": "204"  
}
```

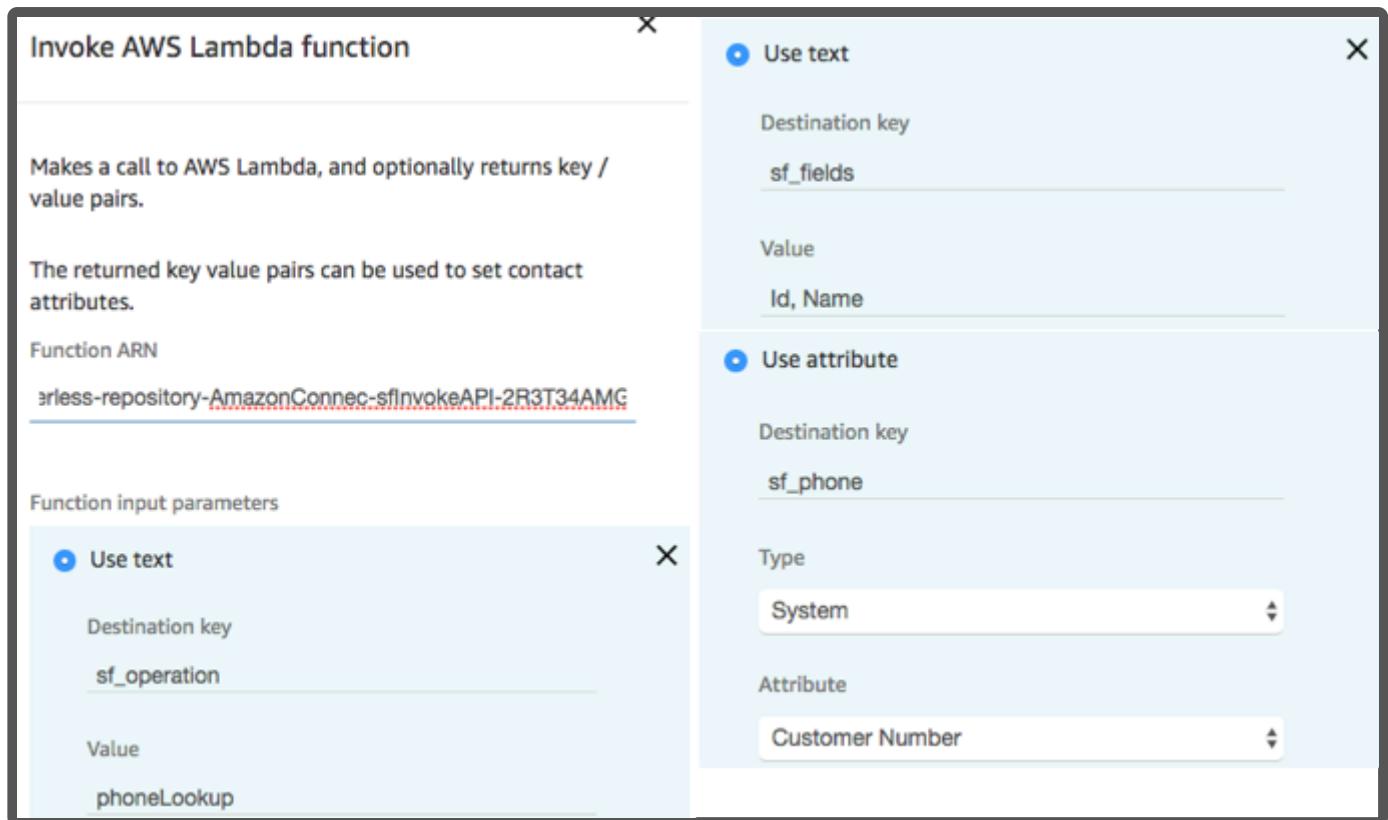
The "204" status indicates a success.

Salesforce Phone Lookup

This operation is invoked by setting **sf_operation** to **phoneLookup**. In this case, the Lambda function uses Salesforce Object Search Language (SOSL) to construct text-based search queries. For phoneLookup, the following parameters are required:

- sf_phone
- sf_fields

In the contact flow example below, we look for a customer by phone number.



This operation returns a response of:

```
{
  "Id": "5006g00000BLqurAAD",
  "sf_count": "1",
  "Name": "Jim Smith"
}
```

Salesforce Delete

This operation is invoked by setting **sf_operation** to **delete**. In this case, the Lambda function deletes a Salesforce object based on the parameters passed to it. For delete, the following parameters are required:

- sf_object
- sf_id

In the contact flow example below, we deleting an existing case based on customer input.

Use text

X

Destination key

sf_object

Value

Case

Use attribute

Use text

X

Destination key

sf_id

Value

5004T000004gsR1QAI

Use attribute

[Add another parameter](#)

Invoke AWS Lambda function

X

Makes a call to AWS Lambda and optionally returns key/value pairs, which can be used to set contact attributes. [Learn more](#)

Function ARN

Select a function

serverlessrepo-AmazonConnectSalesforce-sfInvokeAPI- ▾

Use attributes

Function input parameters

Use text

Destination key

sf_operation

Value

delete

Use attribute

This operation returns a response of:

```
{  
  "Response": "None"  
}
```

Salesforce query

This operation is invoked by setting **sf_operation** to **query**. In this case, the Lambda function uses Salesforce Object Query Language (SOQL) to conduct a query against the Salesforce instance. For query, the following parameter is required:

- query

Any additional parameters will replace text values in the original query so that queries can be dynamic based on values stored within the contact flow. For example, the parameter set:

- query: "select field from object"
- field: "Id"
- object: "Task"

Will result in the query: "select Id from Task".

In the contact flow example below, we look for a customer by phone number.

Function input parameters

Use text X

Destination key

sf_operation

Value

query

Use attribute

Use text X

Destination key

query

Value

select Id from Contact where Phone LIKE '%numl

Use attribute

(full text of the value is "select Id from Contact where Phone LIKE '%number%'")

Use text

X

 Use attribute

Destination key

number

Type

System



Attribute

Customer Number



This operation returns a response of:

```
{  
  "sf_records_0_Id": "00303000001RZfIAAW",  
  "sf_count": 1  
}
```

Note that `sf_count` is the count of records matched and not the count of fields in the response.

This means all fields that start with `sf_records_i_` count as one record. If the query above returned the Name as well as the Id and matched more than one record, the response will be:

```
{  
  "sf_records_0_Id": "00303000001RZfIAAW",  
  "sf_records_0_Name": "Name0",  
  "sf_records_1_Id": "00303000001RZfIAAE",  
  "sf_records_1_Name": "Name1",  
  "sf_count": 2  
}
```

Salesforce queryOne

This operation is invoked by setting **sf_operation** to **queryOne** (case sensitive). In this case, the Lambda function uses Salesforce Object Query Language (SOQL) to conduct a query against the Salesforce instance, returning a result only when one record is returned from the query. For query, the following parameter is required:

- query

Any additional parameters will replace text values in the original query so that queries can be dynamic based on values stored within the contact flow. For example, the parameter set:

- query: "select field from object"
- field: "Id"
- object: "Task"

Will result in the query: "select Id from Task".

In the contact flow example below, we look for a customer by phone number.

Use text

X

Destination key

sf_operation

Value

queryone

Use attribute

Use text

X

Destination key

query

Value

select Id from Contact where Phone LIKE '%numl

Use attribute

(full text of the value is "select Id from Contact where Phone LIKE '%number%'")

Use text

X

 Use attribute

Destination key

number

Type

System



Attribute

Customer Number



This operation returns a response of:

```
{  
  "Id": "00303000001RZfIAAW",  
  "sf_count": 1  
}
```

Salesforce createChatterPost

This operation is invoked by setting **sf_operation** to **createChatterPost** (case sensitive). In this case, the Lambda function uses the Salesforce Connect REST API to create a chatter post (see [here](#)). For createChatterPost, the following parameters are required:

- sf_feedElementType
- sf_subjectId
- sf_messageType
- sf_message

The following parameter is optional:

- sf_mention

(refer to the api reference for value types)

Any additional parameters will replace text values in the sf_message so that messages can be dynamic based on values stored within the contact flow. For example, the parameter set:

- sf_message: "Please help me with case `caseId`"
- caseld: 1234

Will result in the message: "Please help me with case 1234".

In the contact flow example below, we leave a chatter post on a contact.

Use text

X

Destination key

sf_operation

Value

createChatterPost

Use attribute

X

Use text

Destination key

sf_feedElementType

Value

FeedItem

Use attribute

Use text

X

Destination key

sf_subjectId

Value

00303000001RZfIAAW

Use attribute

Use text

X

Destination key

sf_messageType

Value

Text

Use attribute

Use text



Destination key

sf_message

Value

I had a problem during the call. My contact id is {}

Use attribute

(full text of the value is "I had a problem during the call. My contact id is contactId.")

Use text



Use attribute

Destination key

contactId

Type

System



Attribute

Contact id



The operation returns a response of:

```
{  
    "Id": "0D503000000ILY5CA0"  
}
```

See the chatter post appear attached to the Subject:

Activity

Chatter

Post

Poll

Question

Share an update...

Share



Search this feed...



apiuser

1m ago



I had a problem during the call. My contact id is 31b41a0b-75a8-449d-adb8-3f5f247a73d6.

Like

Comment



Write a comment...

Salesforce createChatterComment

This operation is invoked by setting **sf_operation** to **createChatterComment** (case sensitive). In this case, the Lambda function uses the Salesforce Connect REST to create a chatter comment (see [here](#)). For createChatterComment, the following parameters are required:

- sf_feedElementId
- sf_commentType
- sf_commentMessage

(refer to the api reference for value types)

Any additional parameters will replace text values in the sf_commentMessage so that messages can be dynamic based on values stored within the contact flow. For example, the parameter set:

- sf_commentMessage: "Please help me with case `caseId`"
- caseld: 1234

In the contact flow example below, we leave a comment on a chatter post.

Use text



Destination key

sf_operation

Value

createChatterComment

Use attribute

Use text



Destination key

sf_feedElementId

Value

0D503000000ILY5CAO

Use attribute

Use text



Destination key

sf_commentType

Value

Text

Use attribute



Use text

Destination key

sf_message

Value

This concern has been addressed.

Use attribute

The operation returns a response of:

```
{  
    "Id": "0D70300000ChhNCAS"  
}
```

See the chatter post appear attached to the Subject:



apiuser

8m ago



I had a problem during the call. My contact id is dda99fbf-6186-4125-ba59-c461d620fdbd.

1 comment · Seen by 1



Like



Comment



apiuser



a few seconds ago

This concern has been addressed.

Like



Write a comment...

Salesforce search

This operation is invoked by setting **sf_operation** to **search** (case sensitive). In this case, the Lambda function uses the Salesforce REST to perform a parameterized search (see [here](#)). For search, the following parameters are required:

- q
- sf_fields
- sf_object

The following parameters are optional:

- where
- overallLimit

(refer to the api reference for value types)

See the below example:

Use text X

Destination key

sf_operation

Value

search

Use attribute

Use text X

Destination key

q

Value

test

Use attribute

Use text



Destination key

sf_object

Value

Case

Use attribute

Use text



Destination key

sf_fields

Value

Subject, Status

Use attribute

Use text

X

Destination key

overallLimit

Value

3

Use attribute

Use text

X

Destination key

where

Value

Status like 'New'

Use attribute

The operation returns a response of:

```
{  
    "sf_records_0_Id": "50001000001B9e6AAG",  
    "sf_records_0_Subject": "test subject",  
    "sf_records_0_Status": "New",  
    "sf_records_1_Id": "50001000001B9eWAAS",  
    "sf_records_1_Subject": "test subject",  
    "sf_records_1_Status": "New",  
    "sf_records_2_Id": "50001000001BDgiAAG",  
    "sf_records_2_Subject": "test subject",  
}
```

```
"sf_records_2_Status": "New",  
"sf_count": 3  
}
```

Note that `sf_count` is the count of records matched and not the count of fields in the response. This means all fields that start with `sf_records_i_` count as one record.

Salesforce searchOne

This operation is invoked by setting `sf_operation` to **searchOne** (case sensitive). In this case, the Lambda function uses the Salesforce REST to perform a parameterized search (see [here](#)). For search, the following parameters are required:

- `q`
- `sf_fields`
- `sf_object`

The following parameter is optional:

- `where`

(refer to the api reference for value types)

See the below example:

Use text



Destination key

sf_operation

Value

searchOne

Use attribute

Use text



Destination key

q

Value

test subject unique

Use attribute

Use text



Destination key

sf_object

Value

Case

Use attribute

Use text



Destination key

sf_fields

Value

Subject, Status

Use attribute

Use text

X

Destination key

overallLimit

Value

3

Use attribute

Use text

X

Destination key

where

Value

Status like 'New'

Use attribute

The operation returns a response of:

```
{  
  "Id": "50001000001BIn6AAG",  
  "Subject": "test subject unique",  
  "Status": "New",  
  "sf_count": 1  
}
```

Amazon Connect Historical Metrics in Salesforce

Amazon Connect can generate a number of historical metric reports to monitor efficiency and utilization, agent performance, and other information about your contact center. Amazon Connect provides you the ability to schedule execution and export of reports, in comma separated value (CSV) format, to the S3 bucket of your choice. This enables broad compatibility across many analytics and WFM tools.

With the AWS Serverless Repository for Salesforce, you can configure the automatic import of reporting data from Amazon Connect into Salesforce. Two different historical reports are available to transport Agent and Queue interval data from Amazon Connect to Salesforce. Once these have been configured and scheduled, you will begin to see data available in the reports that have been included with the CTI Adapter.

Configuring the AWS Services

When you configure schedule reports to run in Amazon Connect, they are saved to your reporting Amazon S3 bucket upon execution. As a part of the schedule configuration, you can determine the frequency with which data is exported. The standard configuration is for execution every 30 minutes; however you can increase the interval time to suit your requirements.

Once you have the reports configured and scheduled, you will then need to activate the trigger for the reports bucket that will invoke an AWS Lambda function included in the AWS Serverless Repository for Salesforce. This function will process the report and import the data to Salesforce.

NOTE:

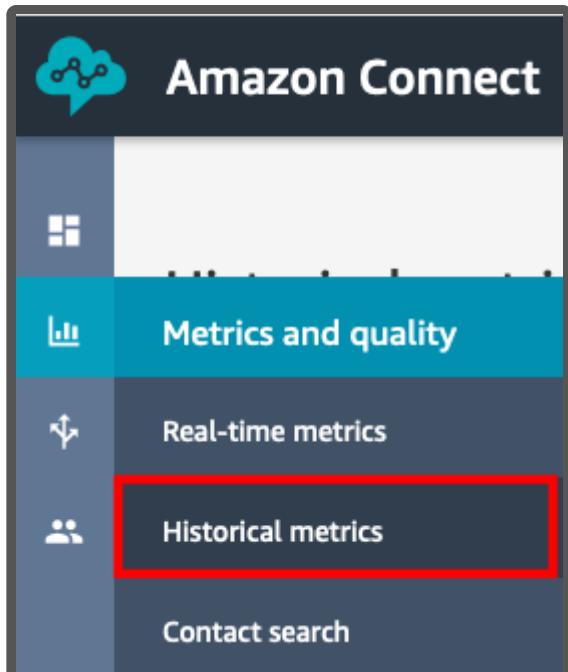
Each CTI adapter release will only be able to support a subset of all of the metrics available as new metrics are continuously being added by Connect after CTI adapter releases have already been released to the public.

When migrating from an older version of the CTI Adapter (At least earlier than v4.5), users may notice that `ACSFCCP_HistoricalReportData` is no longer being updated/populated. We have moved away from using this object but because Salesforce does not allow objects to be deleted, customers may still have these object types present in their Salesforce instance.

Configuring the Historical Reports in Amazon Connect

1. Login to your Amazon Connect instance as an Administrator

2. From the left navigation, choose **Metrics and Quality** then select **Historical metrics**



3. On the **Historical metrics** page, select Contact metrics

A screenshot of the 'Historical metrics' page. At the top, there is a message: 'Select the type of report and metrics you would like to view.' Below this are three categories: 'Queues', 'Agents', and 'Phone numbers'. Each category has a sub-section to its right. The 'Queues' section has a sub-section labeled 'Contact metrics' which is highlighted with a red box. The 'Agents' section has a sub-section labeled 'Agent performance'. The 'Phone numbers' section has a sub-section labeled 'Contact metrics'. There are also small dropdown arrows next to each category's sub-section.

4. Once the **Historical metrics: Queues** report loads, select the cog in the upper right to edit the report

5. On the **Interval & Time** range tab, set the parameters as follows:

a. Interval: 30 minutes

b. Time Zone: UTC

c. Time Range: Last 24 Hours

6. Leave the **Groupings** and **Filters** tabs set to their defaults

7. Select the **Metrics** Tab.

8. Select ALL selectable options

a. Note: With each release, we support a subset of the fields given. If you see an **INVALID_FIELD** error in your Lambda Cloudwatch (with the "Adherence__c" field for example), the selectable option for that field was most likely added after we released a new version.

9. Select **Apply**

10. Once the report saves, select the dropdown menu next to the Save button and choose Schedule

11. Set the name as **sflIntervalQueue** and choose **Continue**

12. On the **Note** screen, choose **Continue**

13. On the **Recurrence** tab in the Schedule Report setup, set the options as:

a. Generate this report: Hourly

b. Every: 0.5 hour(s)

c. Starting at: 1AM

d. For the Previous: 0.5 hour(s)

Schedule Report

sfIntervalQueue

Recurrence

Delivery Options

Generate this report

Hourly ▾ every 0.5 ▾ hour(s)

Starting at

Time zone

1 AM ▾

UTC

For the previous

0.5 ▾ hour(s)

Create

Cancel

14. Select the **Delivery Options** tab

15. In the Prefix field, enter **SFDC/Queue**

Schedule Report

sIntervalQueue

Recurrence

Delivery Options

Default location

connect-[REDACTED]/connect/sfctifinal022020/Reports

Prefix

SFDC/Queue

File name

connect-[REDACTED]/connect/sfctifinal022020/Reports/SFDC/Queue/sIntervalQueue-YYYY-MM-DDThh:mm:ssZ.csv

16. Note the File name. The file name contains the bucket, path, and filename that will be used when executing the report. You will use the **bucket name** and **path** in later steps.

Schedule Report

sIntervalQueue

Recurrence

Delivery Options

Default location

connect-b0e7681ccc4d/connect/sfctifinal022020/Reports

Prefix

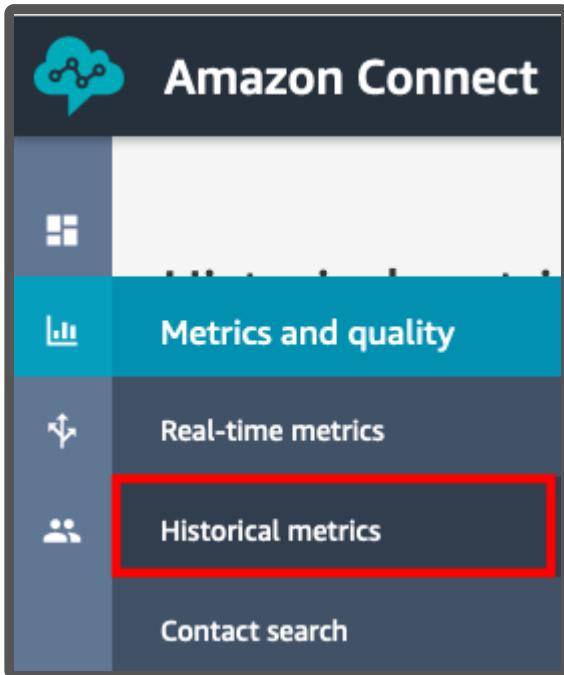
SFDC/Queue

File name

connect-[REDACTED]/connect/sfctifinal022020/Reports/SFDC/Queue/sIntervalQueue-YYYY-MM-DDThh:mm:ssZ.csv

17. Choose **Create**

18. Once the report is created, from the left navigation, choose **Metrics and Quality** then select **Historical metrics**



19. On the **Historical metrics** page, select **Agent performance**

This screenshot shows the 'Historical metrics' report for agents. At the top, it says 'Select the type of report and metrics you would like to view.' Below that are three tabs: 'Queues' (selected, grey background), 'Agents' (white background), and 'Phone numbers'. Under each tab, there are two sub-options: 'Contact metrics' and 'Agent performance' (which is highlighted with a red rectangular border). To the right of each sub-option is a dropdown arrow icon.

20. Once the **Historical metrics: Agents** report loads, select the cog in the upper right to edit the report

21. On the **Interval & Time** range tab, set the parameters as follows:

- a. Interval: 30 minutes
- b. Time Zone: UTC
- c. Time Range: Last 24 Hours

22. Leave the **Groupings** and **Filters** tabs set to their defaults

- a. If you do decide to modify the groupings, be sure to keep Intervals as a grouping. We use this internally to categorize the reports on the Salesforce side

Table Settings

X

Interval & Time range

Groupings

Filters

Metrics

Select the values you'd like to group your metrics by, and add them to the right in the order you prefer.

Grouping options

Selected groupings (Maximum 5)

Agent Hierarchy Level One

+

Interval

i

Agent Hierarchy Level Two

+

Agent

-

Agent Hierarchy Level Three

+

3

Agent Hierarchy Level Four

+

4

Agent Hierarchy Level Five

+

5

Phone Number

+

Routing Profile

+

Cancel

Apply

23. Select the **Metrics** Tab.

24. Select the following metrics (deselect any others):

Note You should be able to use all metrics, but these are the important ones.

- After contact work time
- Agent on contact time
- Agent idle time
- Non-Productive Time
- Average after contact work time
- Average handle time
- Average customer hold time
- Average agent interaction and customer hold time
- Average agent interaction time
- Contacts agent hung up first
- Contacts handled

- Contacts handled incoming
- Contacts handled outbound
- Contacts put on hold
- Contacts hold disconnect
- Contacts transferred out
- Contacts transferred out internal
- Contacts transferred out external
- Error status time
- Agent answer rate
- Agent non-response
- Occupancy
- Online time
- Agent interaction and hold time
- Agent interaction time
- Average outbound agent interaction time
- Average outbound after contact work time

25. Select **Apply**

26. Once the report saves, select the dropdown menu next to the Save button and choose Schedule

27. Set the name as **sflIntervalAgent** and choose **Continue**

28. On the **Note** screen, choose **Continue**

29. On the **Recurrence** tab in the Schedule Report setup, set the options as:

- a. Generate this report: Hourly
- b. Every: 0.5 hour(s)
- c. Starting at: 1AM
- d. For the Previous: 0.5 hour(s)

Schedule Report

sflintervalAgent

Recurrence

Delivery Options

Generate this report

Hourly ▾ every 0.5 ▾ hour(s)

Starting at

1 AM ▾

Time zone

UTC

For the previous

0.5 ▾ hour(s)

30. Select the **Delivery Options** tab

31. In the Prefix field, enter **SFDC/Agent**

sflintervalAgent

Recurrence

Delivery Options

Default location

connect-[REDACTED]/connect/sfctifinal022020/Reports

Prefix

SFDC/Agent

File name

connect-[REDACTED]/connect/sfctifinal022020/Reports/SFDC/Agent/sflintervalAgent-YYYY-MM-DDThh:mm:ssZ.csv

32. Note the File name. The file name contains the bucket, path, and filename that will be used when executing the report. You will use the **bucket name** and **path** in later steps.

File name

connect-[REDACTED]/connect/sfctifinal022020/Reports/SFDC/Agent/sflintervalAgent-YYYY-MM-DDThh:mm:ssZ.csv

33. Choose **Create**

Once you have created the two reports and set their schedule, the next thing you will need to do is to configure a trigger that executes a Lambda function when the report is generated and stored in S3.

Creating the AWS Lambda Trigger for the Queue Data

1. In a new browser tab, login to the [AWS console](#)
2. Open the [AWS Lambda Console](#)
3. In the Add filter field of the AWS Lambda console, enter sflIntervalQueue and press enter to filter the list of functions
4. Select the Lambda function that includes sflIntervalQueue in the name
5. Expand the Designer section
6. Select Add trigger

The screenshot shows the AWS Lambda Function Configuration page. At the top, there are three tabs: Configuration (highlighted in orange), Permissions, and Monitoring. Below the tabs, a section titled "Designer" is expanded, indicated by a downward arrow icon. A blue link "Go back to application serverlessrepo-AmazonConnectSalesforceLambda" is visible. On the right side, there is a list of resources: "serverlessrepo-Ama" (with a red box around it) and "sflIntervalQueue-3ZN" (with a blue box around it), followed by "Layers". At the bottom left, there is a button labeled "+ Add trigger" with a red box around it.

7. In Trigger configuration, select S3 from the dropdown list

Add trigger

Trigger configuration

Select a trigger



DynamoDB

aws database nosql



Kinesis

analytics aws streaming



S3

aws storage

S3



SNS

aws messaging notifications pub-sub push



SQS

aws queue

8. Referring to the notes from the report configuration earlier, select the appropriate bucket
9. Change the Event type to PUT
10. Referring to the notes from the report configuration earlier, set the Prefix to the path value for your report
11. Set the Suffix to .csv
12. The trigger configuration should now be similar to the following:

Add trigger

Trigger configuration



S3
aws storage



Bucket

Please select the S3 bucket that serves as the event source. The bucket must be in the same region as the function.

connect-[REDACTED]



Event type

Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual events cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.

PUT



Prefix - optional

Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters.

connect/sfctifinal022020/Reports/SFDC/Queue/

Suffix - optional

Enter a single optional suffix to limit the notifications to objects with keys that end with matching characters.

.CSV

Lambda will add the necessary permissions for Amazon S3 to invoke your Lambda function from this trigger. [Learn more](#) about the Lambda permissions model.

Enable trigger

Enable the trigger now, or create it in a disabled state for testing (recommended).

Cancel

Add

13. Select Add

14. If everything has been configured correctly, you should receive a success message.

Creating the AWS Lambda Trigger for the Agent Data

1. In a new browser tab, login to the [AWS console](#)
2. Open the [AWS Lambda Console](#)
3. In the Add filter field of the AWS Lambda console, enter sfIntervalAgent and press enter to filter the list of functions
4. Select the Lambda function that includes sfIntervalAgent in the name

5. Expand the Designer section

6. Select Add trigger

The screenshot shows the AWS Lambda Configuration interface. At the top, there are three tabs: Configuration (highlighted in orange), Permissions, and Monitoring. Below the tabs, a section titled 'Designer' is expanded, indicated by a downward arrow icon. A blue link 'Go back to application serverlessrepo-AmazonConnectSalesforceLambda' is visible. On the right side, there is a Lambda function card for 'serverlessrepo-AmazonConnectSalesforceLambda' and a 'Layers' section. In the bottom-left corner of the main area, a red box highlights a white button with a plus sign and the text '+ Add trigger'.

7. In Trigger configuration, select S3 from the dropdown list

The screenshot shows the 'Add trigger' configuration page under the Lambda section. The title 'Add trigger' is at the top. Below it is a section titled 'Trigger configuration'. A search bar labeled 'Select a trigger' contains the text 'S3'. A list of trigger types is shown, with the 'S3' option highlighted by a red box. Other options include DynamoDB, Kinesis, SNS, and SQS. To the right of the S3 option, there is a small 'S3' icon.

8. Referring to the notes from the report configuration earlier, select the appropriate bucket
9. Change the Event type to PUT
10. Referring to the notes from the report configuration earlier, set the Prefix to the path value for your report
11. Set the Suffix to .csv
12. The trigger configuration should now be similar to the following:

Add trigger

Trigger configuration

 S3
aws storage

Bucket
Please select the S3 bucket that serves as the event source. The bucket must be in the same region as the function.
connect-[REDACTED] ▼ C

Event type
Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual events cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.
PUT ▼

Prefix - optional
Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters.
connect/sfctifinal022020/Reports/SFDC/Agent/

Suffix - optional
Enter a single optional suffix to limit the notifications to objects with keys that end with matching characters.
.csv

Lambda will add the necessary permissions for Amazon S3 to invoke your Lambda function from this trigger. [Learn more](#) about the Lambda permissions model.

Enable trigger
Enable the trigger now, or create it in a disabled state for testing (recommended).

Cancel Add

13. Select **Add**
14. If everything has been configured correctly, you should receive a success message.

Verifying the Data Import in Salesforce

Once you have configured the reports and added the triggers, you should start to see data in Salesforce after ~30 minutes. The Amazon Connect CTI Adapter comes with a predefined set of reports. These reports can be customized and additional reports can be created by leveraging the imported data.

Viewing Amazon Connect Reports in Salesforce

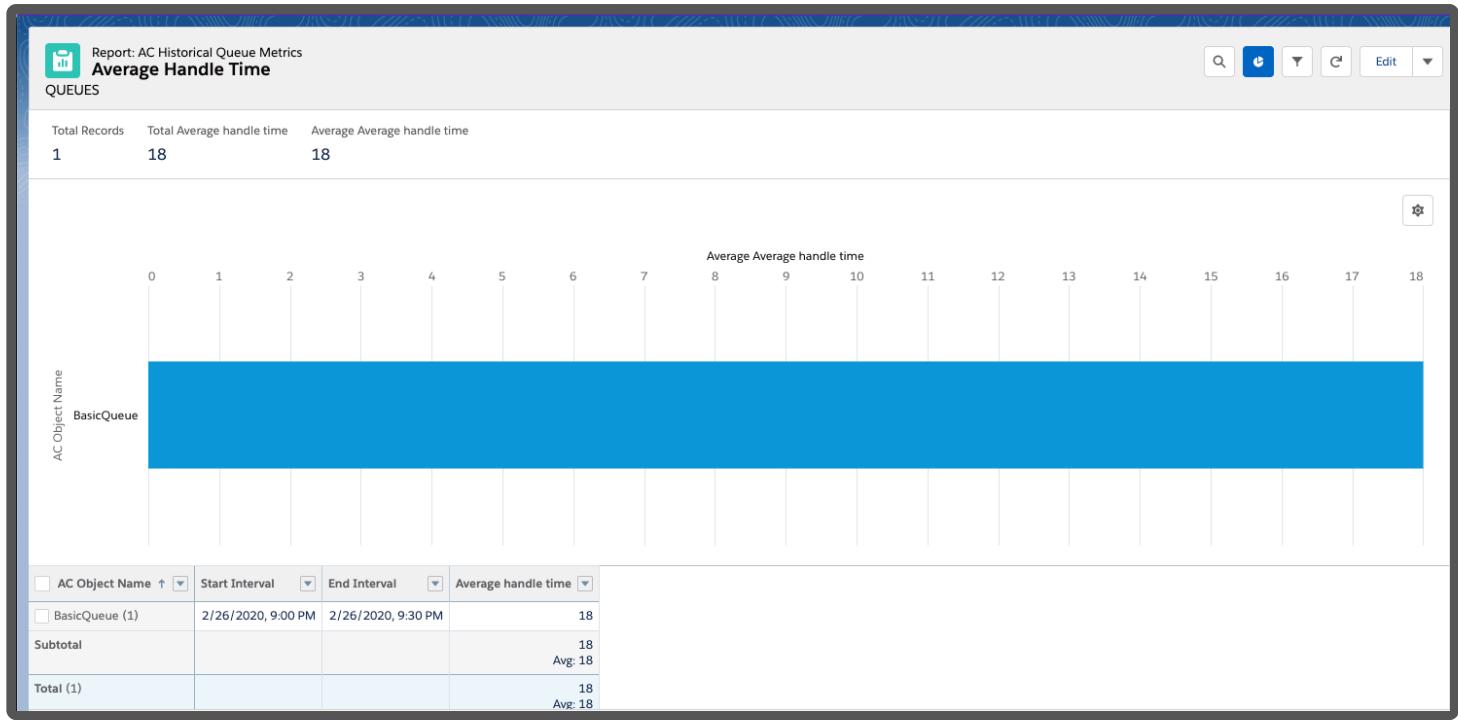
1. Log in into your Salesforce org and go to the **Service Console**
2. Expand the **navigation menu** by selecting the down arrow and choose **Reports**
3. In the left Navigation, select **All Folders**
4. Select the **Amazon Connect Reports** folder

The screenshot shows the Service Console interface. At the top, there's a navigation bar with a grid icon, the text "Service Console", and a dropdown menu labeled "Reports". Below this is a sidebar titled "Reports" with "All Folders" selected. It shows "1 item" under "RECENT". A table lists the item: "Recent" followed by "Amazon Connect Reports", which is highlighted with a red box. Other options like "Created by Me" are also listed.

5. In the list of reports, choose Average Handle Time queue report

The screenshot shows the Reports page with "All Folders > Amazon Connect Reports" selected. There are "30 items" listed. A table displays the reports, with columns for "RECENT", "NAME", "DESCRIPTION", and "FOLDER". The "NAME" column includes links like "Average Queue Abandon Time", "Average Occupancy Today", and "Average Handle Time", which is highlighted with a red box. The "FOLDER" column shows all entries belong to "Amazon Connect Reports".

6. Once the report loads, you should see data (provided calls have queued in this Amazon Connect instance today)



Amazon Connect Real-Time Metrics in Salesforce

The CTI adapter includes real-time reporting tools which provide visibility into critical data which help improve the utilization of your agents and allows insight into overall queue performance. Once you have deployed the AWS Serverless Application Repository for Salesforce your Amazon Connect instance will push real-time metric data to Salesforce every 15 seconds. This data can be viewed from two tools that were included with the CTI Adapter installation.

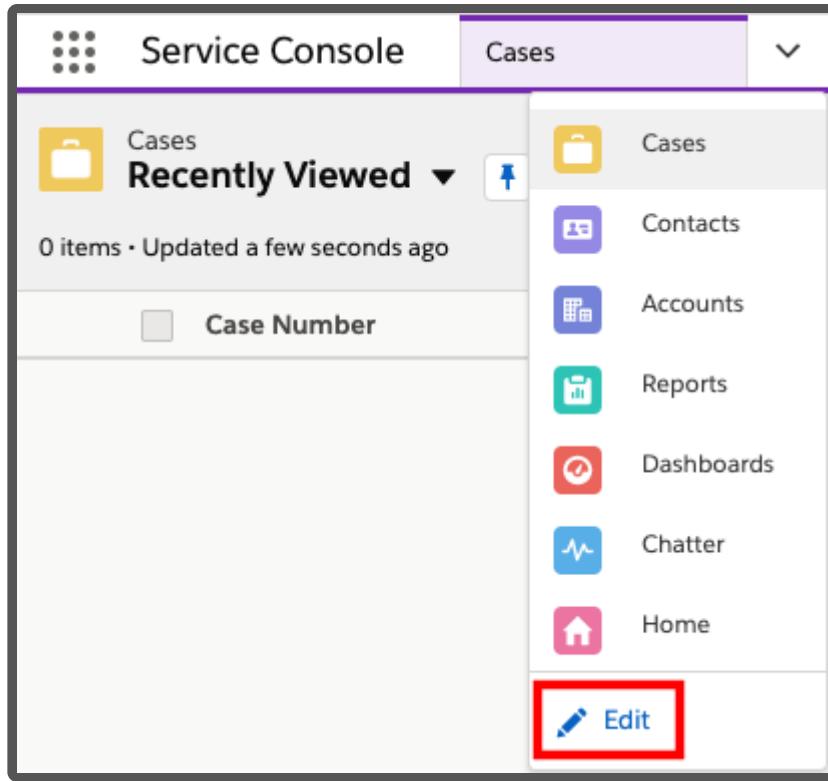
The first view, AC Queue Metrics queue provides details about current queue staffing and the distribution of contacts by queue. The second view, AC Real Time Queue Metrics, allows you to select a specific queue and view the real-time metrics for that queue.

Deployment and Configuration

Once you have deployed the AWS Serverless Application Repository for Salesforce and provided the appropriate credentials, there is no further configuration required to make the data flow work. The only remaining task is to add the real-time views to your Salesforce console.

Adding Real-Time Reports to the Service Console

1. Log in into your Salesforce org and go to the **Service Console**
2. Expand the **navigation menu** by selecting the down arrow and choose **Edit**.



3. On the Edit Service Console App Navigation Items page, select **Add More Items**

Edit Service Console App Navigation Items

Personalize your nav bar for this app. Reorder items, and rename or remove items you've added.
[Learn More](#) ⓘ

NAVIGATION ITEMS (7)	
AC Queue Metrics	+ (highlighted with a red box)
AC Real Time Queue Metrics	+ (highlighted with a red box)
AC Case Metrics	
AC Case History	
AC Case Status	
AC Case Type	
AC Case Work Log	

Add More Items (highlighted with a red box)

4. Select the + next to **AC Queue Metrics** and **AC Real Time Queue Metrics**

5. Select **Add 2 Nav Items**

6. Change the order of your Navigation Items if desired, then choose **Save**

Edit Service Console App Navigation Items

Personalize your nav bar for this app. Reorder items, and rename or remove items you've added.

[Learn More](#) 

 2 items added to your list. Save your updates.

NAVIGATION ITEMS (10)

[Add More Items](#)

-  AC CTI Adapters 
-  Cases
-  Contacts
-  Accounts
-  Reports
-  Dashboards
-  Chatter
-  Home
-  AC Queue Metrics 
-  AC Real Time Queue Metrics 

[Reset Navigation to Default](#) 

[Cancel](#)

[Save](#)

7. Once the save completes, expand the **navigation menu** by selecting the down arrow and choose **AC Queue Metrics**



1 item · Updated a few seconds ago

 CTI Adapters

1

 AC Lightning

8. The AC Queue Metrics view will display and any relevant data will update every 15 seconds.

The screenshot shows the 'AC Queue Metrics' view within the Service Console. At the top, there's a header bar with the Service Console logo and the title 'AC Queue Metrics'. Below the header is a section titled 'Real Time Metrics' with a subtitle 'Live Queue Data'. A table displays the following data for 'BasicQueue':

Queue Name	Agents Available	Agents Error	Agents Non Productive	Agents Online	Agents Staffed	Agents After Contact Work	Contacts In Queue	Contacts Scheduled	Oldest Contact Age
BasicQueue	1	1	0	2	2	0	0	0	0

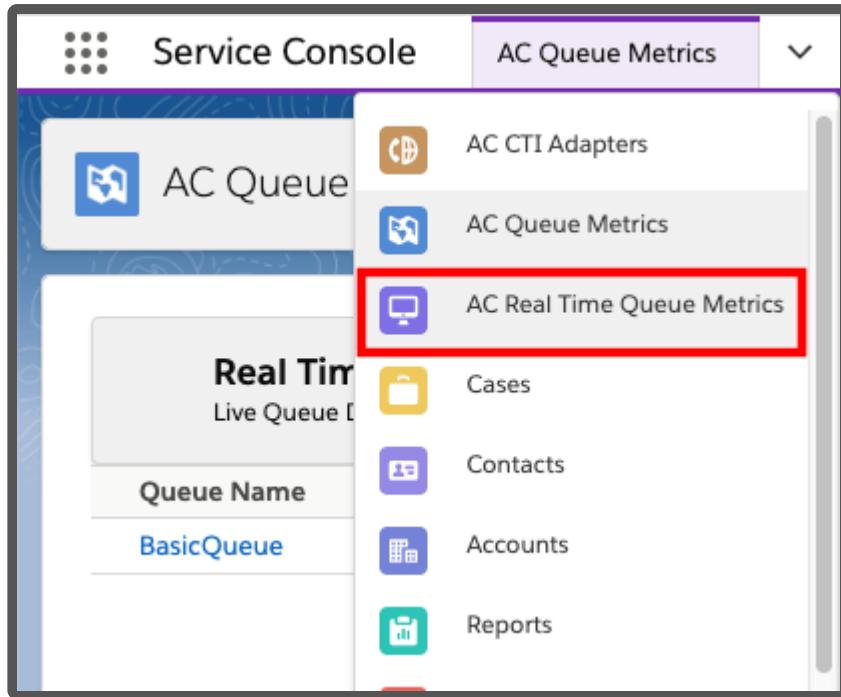
9. Scroll down to view the **AC Contact Metrics Dashboard

The screenshot shows the 'AC Contact Metrics' dashboard. At the top, there's a header bar with the Service Console logo and the title 'AC Contact Metrics'. Below the header are several cards displaying contact metrics for 'BasicQueue':

- Contacts Incoming:** Sum of Contacts Incoming: 4
- Contacts Queued:** Sum of Contacts queued: 4
- Contacts Handled Incoming:** Sum of Contacts handled incoming: 4 (100% of 4)
- Contacts Abandoned:** Sum of Contacts abandoned: 0
- Average Queue Abandon Time:** Sum of Average queue abandon time: 0
- Average Handle Time:** Sum of Average handle time: 18
- Contact Handle Time:** Sum of Contact handle time: 75
- Average Service Level 120 Seconds:** Average Service level 120 seconds: 100%

A tooltip is visible over the 'Contacts Handled Incoming' card, showing the value '4 (100% of 4)'.

10. Expand the navigation menu by selecting the down arrow and choose **AC Real Time Queue Metrics



11. Change the List View to **ALL**

The screenshot shows the "AC Real Time Queue Metrics" list view. At the top, there's a header with a monitor icon, the title "AC Real Time Queue Metrics", a dropdown menu set to "All", and a search icon. Below the header, a table has one item listed: "1 item · LIST VIEWS". Under "LIST VIEWS", there are two options: "All" (which is checked) and "Recently Viewed (Pinned list)".

12. Select a queue to view the detailed real-time statistics for that specific queue

The screenshot shows the Service Console interface with the title bar "Service Console" and "AC Real Time Queue Met...". The main content area displays the "BasicQueue" metric for the "BasicQueue" queue. The "Details" tab is selected, showing various performance metrics:

Metric	Value
Queue Name	BasicQueue
Queue ARN	
Agents After Contact Work	0
Agents Available	0
Agents Error	1
Agents Non Productive	0
Agents OnCall	0
Agents Online	1
Queue Id	3caa8bb5-9426-4b58-8bae-f405b6360cbe
Created By	apiuser, 2/24/2020, 4:51 PM
Owner	apiuser
Agents Staffed	1
Contacts In Queue	0
Contacts Scheduled	0
Oldest Contact Age	0
Last Modified By	apiuser, 2/26/2020, 9:38 PM

Contact Channel Analytics

In addition to the CTI adapter's native ability to provide direct playback links to call recordings in Amazon Connect, the AWS Serverless Application Repository for Salesforce includes several functions that allow you to process recordings, perform quality analytics functions, and bring data into Salesforce.

This processing is done post-call, using the Contact Trace Record (CTR) as the initiation path. The following quality analytics options are available:

- **Call Recording Streaming:** streams the actual audio file into Salesforce. This option is not mandatory for the others to function.
- **Recording Transcript:** you can choose to have your call recordings transcribed to text and presented in a visual format that resembles a chat conversation. This allows for quick scanning of a call to identify key segments of conversation. This option is required if you wish to include the next level of analysis
- **AI-Driven Contact Analysis:** once the recordings have been transcribed to text, you can also indicate that you wish to do further analysis of the conversation using [Amazon Comprehend](#).

Available options are:

- **Sentiment Analysis:** returns the overall sentiment of the conversation (Positive, Negative, Neutral, or Mixed).
- **Keyphrase Extraction:** returns the key phrases or talking points and a confidence score to support that this is a key phrase.
- **Language Detection:** returns the dominant language with a confidence score to support that a language is dominant
- **Custom Entities:** allows you to customize the AI to identify terms that are specific to your domain
- **Syntax Analysis:** analyze the transcript using tokenization and Parts of Speech (PoS), and identify word boundaries and labels like nouns and adjectives within the text.

If you would like to set up streaming with Contact Lens, please finish the [Call Recording Streaming](#) section below and then follow the [Contact Lens Streaming](#) instructions and possibly the [Post Call Contact Lens Import](#) instructions.

Call Recording Streaming

You can stream Call Recordings in your Salesforce Org. This allows for easy access to the recordings from within Salesforce and can be used in conjunction with the other contact channel analytics features to provide a complete view of the customer interaction.

The import of call recordings is not required to activate the other contact channel analytics features.

Once enabled during the AWS Serverless Application Repository for Salesforce, recording import is activated on a call by call basis by adding a specific contact attribute. This attribute is used during Contact Trace Record processing to trigger the call import.

NOTE: After Call Work time is a part of the Contact Trace Record. As such, CTRs are not generated until the agent leaves the after call work state. If you are not seeing a recording import, please make sure the agent has completed the call and left the after call work state.

Cloudformation Template

To make sure that the AWS resources are set up, make sure that the *PostcallRecordingImportEnabled* parameter is set to true in your Cloudformation stack:

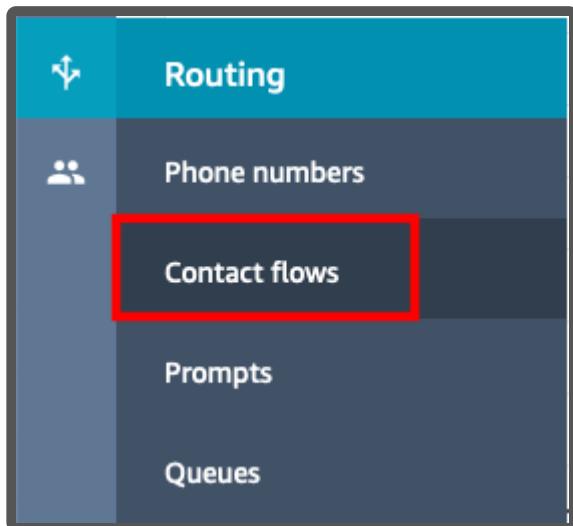
The screenshot shows the AWS CloudFormation console with the 'Parameters' tab selected. A list of parameters is displayed, and one specific parameter, 'PostcallRecordingImportEnabled', is highlighted with a red box. The value for this parameter is 'true'.

Key	Value
AmazonConnectInstanceId	
AmazonConnectQueueMaxRecords	
AmazonConnectQueueMetricsMaxRecords	
CTREventSourceMappingMaximumRetryAttempts	
CTRKinisisARN	
ConnectRecordingS3BucketName	
ConnectReportingS3BucketName	
HistoricalReportingImportEnabled	
LambdaLoggingLevel	
PostcallCTRImportEnabled	true
PostcallRecordingImportEnabled	

Note: If you are expecting more than 1000 concurrent calls, you may have to increase the timeout for the `sfCTRTrigger` lambda.

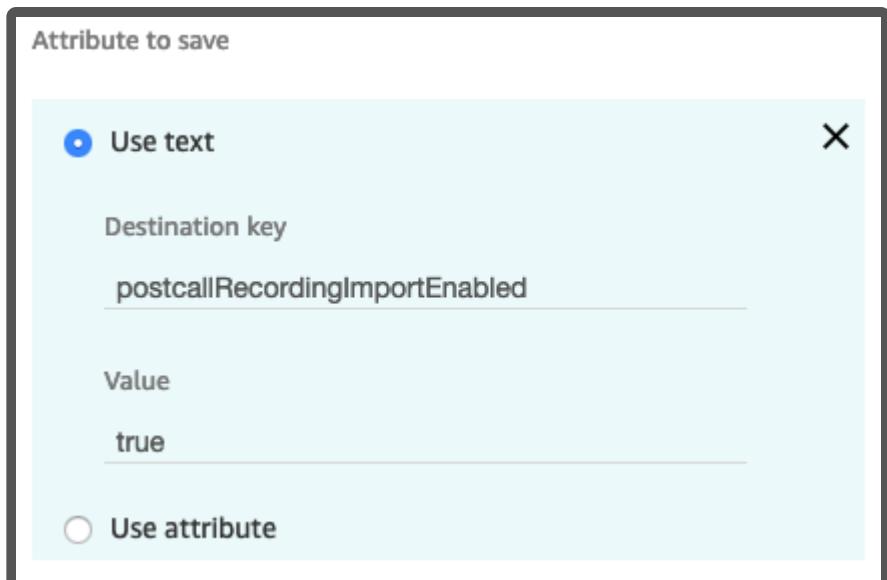
Enabling call recording streaming

1. Login to your Amazon Connect instance as an Administrator
2. From the left navigation, choose **Routing** then select **Contact flows**



3. Open the contact flow that you want to use to enable call recording import. This contact flow must have Amazon Connect's native recording turned on.
4. In your contact flow, before you transfer to queue, add a new **Set contact attributes** block
5. Configure the block to set a contact attribute as follows:
 - a. **Destination key:** postcallRecordingImportEnabled

b. **Value:** true



6. Save the Set contact attributes block. Make sure it is appropriately connected to your contact flow, and **Publish** the flow.
7. Wait approximately 2 minutes to give the contact flow time to publish.
8. Place a call, connect to your agent, speak for a few moments to test the audio, then end the call. Make sure the agent exits after call work
9. After a minute or so, a new Contact Channel Analytics record should be imported, and when opening it, you should be able to stream the audio. (See section [Adding Contact Channel Analytics to the Service Console](#). below).

Adding users to the AC_CallRecording permission set

This step is only necessary for non admin user accounts for the non contact lens case.

1. In the setup search box, search for "Permission sets". Select the "AC_CallRecording" permission set. Select "Manage Assignments".

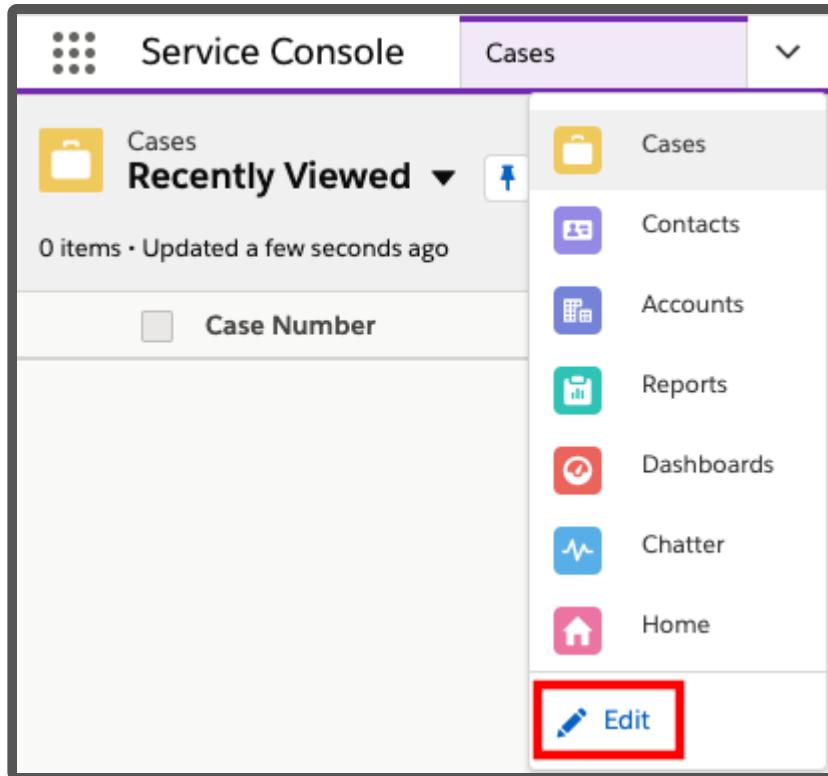
The screenshot shows the Salesforce Setup interface. On the left, there's a sidebar with 'Users' and 'Custom Code' sections, and a search bar at the top. The main area is titled 'Permission Sets' under 'SETUP'. It shows a permission set named 'AC_CallRecording' with fields for 'Description', 'License', 'Session Activation Required', and 'Last Modified By'. Below this is a section titled 'Apps' with 'Assigned Apps' and 'Assigned Connected Apps' settings. The 'Manage Assignments' button in the top right is highlighted with a red box.

2. Select "Add Assignments". Add the users that should have access to the audio recordings and select "assign".

The screenshot shows the 'Assign Users' page under 'Permission Sets'. It lists 'All Users' with a 'View' dropdown set to 'All Users'. The main area is a table where users can be assigned to a permission set. The 'Action' column contains checkboxes for each user, and the 'Full Name' column lists their names. The 'Assign' button in the top right of the table is highlighted with a red box.

Adding Contact Channel Analytics to the Service Console

1. Log in into your Salesforce org and go to the **Service Console**
2. Expand the **navigation menu** by selecting the down arrow and choose **Edit**.



3. On the Edit Service Console App Navigation Items page, select **Add More Items**

Edit Service Console App Navigation Items

Personalize your nav bar for this app. Reorder items, and rename or remove items you've added.
[Learn More](#) ⓘ

NAVIGATION ITEMS (7)	Add More Items
AC Contact Channel Analytics	
Chatter	
Home	
Accounts	
Reports	
Contacts	
Cases	

4. Select the + next to **AC Contact Channel Analytics**

5. Select **Add 1 Nav Item**

6. Change the order of your Navigation Items if desired, then choose **Save**

Edit Service Console App Navigation Items

Personalize your nav bar for this app. Reorder items, and rename or remove items you've added.

[Learn More](#) i

i 1 item added to your list. Save your updates.

NAVIGATION ITEMS (11)

[Add More Items](#)

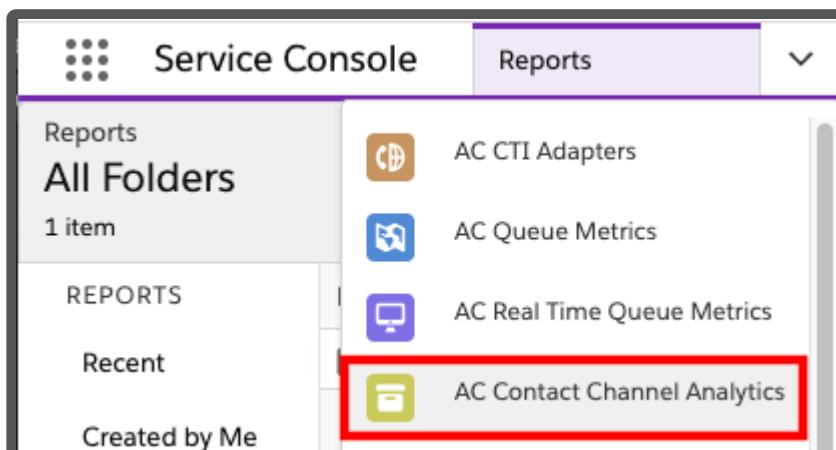
- ≡  AC CTI Adapters X
- ≡  AC Queue Metrics X
- ≡  AC Real Time Queue Metrics X
- ≡  Cases
- ≡  Contacts
- ≡  Accounts
- ≡  Reports
- ≡  Dashboards
- ≡  Chatter
- ≡  Home
- ≡  AC Contact Channel Analytics X

[Reset Navigation to Default](#) i

[Cancel](#)

[Save](#)

7. Once the save completes, expand the **navigation menu** by selecting the down arrow and choose **AC Contact Channel Analytics**



8. Change the list view from Recently Viewed to All

AC Contact Channel Analytics
Recently Viewed ▾

0 items LIST VIEWS

All

✓ Recently Viewed (Pinned list)

9. Once the view refreshes, you should see your record(s)

AC Contact Channel Analytics
All

1 item • Sorted by Contact Channel Analytics Name • Filtered by all ac contact channel analytics • Updated a few seconds ago

	Contact Channel Analytics Name ↑
1	CCA 000001

10. Select the recording to open it

11. In the top right, you will see a button to stream the recording.

AC Contact Channel Analytics
CCA 000022

Fields

General

Recording

▶ 0:00 / 0:24

12. NOTE: The recording playback, waveform, and transcript views are only active when you also choose to activate recording transcripts.

Recording Transcripts

Enabling the Recording Transcripts activates a process to run your contact recordings through Amazon Transcribe which uses a deep learning process to convert text to speech accurately and quickly. In addition, this process also creates a visual waveform of the recording, enables the in-app recording playback, and provides a visual representation of the conversation.

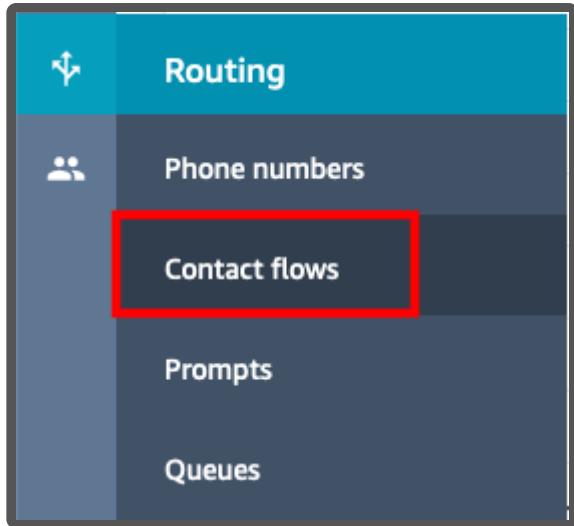
Once enabled during the AWS Serverless Application Repository for Salesforce, recording transcription is activated on a call by call basis by adding a specific contact attribute. This attribute is used during Contact Trace Record processing to trigger the transcription.

Make sure the Salesforce user accessing recording transcription are added to the AC_CallRecording permission set, as described in the previous section.

Enabling recording transcription

1. Login to your Amazon Connect instance as an Administrator

2. From the left navigation, choose **Routing** then select **Contact flows**



3. Open the contact flow that you want to use to enable call transcription. This contact flow must have Amazon Connect's native recording turned on, since the transcription is dependent on it.

4. In your contact flow, before you transfer to queue, add a new **Set contact attributes** block

5. Configure the block to set two contact attributes as follows:

i. Attribute 1: enables the transcription process

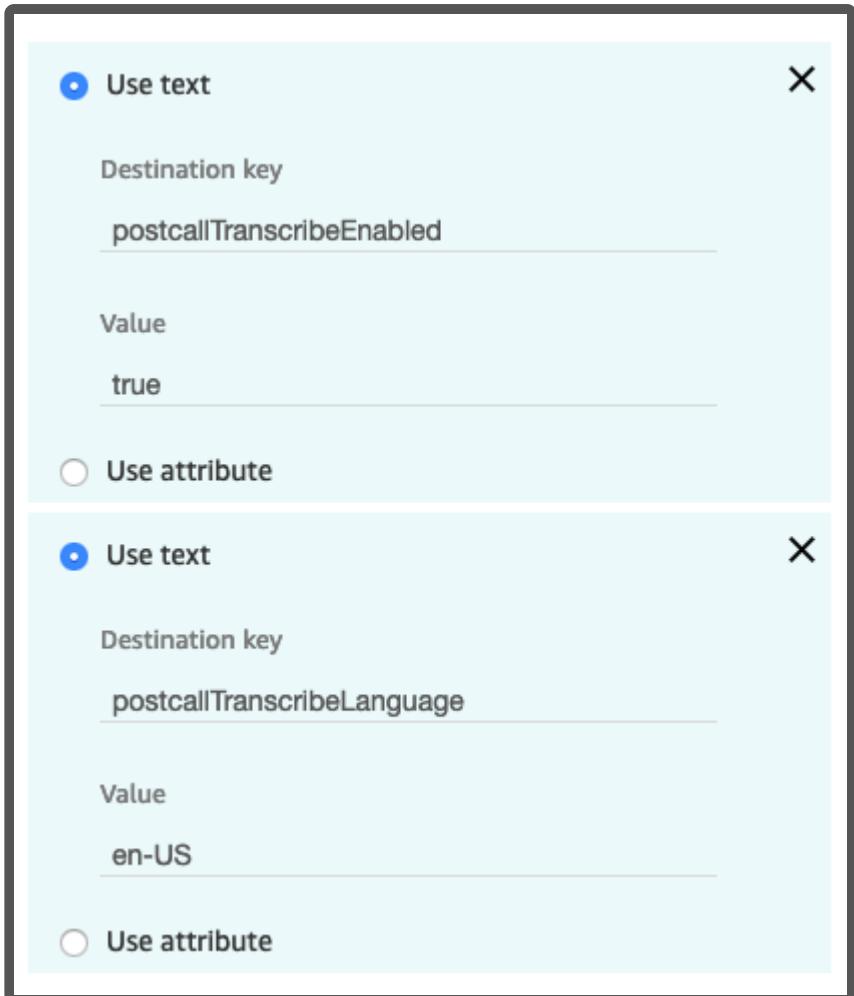
a. **Destination key:** postcallTranscribeEnabled

b. **Value:** true

ii. Attribute 2: specifies the transcription language

-a. **Destination key:** postcallTranscribeLanguage

b. **Value:** en-US (See [Amazon Transcribe API Reference](#) for valid language codes)



6. **Save** the Set contact attributes block. Make sure it is appropriately connected to your contact flow, and **Publish** the flow.
7. Wait approximately 2 minutes to give the contact flow time to publish.
8. Place a call, connect to your agent, speak for a few moments from both the agent and the customer side to generate a good transcript, then end the call. Make sure the agent exits after call work
9. The transcription will take at least as long as the call did. Wait an appropriate amount of time for the transcription to be available.

Accessing transcriptions

1. Log in into your Salesforce org and go to the **Service Console**
2. Expand the **navigation menu** by selecting the down arrow and choose AC Contact Channel Analytics. If you have not previously added AC Contact Channel Analytics to the navigation menu, complete the steps found in [Adding Contact Channel Analytics to the Service Console](#).

**Quarterly Performance**

CLOSED \$0

OPEN 0

500k

400k

300k



Home



Omni Supervisor



Reports



AC CTI Adapters



AC Contact Channel Analytics



AC Contact Trace Records

3. Change the list view from Recently Viewed to All



AC Contact Channel Analytics

Recently Viewed ▼

0 items

LIST VIEWS

All

Recently Viewed (Pinned list)

4. Once the view refreshes, you should see your record(s)



AC Contact Channel Analytics

Recently Viewed ▼

3 items · Updated a few seconds ago



Contact Channel Analytics Name



Contact Id

1	<input type="checkbox"/> CCA 000002	6df455ce-8e7e-4ee8-806d-b5dff9758d66
2	<input type="checkbox"/> CCA 000001	c3a70eeb-4a9e-4605-8871-4bd0d58c9b51
3	<input type="checkbox"/> CCA 000000	a14b0510-2db7-441c-aac2-55018eb4cbde

5. Select a record to view the details.

6. Once the record opens, note the recording, and the visual version of the transcription

The screenshot shows a call recording interface. At the top, there's a 'Recording' section with a play button, a progress bar indicating 0:00 / 0:24, and a volume icon. Below this is a 'Transcript' section. The transcript starts with a 'Contact Started' entry from the agent. A customer message 'is the test.' is shown with a timestamp of Customer • 2.16 • 2.42. The agent responds with 'See if the transcript' and a timestamp of Agent • 1.88 • 2.17. The customer continues with 'And see if the transcript work' at 3.49 • 3.69, and the agent replies with 'I'm contact?' at 3.39 • 3.63. Finally, the customer says 'from Contact line.' at 5.69 • 6. The entire transcript is contained within a light gray box.

7. Also note that the transcriptions for each side of the conversation are also included as attachments.

AI Driven Contact Analysis

Enabling the AI Driven Contact Analysis function allows you to process the transcribed text using [Amazon Comprehend](#). Amazon Comprehend is a natural language processing service that uses machine learning to find insights and relationships in text.

Once enabled during the AWS Serverless Application Repository for Salesforce, contact analysis is activated on a call by call basis by adding a specific contact attribute. This attribute is used during Contact Trace Record processing to trigger the Amazon Comprehend task.

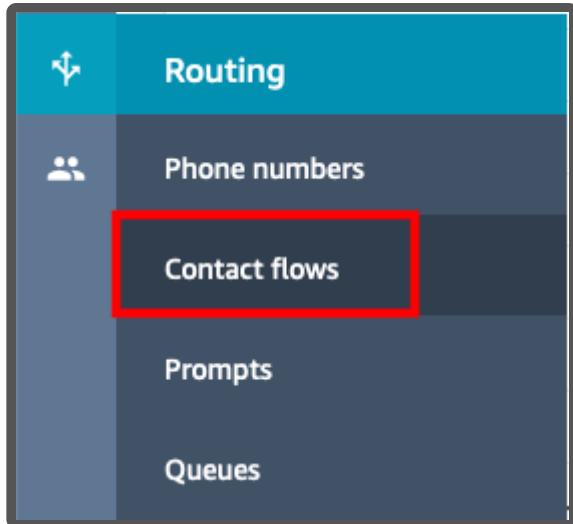
There are five functions available with the integration. Each function is triggered by a code. You can use one code in your contact attribute, or string multiple together as a comma separated list. The available codes and their functions are:

- **snt = Sentiment Analysis**
- **kw = Keyphrase Extraction**
- **dl = Language Detection**
- **ne = Custom Entities**
- **syn = Syntax Analysis**

Enabling AI Driven Contact Analysis

1. Login to your Amazon Connect instance as an Administrator

2. From the left navigation, choose **Routing** then select **Contact flows**



3. Open the contact flow that you want to use to enable AI Driven Contact Analytics. This contact flow must have Amazon Connect's native recording turned on, and transcription enabled as these are both prerequisites for the analytics function.

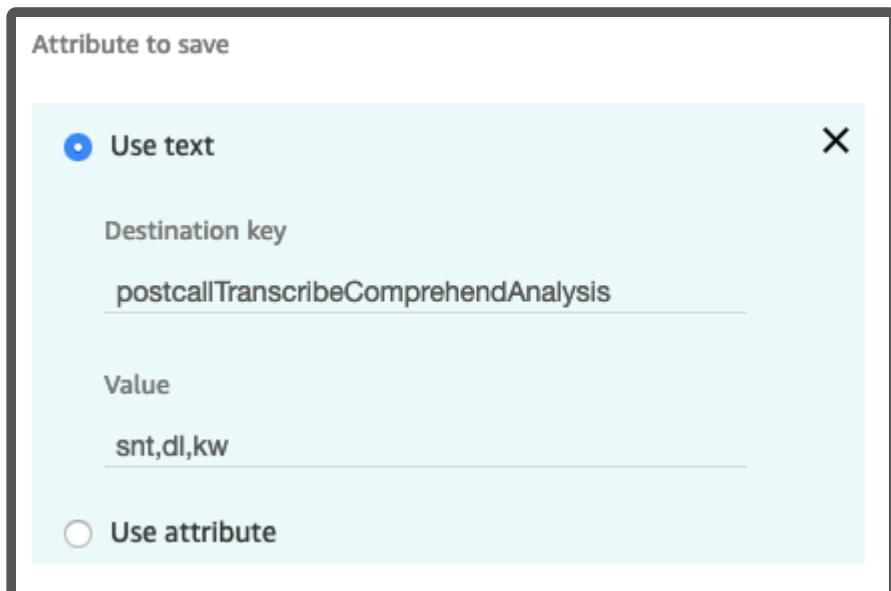
4. In your contact flow, before you transfer to queue, add a new **Set contact attributes** block

5. Configure the block to set a contact attribute as follows:

a. **Destination key:** postcallTranscribeComprehendAnalysis

b. **Value:** snt,dl,kw,syn

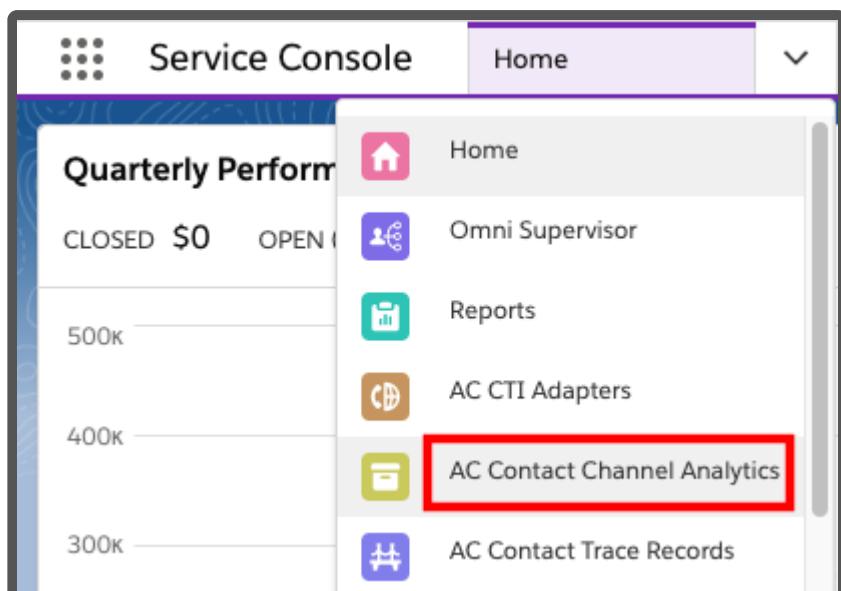
- In this example, we are performing sentiment analysis, language detection, and keyphrase extraction



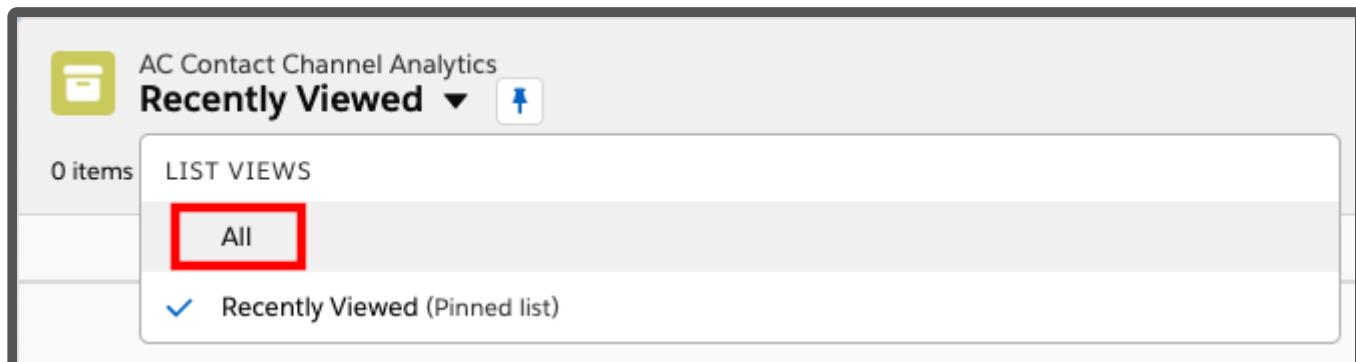
6. **Save** the Set contact attributes block. Make sure it is appropriately connected to your contact flow, and **Publish** the flow.
7. Wait approximately 2 minutes to give the contact flow time to publish.
8. Place a call, connect to your agent, speak for a few moments from both the agent and the customer side to generate a good transcript, then end the call. Make sure the agent exits after call work
9. The contact analysis runs after the transcription, which will take at least as long as the call did. Wait an appropriate amount of time for the analysis to be available.

Accessing the AI Driven Contact Analysis

1. Log in into your Salesforce org and go to the **Service Console**
2. Expand the **navigation menu** by selecting the down arrow and choose AC Contact Channel Analytics. If you have not previously added AC Contact Channel Analytics to the navigation menu, complete the steps found in [Adding Contact Channel Analytics to the Service Console](#).



3. Change the list view from Recently Viewed to **All**



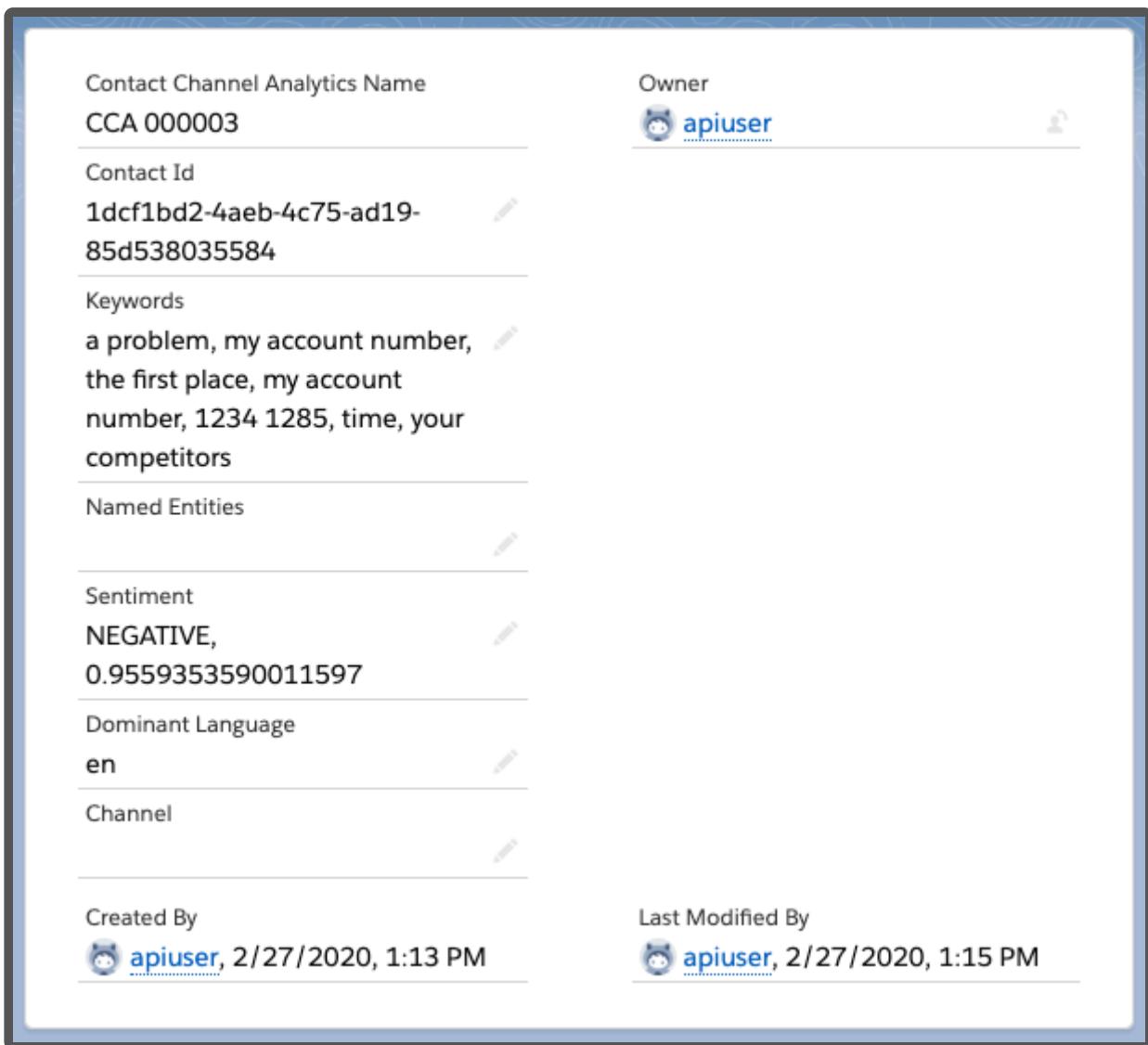
4. Once the view refreshes, you should see your record(s)

3 items · Updated a few seconds ago

	<input type="checkbox"/> Contact Channel Analytics Name	Contact Id
1	<input type="checkbox"/> CCA 000002	6df455ce-8e7e-4ee8-806d-b5dff9758d66
2	<input type="checkbox"/> CCA 000001	c3a70eeb-4a9e-4605-8871-4bd0d58c9b51
3	<input type="checkbox"/> CCA 000000	a14b0510-2db7-441c-aac2-55018eb4cbde

5. Select a record to view the details.

6. Once the record opens, note the Keywords, Sentiment, and Dominant Language



The screenshot shows a contact record for "CCA 000003". The record includes the following fields:

- Contact Channel Analytics Name:** CCA 000003
- Contact Id:** 1dcf1bd2-4aeb-4c75-ad19-85d538035584
- Owner:** apiver
- Keywords:** a problem, my account number, the first place, my account number, 1234 1285, time, your competitors
- Named Entities:** (empty)
- Sentiment:** NEGATIVE, 0.9559353590011597
- Dominant Language:** en
- Channel:** (empty)
- Created By:** apiver, 2/27/2020, 1:13 PM
- Last Modified By:** apiver, 2/27/2020, 1:15 PM

Contact Trace Record Import

In Amazon Connect, data about contacts is captured in contact trace records (CTR). This data can include the amount of time a contact spends in each state: customer on hold, customer in queue, agent interaction time. The basis for most historical and real-time metrics in Amazon Connect is the

data in the CTR. When you create metrics reports, the values displayed for **most** (not all) metrics in the report are calculated using the data in the CTRs.

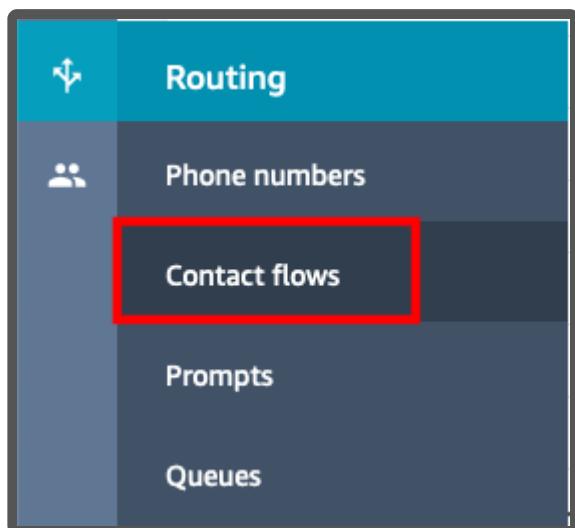
CTRs are available within your Amazon Connect instance for 24 months from the time when the associated contact was initiated. You can also stream CTRs to Amazon Kinesis to retain the data longer, and perform advanced analysis on it. Additionally, with the AWS Serverless Application Repository for Salesforce, you can import Contact Trace Records into your Salesforce org.

Contact Trace Record Import

Once enabled during the AWS Serverless Application Repository for Salesforce, CTR import is activated on a call by call basis by adding a specific contact attribute. This attribute is used during Contact Trace Record processing to trigger the import task.

Enabling Contact Trace Record Import

1. Login to your Amazon Connect instance as an Administrator
2. From the left navigation, choose **Routing** then select **Contact flows**



3. Open the contact flow that you want to use to enable call recording import.
4. In your contact flow, before you transfer to queue, add a new **Set contact attributes** block
5. Configure the block to set a contact attribute as follows:
 - a. **Destination key:** postcallCTRImportEnabled
 - b. **Value:** true

Attribute to save

Use text X

Destination key
postcallCTRImportEnabled

Value
true

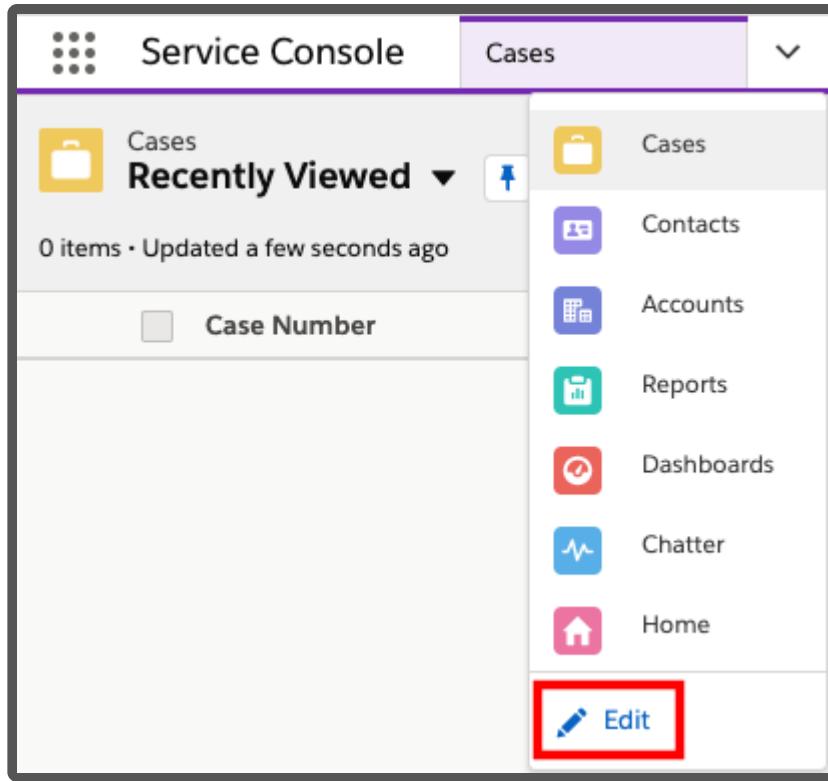
Use attribute

6. Save the Set contact attributes block. Make sure it is appropriately connected to your contact flow, and **Publish** the flow.
7. Wait approximately 2 minutes to give the contact flow time to publish.
8. Place a call, connect to your agent, speak for a few moments, then end the call. Make sure the agent exits after call work
9. The Contact Trace Record is emitted shortly after call completion and the import happens almost immediately.

Note: If you are expecting more than 1000 concurrent calls, you may have to increase the timeout for the `sfCTRTrigger` lambda.

Adding Contact Trace Records to the Service Console

1. Log in into your Salesforce org and go to the **Service Console**
2. Expand the **navigation menu** by selecting the down arrow and choose **Edit**.



3. On the Edit Service Console App Navigation Items page, select **Add More Items**

Edit Service Console App Navigation Items

Personalize your nav bar for this app. Reorder items, and rename or remove items you've added.
[Learn More](#) ⓘ

NAVIGATION ITEMS (7)	Add More Items
AC Contact Trace Records	
Case Number	
Chatter	
Dashboard	
Home	
Recent Activity	

4. Select the + next to **AC Contact Trace Records**

5. Select **Add 1 Nav Item**

6. Change the order of your Navigation Items if desired, then choose **Save**

Edit Service Console App Navigation Items

Personalize your nav bar for this app. Reorder items, and rename or remove items you've added.

[Learn More](#) 

 1 item added to your list. Save your updates.

NAVIGATION ITEMS (12)

[Add More Items](#)

  AC CTI Adapters	
  AC Queue Metrics	
  AC Real Time Queue Metrics	
  AC Contact Channel Analytics	
  Cases	
  Contacts	
  Accounts	
  Reports	
  Dashboards	
  Chatter	
  Home	
  AC Contact Trace Records	

[Reset Navigation to Default](#) 

[Cancel](#)

[Save](#)

7. Once the save completes, expand the **navigation menu** by selecting the down arrow and choose **AC Contact Trace Records**

The screenshot shows the Service Console interface. On the left, there's a sidebar with a chart titled "Quarterly Performance" showing "CLOSED \$1,820,000". On the right, a list of metrics is displayed, each with an icon and a name: AC CTI Adapters, AC Queue Metrics, AC Real Time Queue Metrics, AC Contact Channel Analytics, AC Contact Trace Records (which is highlighted with a red box), and Cases.

8. Change the list view from Recently Viewed to All

The screenshot shows the "AC Contact Trace Records" list view. At the top, there's a header with a filter icon and the text "AC Contact Trace Records". Below it is a dropdown menu labeled "Recently Viewed ▾" with a blue arrow icon. The menu has two options: "All" (which is highlighted with a red box) and "Recently Viewed (Pinned list)".

9. Once the view refreshes, you should see your record(s)

The screenshot shows the "AC Contact Trace Records" list view after refreshing. The header now says "All" with a blue arrow icon. Below it, the text "5 items · Sorted by Contact Trace Record · Filtered by all ac contact trace records · Updated a few seconds ago" is displayed. The main area shows a table with 5 rows, each containing a checkbox, a number (1-5), and a Contact Trace Record ID (e.g., CTR 0000000000). The first row is expanded to show the full record ID.

		Contact Trace Record ↑
1	<input type="checkbox"/>	CTR 0000000000
2	<input type="checkbox"/>	CTR 0000000001
3	<input type="checkbox"/>	CTR 0000000002
4	<input type="checkbox"/>	CTR 0000000003
5	<input type="checkbox"/>	CTR 0000000004

10. Select a record to view it

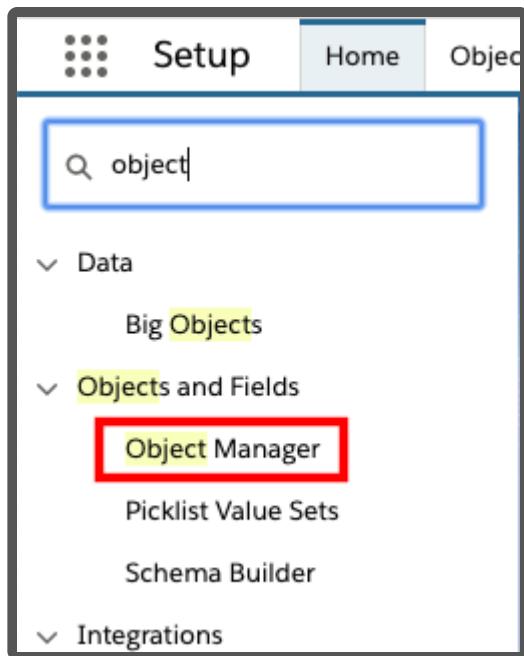
11. Note the ContactId value from Amazon Connect

Display Additional Contact Trace Record Data

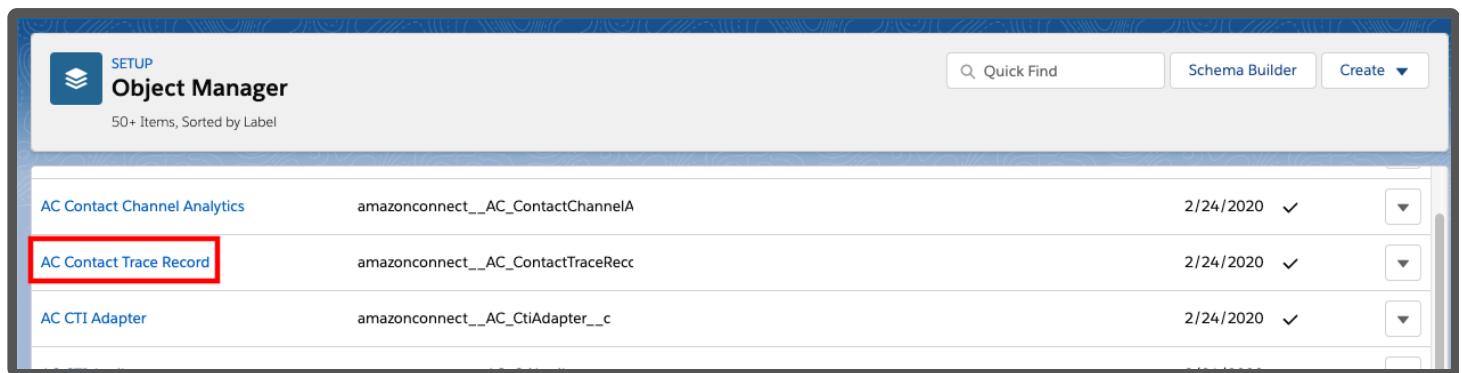
By default, the AC Contact Trace Record layout only contains the ContactId. However, all of the CTR data has been imported. It is likely that you will want to customize this view to show more data.

Customizing the AC Contact Trace Record Layout

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** field, enter object and choose **Object Manager** from the results



3. In the Object Manager, find the **AC Contact Trace Record** object and select it



4. In the left navigation, choose **Page Layouts**

5. Select **AC Contract Trace Record Layout**

6. Select items from the Fields section and add them to the layout as you wish. In the example below, I have selected Agent Username, Queue Name, Queue Duration, After Contact Work Duration, Agent Interaction Duration, and Attributes

AC Contact Trace Record Detail

Standard Buttons: [Edit](#) [Delete](#) [Clone](#) [Change Owner](#) [Change Record Type](#) [Printable View](#) [Sharing](#) Custom Buttons

Information (Header visible on edit only)

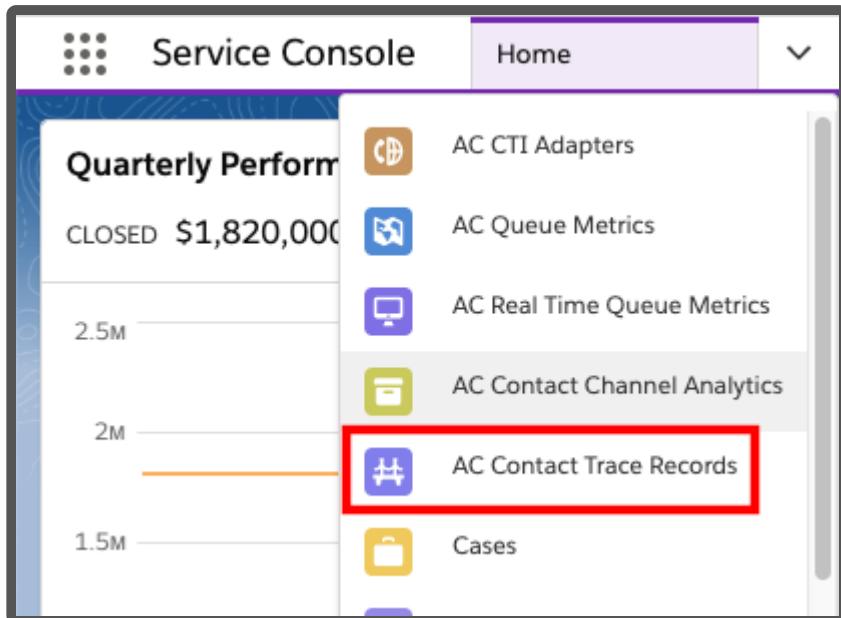
Contact Trace Record	GEN-2004-001234	Owner	Sample Text
* Channel	Sample Text	Agent Username	Sample Text
* ContactId	Sample Text	Queue Name	Sample Text
After Contact Work Duration	76,916	Queue Duration	18,140
Agent Interaction Duration	37,408	Attributes	Sample Text

7. Save the layout

8. Return to the **Service Console**

9. Refresh the browser

10. Expand the **navigation menu** by selecting the down arrow and choose **AC Contact Trace Records**



11. Select a contact trace record

12. You should now see your modified layout



Related **Details**

Contact Trace Record

CTR 000000003

Channel

VOICE

ContactId

71662532-8da9-41bf-bba1-3755ed070cdd

After Contact Work Duration

2

Agent Interaction Duration

10

Owner

apiouser

Agent Username

doug [REDACTED] pm

Queue Name

BasicQueue

Queue Duration

24

Attributes

```
{"phone_number": "+17048076561",  
 "postal_code": "98121",  
 "postcallCTRImportEnabled": "true",  
 "postcallRecordingImportEnabled": "true",  
 "postcallTranscribeEnabled": "true",  
 "postcallTranscribeLanguage": "en-US"}
```

Created By

apiouser, 2/27/2020, 10:38 AM

Last Modified By

apiouser, 2/27/2020, 10:38 AM

Postcall Contact Lens Import

Contact Lens for Amazon Connect is a set of machine learning (ML) capabilities integrated into Amazon Connect. With Contact Lens for Amazon Connect, contact center supervisors can better understand the sentiment, trends, and compliance of customer conversations to effectively train agents, replicate successful interactions, and identify crucial company and product feedback.

Contact Lens are available within your Amazon Connect instance in CTR page, and Contact Lens data are stored in Amazon Connect S3 bucket. With the AWS Serverless Application for Salesforce (Amazon Connect Salesforce Lambda), you can import Contact Lens data into your Salesforce org.

Contact Lens Import

Before using AWS Serverless Application (Amazon Connect Salesforce Lambda) to import Contact Lens data, you need to enable Contact Lens in Amazon Connect. More information can be found at <https://docs.aws.amazon.com/connect/latest/adminguide/enable-analytics.html>.

Once enabled during the installation of AWS Serverless Application (Amazon Connect Salesforce Lambda), Contact Lens import is activated on a call by call basis by adding a specific contact attribute. This attribute is used during Contact Lens processing to trigger the import task.

Creating the AWS Lambda Trigger for the Contact Lens Data

1. Make sure you set **ContactLensImportEnabled** to **true** during the deployment of Amazon Connect Salesforce Lambda application.
2. Once the deployment is finished, you need to configure a trigger that invokes a Lambda function when Contact Lens output file is generated and stored in S3.
3. In a browser tab, login to the [AWS Console](#).
4. Open the [AWS Lambda Console](#).
5. In the filter field of the AWS Lambda console, enter sfProcessContactLens and press enter to filter the list of functions.
6. Select the Lambda that includes sfProcessContactLens in the name.
7. Expand the Designer section.
8. Select Add trigger

The screenshot shows the AWS Lambda Designer interface. At the top, there are three tabs: Configuration (highlighted in orange), Permissions, and Monitoring. Below the tabs, a section titled "Designer" is expanded, indicated by a downward arrow. A link "Go back to application serverlessrepo-AmazonConnectSalesforceLambda" is visible. On the right side, a Lambda function card is displayed with the following details:

- Icon: Lambda symbol
- Name: serverlessrepo-Amazo nConnectS-sfProcessC ontactLens-X7O29Q1 V175S
- Layers: (0)

At the bottom left, a button labeled "+ Add trigger" is highlighted with a red box.

9. In Trigger configuration, select S3 from the dropdown list

Add trigger

Trigger configuration

Select a trigger



DynamoDB

aws database nosql



Kinesis

analytics aws streaming



S3

aws storage

S3



SNS

aws messaging notifications pub-sub push



SQS

aws queue

10. Select the bucket of your Amazon Connect instance. You can find your Amazon Connect bucket name by clicking on your Amazon Connect instance alias in Amazon Connect console.
11. Change the Event type to PUT.
12. Set the Prefix to **Analysis/Voice/2020**. Note that this might change as the date changes so you will need to update this on the first day of every new year.
13. Set the Suffix to .json
14. The trigger configuration should now be similar to the following:

Add trigger

Trigger configuration



S3

aws storage

Bucket

Please select the S3 bucket that serves as the event source. The bucket must be in the same region as the function.

connect-[REDACTED]



Event type

Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual events cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.

PUT



Prefix - optional

Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters.

Analysis/Voice/2020

Suffix - optional

Enter a single optional suffix to limit the notifications to objects with keys that end with matching characters.

.json

Lambda will add the necessary permissions for Amazon S3 to invoke your Lambda function from this trigger. [Learn more](#) about the Lambda permissions model.

The Lambda console no longer supports disabling S3 and CloudWatch Logs triggers. Delete these triggers to stop further actions.



Recursive invocation

If your function writes objects to an S3 bucket, ensure that you are using different S3 buckets for input and output. Writing to the same bucket increases the risk of creating a recursive invocation, which can result in increased Lambda usage and increased costs. [Learn more](#)

I acknowledge that using the same S3 bucket for both input and output is not recommended and that this configuration can cause recursive invocations, increased Lambda usage, and increased costs.

[Cancel](#)[Add](#)

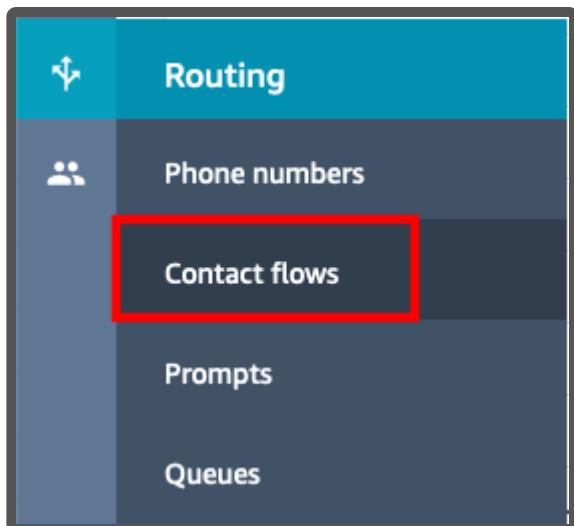
15. Select **Add**

16. If everything has been configured correctly, you should received a success message.

Enabling Contact Lens Import

1. Login to your Amazon Connect instance as an Administrator

2. From the left navigation, choose **Routing** then select **Contact flows**

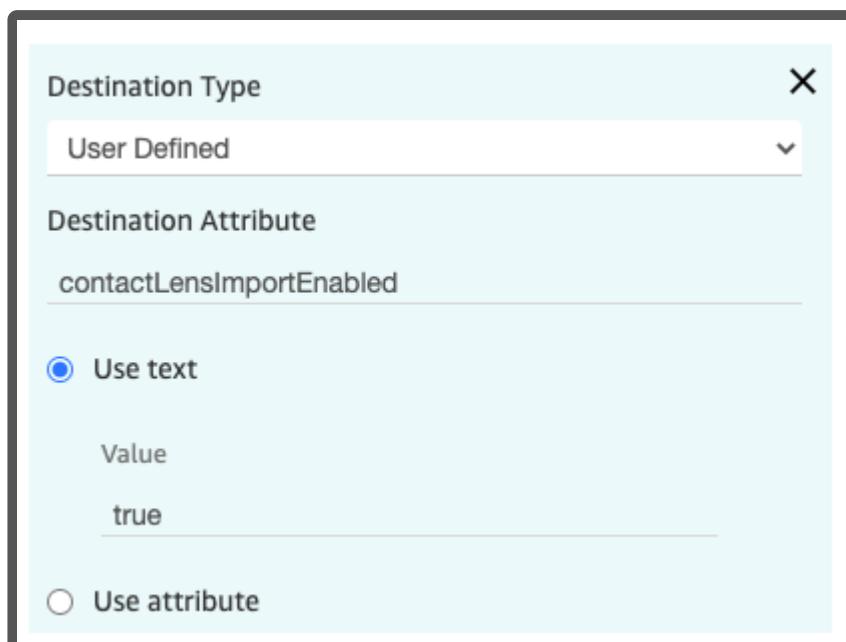


3. Open the contact flow that you want to use to enable Contact Lens import.

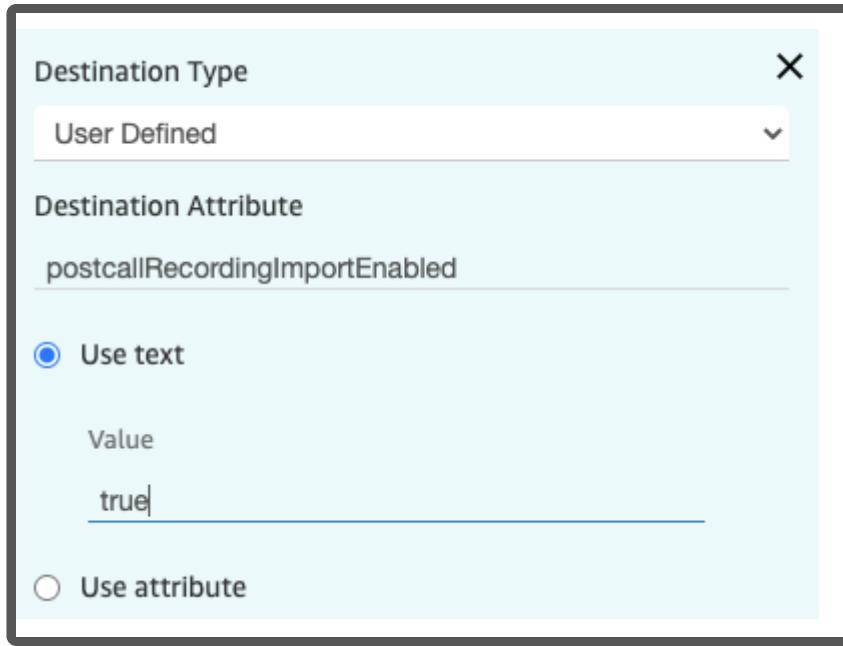
4. In your contact flow, before you transfer to queue, add a new **Set contact attributes** block

5. Configure the block to set a few contact attributes:

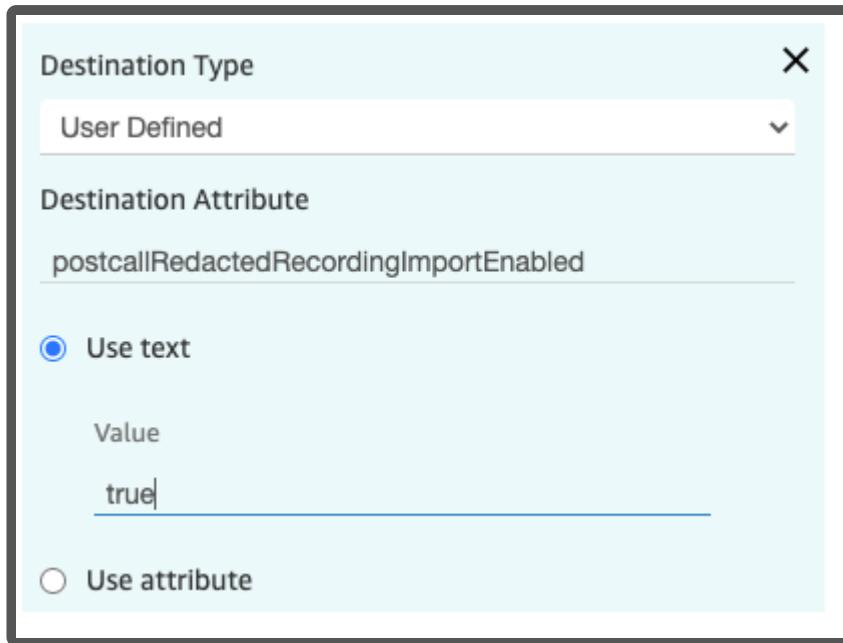
- To turn on Contact Lens data import, set ***contactLensImportEnabled*** to **true**.



- For recording import, there are two options: original call recording and redacted call recording. **Note that you can only import one of the recordings for each contact.**
 - To turn on original recording import, set ***postcallRecordingImportEnabled*** to **true**



- To turn on redacted recording import, set ***postcallRedactedRecordingImportEnabled*** to **true**



- Save the Set contact attributes block. Make sure it is appropriately connected to your contact flow, and **Publish** the flow.
- Wait approximately 2 minutes to give the contact flow time to publish.
- Place a call, connect to your agent, speak for a few moments, then end the call. Make sure the agent exits after call work
- The Contact Lens data is emitted a couple of minutes after call completion and the import happens almost immediately.

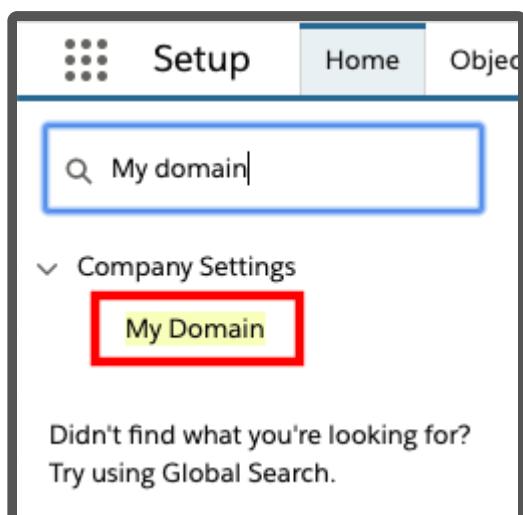
Configuring My Domain in Salesforce

The latest CTI adapter includes several lighting components that provide a better administrative user experience. Salesforce requires that My Domain be enabled to make use of lightning components. Setting up My Domain is a fairly simple setup, but it does require some time for the changes to propagate, so it will be helpful to complete this configuration in advance of your CTI adapter deployment.

Register Your Domain

Step 1 in the process is registering your domain in Salesforce. This allows you to check availability of the domain and complete the registration process. It will take some amount of time for the registration to complete.

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** field, enter **My Domain**, then select **My Domain** from the result list



3. In the **My Domain Step 1** section, enter your desired domain name and select **Check Availability** to determine if the domain is available.

A screenshot of the "Choose Your Domain Name" step in the My Domain configuration wizard. The page has a heading "Choose Your Domain Name" and instructions: "Enter a domain name and check whether it's available. Be sure of your name before registering. Only Salesforce Customer Support can change your domain name once it's registered." It also states: "Your domain name can be up to 34 characters. It can include letters, numbers, and hyphens; but it can't start or end with a hyphen." A text input field contains the URL "https://sfseorgb-dev-ed.my.salesforce.com/", and a "Check Availability" button below it is highlighted with a red rectangular border.

4. If the domain is not available, you will need to try a different name.

5. If the domain is available, select **Register Domain

The screenshot shows a web browser with the URL <https://sfseorgb-dev-ed.my.salesforce.com/>. At the top, there is a 'Check Availability' button and a green circular icon with a checkmark and the word 'Available'. Below these, a red box highlights the 'Register Domain' button. A tooltip next to it says: 'After you click Register Domain, Salesforce takes a few minutes to update its naming registries. You receive an email when it's done.'

6. The domain registration process will begin. You will receive an email once it is complete. Once you receive the confirmation, you may continue with the next section.

The screenshot shows the 'My Domain Step 2' page. It features a title 'Step 2 Domain Registration Pending' above a flowchart with four blue rounded rectangles connected by arrows. The steps are: 'Choose Domain Name' → 'Domain Registration Pending' (with a clock icon) → 'Domain Ready for Testing' → 'Domain Deployed to Users'. Below the flowchart, a message says 'Your domain name is sfseorgb-dev-ed.my.salesforce.com'. At the bottom, a note says 'Registering your domain. You'll receive an email when it's ready for testing.'

Deploy the Domain to Your Users

Once the domain registration process completes, you then need to deploy the domain to your users.

1. Log in into your Salesforce org and go to **Setup**

2. In the **Quick Find** field, enter **My Domain**, then select **My Domain** from the result list

The screenshot shows the Salesforce Setup interface. At the top, there is a navigation bar with 'Setup', 'Home', and 'Objects'. Below it is a search bar with the text 'My domain'. Under the 'Company Settings' section, there is a list with a red box around the 'My Domain' item. At the bottom, there is a message: 'Didn't find what you're looking for? Try using Global Search.'

3. In the **My Domain Step 2** section, note the domain name, then select the **Log in** button to login using the new domain.

Your domain name is **sfseorgb-dev-ed.my.salesforce.com**

Your domain name is ready. Log in to test it out.

Log in

To test your new domain, click tabs and links. If you've customized the UI, check for hard links to your original URL.

4. Once the login completes, you should see your new domain in the address bar of your browser.
You should also be returned to the My Domain configuration.

5. Select the Deploy to Users button to deploy your domain

Your domain name is **sfseorgb-dev-ed.my.salesforce.com**

Your domain name is ready. Log in to test it out.

Log in

To test your new domain, click tabs and links. If you've customized the UI, check for hard links to your original URL.

Deploy to Users Roll out the new domain to your org. i

6. You should get a popup message that warns you about the domain deployment. Select OK.

...edded page at sfseorgb-dev-ed.my.salesforce.com says

When you deploy the new domain, we activate it immediately. Only Salesforce Customer Support can disable or change your domain name once it's deployed.

Cancel

OK

7. Deployment should now be complete

Configure Salesforce Omnichannel for Testing

In order to sync your Connect User status with your Omni-Channel agent status, you must configure Omni-Channel Presence Syncing. This will make your Omni-Channel presence status match your Amazon Connect Agent Status and vice versa.

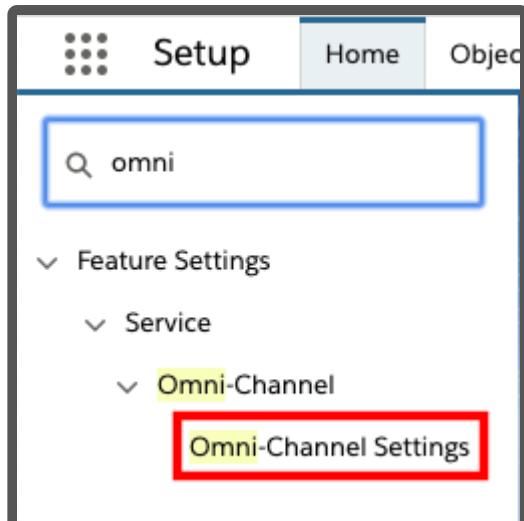
Enable Omnichannel

First, we must enable omni-channel. Once you enable Omni-Channel, you will have access to the other components in Salesforce that will be required for Omni-Channel setup.

Enable Omnichannel in Your Salesforce Org

1. Log in into your Salesforce org and go to **Setup**

2. In the **Quick Find** field, enter omni and choose **Omni-Channel Settings** from the results



3. Select the checkbox for Enable Omni-Channel and choose Save

A screenshot of the 'Omni-Channel Settings' page. The title is 'Omni-Channel Settings'. A descriptive text states: 'Omni-Channel routes work items to your support agents. It sets agent capacity for accepting work and agent availability.' Below this, there are several configuration options: 'Enable Omni-Channel' (checkbox checked), 'Enable Skills-Based Routing' (checkbox unchecked), 'Enable Secondary Routing Priority' (checkbox unchecked), and 'Display a login confirmation upon loading a console with Omni-Channel' (checkbox unchecked). At the bottom right are 'Save' and 'Cancel' buttons.

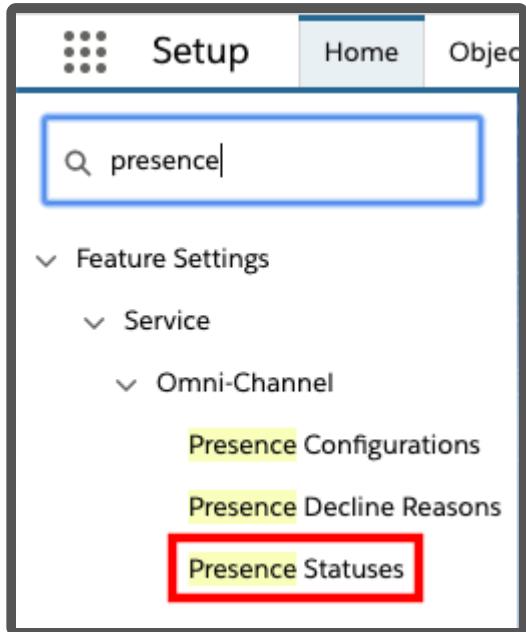
4. Omni-Channel is now enabled.

Configure Presence Statuses

Once you have enabled Omni-Channel, you will need to configure presence statuses to reflect the different presence states that you wish your Omni-Channel agents to enter. These do not need to match agent statuses in Amazon Connect exactly, but it does make it easier to track what you are doing.

Add a Presence Status

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** field, enter presence and choose **Presence Statuses** from the results



3. In the Presence Statuses page, choose New
4. Provide a status name, for example Lunch
5. Set the Status options appropriately, for example, Busy
6. For Online statuses, you will need to provide a channel. Please reference the [Omni-Channel documentation](#) for details
7. Choose Save

Presence Statuses

Let agents indicate when they're online and available to receive work items from a specific service channel, or whether they're away or offline.

The screenshot shows the 'Basic Information' section of a presence status configuration. It includes fields for 'Status Name' (Lunch) and 'Developer Name' (Lunch), both with red borders. Below this is a 'Status Options' section with a dropdown menu set to 'Busy'. At the bottom are 'Save' and 'Cancel' buttons.

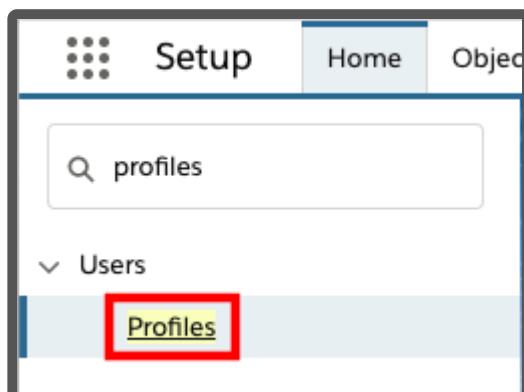
8. Repeat as necessary for all desired statuses

Configure Profiles to Use the New Statuses

Before agents can use the statuses that have been configured, you will need to make sure that they have been provided rights to them. This is done by modifying the profiles assigned to your agents.

Modify Profiles to Use New Statuses

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** field, enter profiles and choose **Profiles** from the results



3. Select the profile assigned to your users
4. Hover over the Enabled Service Presence Status link and choose Edit



Profile

System Administrator

Help for this Page

Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information.

If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more record types available to users with this profile.

[Login IP Ranges \[0\]](#) | [Enabled Apex Class Access \[0\]](#) | [Enabled Visualforce Page Access \[21\]](#) | [Enabled External Data Access \[0\]](#) | [Enabled Network Credential Access \[0\]](#) | [Enabled Custom Metadata Type Access \[0\]](#) |
[Enabled Custom Setting Definitions Access \[0\]](#) | [Enabled Flow Access \[0\]](#) | [Enabled Service Presence Status Access \[0\]](#) | [Enabled Custom Permissions \[0\]](#)

Enabled Service Presence Status Access

Edit

Enabled Service Presence Status Access Help

No Service Presence Status enabled

5. Select the available status from the left, then choose the Add button to add it to the Enabled Service Presence Statuses field

Enable Service Presence Status Access

[Save](#) [Cancel](#)

Available Service Presence Statuses

--None--

Enabled Service Presence Statuses

Lunch


Add 
Remove

6. Select Save

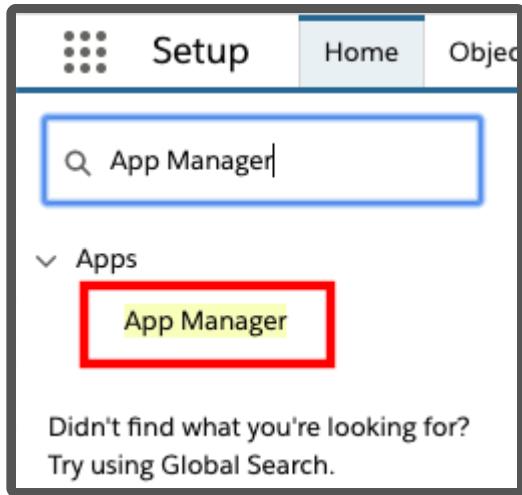
7. Repeat as necessary for other statuses or profiles.

Add Omni-Channel to the Utility Bar

To provide agents access to the Omni-Channel tool, you will need to add it to the Service Console.

Add the Omni-Channel Utility Item

1. Log in into your Salesforce org and go to **Setup**
2. In the **Quick Find** box, type **App Manager**, then choose **App Manager** from the result list.



3. Expand the drop-down menu associated to Service Console and select **Edit**.

12	Salesforce Chatter	Chatter	The Salesforce Chatter social network, including profiles and feeds	1/21/2020, 8:46 PM	Classic	✓	▼
13	Service	Service	Manage customer service with accounts, contacts, cases, and more	1/21/2020, 8:46 PM	Classic	✓	▼
14	Service Console	LightningService	(Lightning Experience) Lets support agents work with multiple re...	1/21/2020, 8:46 PM	Lightning	✓	▼
15	Site.com	Sites	Build pixel-perfect, data-rich websites using the drag-and-drop Sit...	1/21/2020, 8:46 PM	Classic	Edit	

4. Once the **Lightning App Builder** opens, select **Utility Items (Desktop Only)** from the left Navigation



App Settings

App Details & Branding

App Options

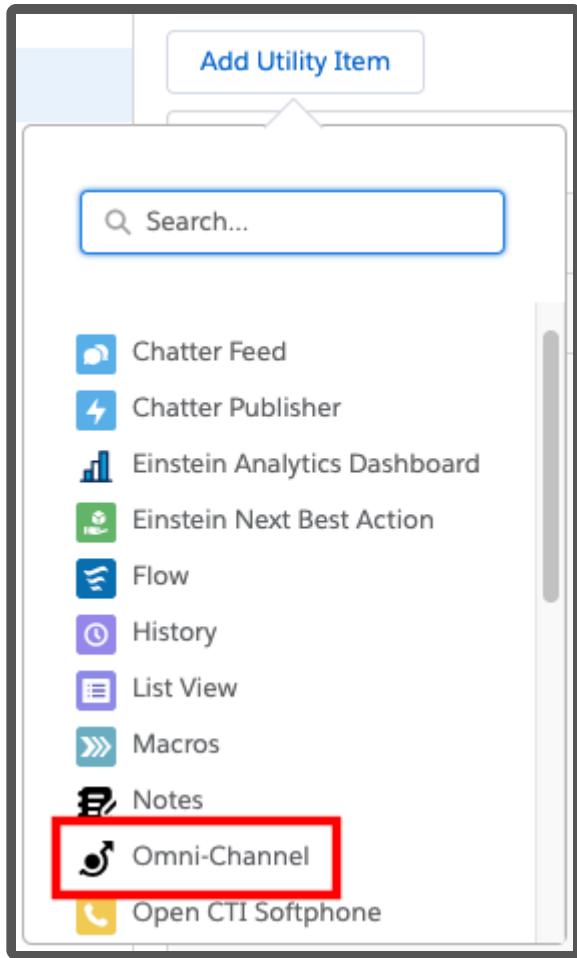
Utility Items (Desktop Only)

Navigation Items

Navigation Rules

User Profiles

5. Choose Add Utility Item, then select Omni-Channel



6. Adjust the order of the utility items as desired and select Save.

7. Return to the Service Console and refresh your browser.

8. You should now see the Omni-Channel utility item.



Appendix B: Configuring Salesforce as Your Identity Provider

Prerequisites

To complete the SSO integration between Salesforce and Amazon Connect, you need:

1. An Amazon Connect Instance configured for SAML authentication

2. Appropriate AWS permissions to create Identity and Access Management (IAM) roles and policies
3. Administrator permissions for your Salesforce Org
4. Amazon Connect CTI Adapter AppExchange package installed and configured

Configuring Salesforce as an Identity Provider

First, we need to enable Salesforce to act as an identity provider (IdP). An IdP performs end user authentication and provides the credentials to the requesting service provider. In this case, Salesforce server as the IdP and Amazon Connect the service provider, while being embedded in Salesforce.

Setup Identity Provider & Download Metadata

1. Log in into your Salesforce org and go to **Setup**.
2. In the **Quick Find** field, type **Identity Provider**, then select **Identity Provider** from the result list
3. Identity Provider may be enabled by default. If not, choose **Enable Identity Provider**, then select the appropriate certificate and select Save.

Identity Provider

Enable Salesforce.com as an identity provider so you can use single sign-on with other web sites, and define the appropriate service providers whose applications support single sign-on. You can switch to different service providers without having to log in again. [Learn more...](#)

Identity Provider Setup	Enable Identity Provider
Click Enable Identity Provider to enable your Salesforce.com organization as an identity provider.	
Service Providers	Service Providers are now created via Connected Apps. Click here.
Name	Created Date
No Service Providers	

4. Choose **Download Metadata** and save the file to your computer.

Identity Provider

Help for this Page 

Enable Salesforce.com as an identity provider so you can use single sign-on with other web sites, and define the appropriate service providers whose applications support single sign-on. You can switch to different service providers without having to log in again. [Learn more...](#)

Quick Tips

- Certificates and Keys
- About Single Sign-On
- My Domain

Identity Provider Setup [Edit](#) [Disable](#) [Download Certificate](#) [Download Metadata](#) 

▼ Details
Issuer https://ctiadapterdemo-dev-ed.my.salesforce.com

▼ Currently chosen certificate details

Label	SelfSignedCert_17Feb2020_221125	Unique Name	SelfSignedCert_17Feb2020_221125
Created Date	2/17/2020, 2:11 PM	Expiration Date	2/17/2021, 4:00 AM
Key Size	2048		

▼ SAML Metadata Discovery Endpoints
Salesforce Identity https://ctiadapterdemo-dev-ed.my.salesforce.com/.well-known/samlidp.xml

Configure the Identity Provider, Policy, and Role in the AWS Console

Next, you need to configure the identity provider (Salesforce) in the AWS console and provide access to Amazon Connect via IAM policies and roles. This allows AWS to acknowledge Salesforce as the identity provider and to provide users authenticated through Salesforce with the access required to login to Amazon Connect.

Configure the Identity Provider

1. Login to the [AWS console](#)
2. Open the [AWS identity and Access Management \(IAM\) Console](#)
3. Select **Identity providers**

Identity and Access Management (IAM)

Dashboard

▼ Access management

- Groups
- Users
- Roles
- Policies
- Identity providers** 
- Account settings

4. Choose **Add Provider**

5. On the Configure Provider screen, select **SAML** as the Provider Type

Add an Identity provider

Configure provider

Provider type

SAML

Establish trust between your AWS account and a SAML 2.0 compatible Identity Provider such as Shibboleth or Active Directory Federation Services.

OpenID Connect

Establish trust between your AWS account and Identity Provider services, such as Google or Salesforce.

6. Set the Provider Name to **SalesforceConnect**

7. Import the metadata file you downloaded previously by selecting Choose File and navigating to the downloaded metadata file.

8. Select Next Step

9. Choose Create

10. The Identity provider has been created

Create the IAM Role and Policy

1. Login to the [AWS console](#)

2. Open the [AWS identity and Access Management \(IAM\) Console](#)

3. Select **Roles**, then choose **Create role**

4. Choose **SAML 2.0 federation**

5. In the SAML provider dropdown, select the provider you just created, which should be named **SalesforceConnect**

6. Select the radio button for **Allow programmatic and AWS Management Console access**. The Attribute and Value fields should auto-populate

Create role

1 2 3 4

Select type of trusted entity

 AWS service EC2, Lambda and others	 Another AWS account Belonging to you or 3rd party	 Web identity Cognito or any OpenID provider	 SAML 2.0 federation Your corporate directory
---	--	--	---

Allows users that are federated with SAML 2.0 to assume this role to perform actions in your account. [Learn more](#)

Choose a SAML 2.0 provider

If you're creating a role for API access, choose an Attribute and then type a Value to include in the role. This restricts access to users with the specified attributes.

SAML provider ▼

[Create new provider](#) [Refresh](#)

Allow programmatic access only
 Allow programmatic and AWS Management Console access

Attribute ▼

Value*

Condition [+ Add condition \(optional\)](#)

7. Select Next: Permissions

8. On the Attach permissions policies page, select **Create policy**. This will open a new browser tab.
9. Choose the **JSON** tab to switch to the JSON editor
10. Replace the existing JSON with the following:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "Statement1",  
      "Effect": "Allow",  
      "Action": "connect:GetFederationToken",  
      "Resource": [  
        "*YOUR ARN**/user/${aws:userid}"  
      ]  
    }  
  ]}
```

11. Replace **YOUR ARN** with the ARN of your Amazon Connect instance. To find your Amazon Connect instance ARN:
12. Open a new tab in your browser and navigate to [Amazon Connect Console](#)
13. Click on the name (alias) of your Amazon Connect instance
14. Copy the Instance ARN and paste it to your computer's notepad (you will use it in a few places)
15. Choose **Review policy**
16. Set the Name to **SalesforceConnectPolicy**
17. Select **Create Policy**
18. Once the Policy has been created, close the tab, go back to the original (Role) tab in your browser and select the **Refresh** button (do not refresh the browser)
19. In the search field, enter **SalesforceConnectPolicy** and select the box to attach the policy.

The screenshot shows the 'Create role' wizard in the AWS IAM console. The current step is 'Attach permissions policies' (step 2). A table lists the attached policy 'SalesforceConnectPolicy'. The policy is filtered by name and is used as 'None'.

Policy name	Used as
SalesforceConnectPolicy	None

20. Choose **Next: Tags** and set tags if desired, then choose **Next: Review**
21. Name the Role **SalesforceConnectRole** and provide a description if you like
22. Select Create role

Complete the Base Salesforce Configuration

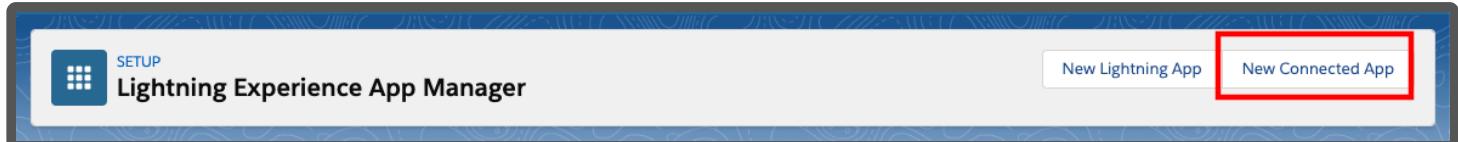
Next, you need to configure a Connect App in Salesforce and provide further configuration to complete the SAML integration.

Create the Connected App in Salesforce

1. Log in into your Salesforce org and go to **Setup**

2. In the **Quick Find** field, type **App Manager**, then select **App Manager** from the result list

3. Select New Connected App



4. Provide a name for the Connected App, such as **AmazonConnectSAML**, then press tab and the API Name should auto-populate

5. Provide an email contact address

A screenshot of the 'New Connected App' configuration page. At the top, it says 'New Connected App' and has 'Save' and 'Cancel' buttons. Below that is a section titled 'Basic Information' with the following fields:

Connected App Name	AmazonConnectSAML
API Name	AmazonConnectSAML
Contact Email	dougjaso+ctiadapterdemo@amazon.com
Contact Phone	[empty]
Logo Image URL	[empty] <small>Upload logo image or Choose one of our sample logos</small>
Icon URL	[empty] <small>Choose one of our sample logos</small>
Info URL	[empty]
Description	[empty]

6. In the Web App Settings section, choose **Enable SAML**

7. Leave Start URL empty

8. Set Entity Id to the same name that you gave the Identity Provider in the IAM console, which should be **SalesforceConnect**

9. Set ACS URL as <https://signin.aws.amazon.com/saml>

10. Set Subject Type as **Persistent ID**

Web App Settings

Start URL	<input type="text"/>
Enable SAML	<input checked="" type="checkbox"/>
Entity Id	<input type="text"/> SalesforceConnect
ACS URL	<input type="text"/> https://signin.aws.amazon.com/saml
Enable Single Logout	<input type="checkbox"/>
Subject Type	Persistent ID
Name ID Format	<input type="text"/> urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified
Issuer	<input type="text"/> https://ctiadapterdemo-dev-ed.my.salesforce.com
IdP Certificate	<input type="text"/> Default IdP Certificate
Verify Request Signatures	<input type="checkbox"/>
Encrypt SAML Response	<input type="checkbox"/>

11. Choose **Save**. The screen should refresh and the new Connected App should be displayed

12. Scroll down to the **Custom Attributes** section and select **New**

13. Set Key as <https://aws.amazon.com/SAML/Attributes/RoleSessionName>

14. Set Value as **\$User.Email**

15. Select **Save**

Create Custom Attribute

Key	<input type="text"/> https://aws.amazon.com
Value	<input type="text"/> <div style="border: 1px solid #ccc; padding: 5px; width: 100%;"> Insert Field Insert Operator ▾ \$User.Email </div>
<input type="button" value="Save"/> <input type="button" value="Cancel"/>	

16. Select New again to configure another custom attribute

17. Set Key as <https://aws.amazon.com/SAML/Attributes/Role>

18. The Value is going to be a combination of the Identity Provider and IAM Role ARNs.

a. In a new tab, open the [AWS identity and Access Management \(IAM\) Console](#)

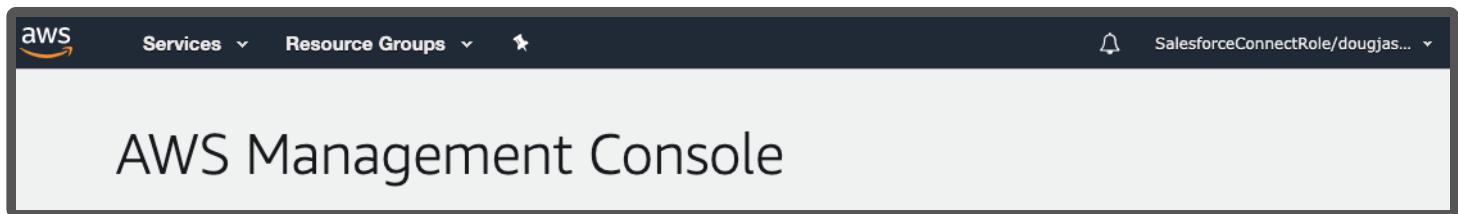
- b. On the left navigation, select **Identity providers**
- c. Select the Identity provider you created earlier, which should be named **SalesforceConnect**
- d. Copy the **Provider ARN** to your computer's notepad
- e. Return to the IAM console and select **Roles**
- f. Select the Role you created earlier, which should be **SalesforceConnectRole**
- g. Copy the **Role ARN** to your computer's notepad
- h. Format the combined value as follows: 'Identity Provider ARN' & ',' & 'Role ARN'
- i. Paste the formatted value into the Custom Attribute Value

19. Select **Save**

The screenshot shows the 'Create Custom Attribute' dialog box. At the top, it says 'Create Custom Attribute'. Below that, there are two fields: 'Key' and 'Value'. The 'Key' field contains 'https://aws.amazon.com'. The 'Value' field contains a text area with the following code:
'arn:aws:iam::YOURACCOUNT:saml-provider/SalesforceConnect' & ',' &
'arn:aws:iam::YOURACCOUNT:role/SalesforceConnectRole'
Below the text area are 'Insert Field' and 'Insert Operator' buttons. At the bottom of the dialog box are 'Save' and 'Cancel' buttons.

20. At the top of the Connected App description, select **Manage**
21. Scroll down to the **SAML login Information** section
22. Copy the **IdP-Initiated Login URL** to your computer's notepad
23. Scroll down to find the Profiles section, then select **Manage Profiles**
24. Select a profile from the list, for example System Administrator for testing purposes
25. Choose **Save**
26. Open a new tab in your browser and navigate to IdP-Initiated Login URL that you copied in an earlier step

27. The browser will redirect to AWS Console and log you in automatically as a federated user
Note: you may be able to see AWS services, but you should have no configuration rights.



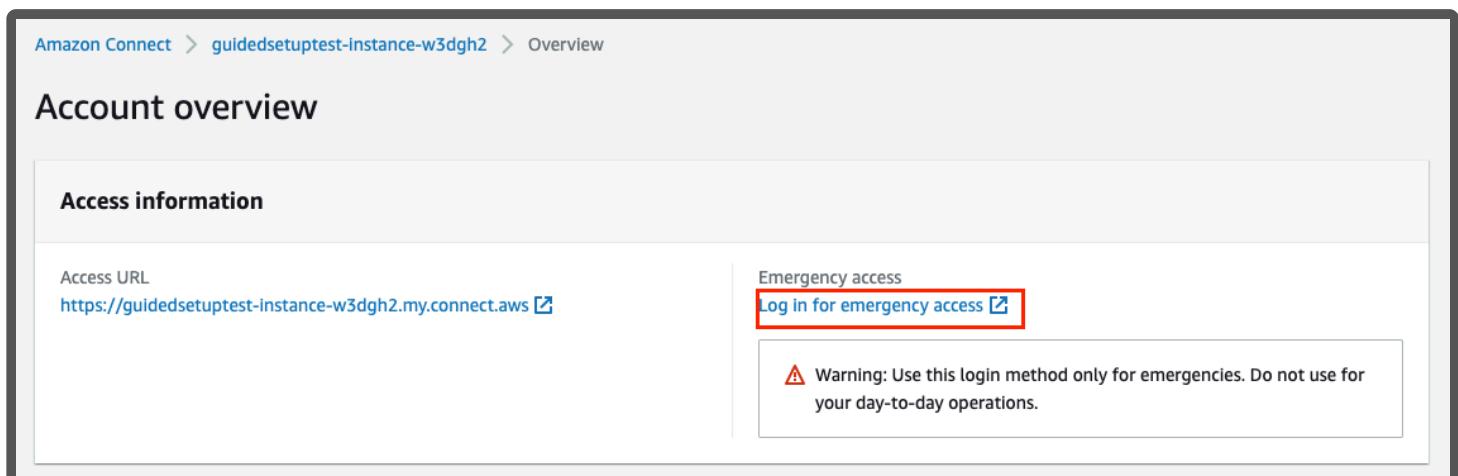
28. The Federated Login consists of the Role name and your Salesforce email address.
29. Initial validation is complete

Complete the Amazon Connect Configuration

The last step in the SAML setup is to add users to Amazon Connect that exist in your Salesforce org, then validate login. It is critical that the usernames for both platforms match exactly.

Add Users to Amazon Connect

1. In a new browser tab, login to the [AWS console](#)
2. Open the [Amazon Connect Console](#)
3. Select the name (alias) of your Amazon Connect instance
4. Choose **Log in for emergency access**



5. Within the Amazon Connect administration portal, select **Users** then choose **User Management**
6. Leave **Create and setup a new user** selected and choose **Next**
7. Complete the First and Last name fields as appropriate

8. Set the login name to match the **Email Address** of your Salesforce user

9. Set the **Routing Profile**. In this example, the default Basic Routing Profile is shown

10. Set the **Security Profile**. In this example, *Admin* is shown

Add new user

1 Select source 2 Add user details 3 Verify user details

First name: Jason, Last name: Douglas, Login name: i+ctladapterdemo@amazon.com

Routing Profile: Basic Routing Profile

Security Profiles: Admin

Phone Type: Soft phone, Auto-Accept Call (unchecked)

After call work (ACW) timeout: 0

11. Select **Save**

12. Select **Create Users**

13. Repeat this process as required for your staff

Final Configuration for the Lightning Experience

Now that all of the underlying pieces are in place, the last steps are to create the Amazon Connect Single Sign On URL and validate that it works correctly, then configure the Lightning CTI adapter and login the agent.

Create the Amazon Connect SSO URL

You create the Amazon Connect SSO URL by combining the IdP-Initiated Login URL that you copied earlier, and a relay state URL that will redirect the authenticated user to your Amazon Connect instance.

The 'RelayState' will be in the following format (replace `us-west-2` with the region you are using):

```
https://us-west-2.console.aws.amazon.com/connect/federate/InstanceId?destination=%2Fconnect%2Fccp
```

1. To begin, format the relay state URL by replacing **InstanceId** with your Instance Id. To find your Amazon Connect Instance Id:
 - a. Open a new tab in your browser and navigate to the [Amazon Connect Console](#)
 - b. Click on the name (alias) of your Amazon Connect
 - c. From the Instance ARN, copy the portion after the '/'. This is the Instance Id

Distribution settings

Instance ARN

arn:aws:connect:us-west-2:YOUR-ACCOUNT-ID:instance/YOUR-INSTANCE-ID-XXX-XXXXXXX

2. Concatenate the 'IdP-Initiated Login URL' and the 'RelayState', by combining the two with "&RelayState=" in between, for example:

```
https://mXXXXXXrun-dev-ed.my.salesforce.com/idp/login?  
app=0sp0N00000Caid&RelayState=https://us-west-  
2.console.aws.amazon.com/connect/federate/InstanceId?  
destination=%2Fconnect%2Fccp
```

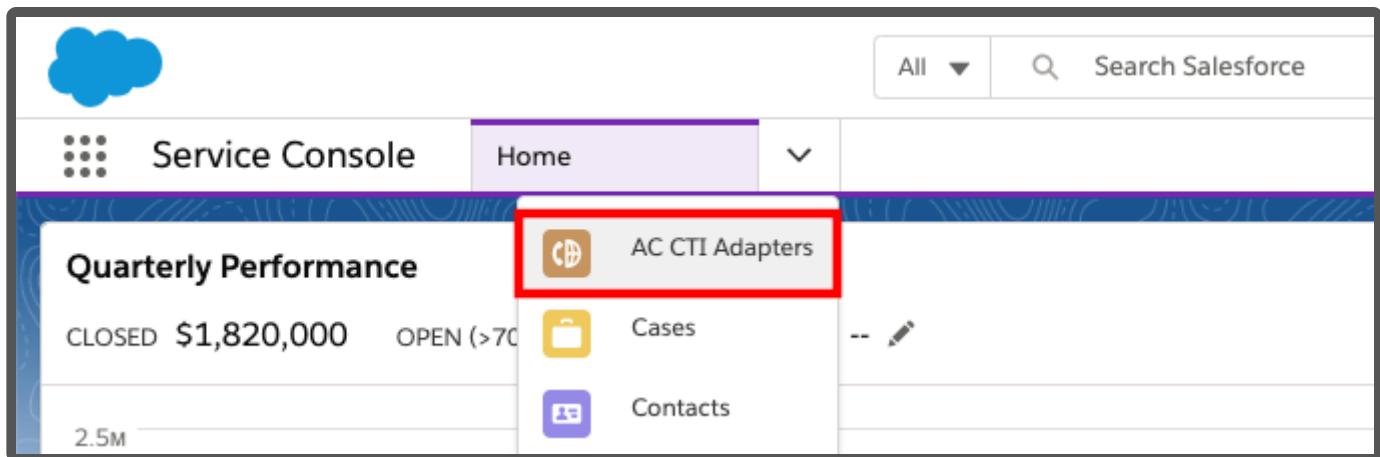
3. This is the Final SSO URL, needed for the Amazon Connect Lightning CTI Adapter Configuration.
4. To validate this URL:
 - a. Open a new tab in the same browser that you are logged into Salesforce
 - b. Paste the fully concatenated URL into the new browser and press enter
 - c. You should automatically login and be redirected to the Amazon Connect Contact Control Panel.
5. Once you validate the full URL, you are ready to add it to the Lightning Adapter

Configure the CTI Lightning Adapter in Salesforce For SSO

Now we are ready to complete the last step in the configuration process: Adding the SSO settings for Salesforce to the Lightning Adapter. This will configure the adapter to authenticate via SSO and redirect to the Amazon Connect Contact Control Panel once authentication completes.

1. Log in into your Salesforce org and go to the **Service Console**

2. Expand the **navigation menu** by selecting the down arrow and choose **AC CTI Adapters**.



3. Select **ACLightningAdapter**

4. Scroll down to the Single SignOn (SSO) section and choose the pencil icon of either field to edit

A screenshot of the 'Single SignOn (SSO)' configuration page. It shows two fields: 'SSO Url' and 'SSO Relay State'. To the right of each field is a small edit icon (pencil). A red box highlights the edit icon for the 'SSO Url' field.

5. For the SSO Url, copy the first part of the SSO URL that you created previously, up to the first question mark (do not copy the question mark), for example:

```
https://mXXXXXrun-dev-ed.my.salesforce.com/idp/login?  
app=0sp0N00000Caid&RelayState=https://us-west-  
2.console.aws.amazon.com/connect/federate/<b>InstanceId</b>?  
destination=%2Fconnect%2Fccp
```

6. Paste this portion of the URL into the **SSO Url** field

A screenshot of the 'Single SignOn (SSO)' configuration page. The 'SSO Url' field is highlighted with a yellow background and contains the URL 'https://sample-dev-ed.my.salesforce.com/idp/login'. The 'SSO Relay State' field below it is also visible.

7. For the SSO Relay State, copy everything AFTER the question mark (do not copy the question mark), for example:

```
https://mXXXXXrun-dev-ed.my.salesforce.com/idp/login?  
app=0sp0N00000Caid&RelayState=https://us-west-  
2.console.aws.amazon.com/connect/federate/<b>InstanceId</b>?  
destination=%2Fconnect%2Fccp
```

8. Paste this portion of the URL into the **SSO Relay State** field

✓ Single SignOn (SSO)

SSO Url
https://sample-dev-ed.my.salesforce.com/idp/login

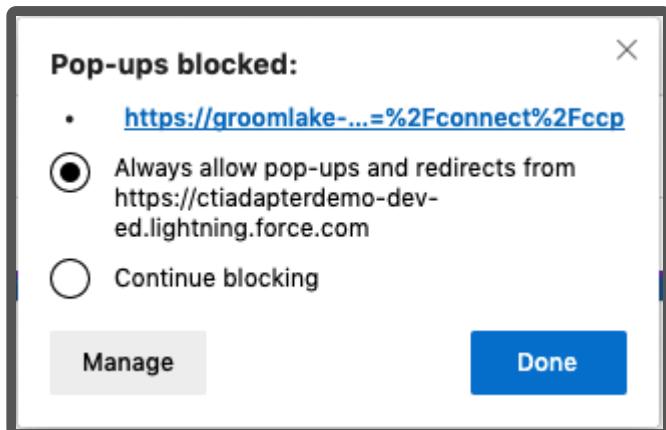
SSO Relay State
app=0sp6g000000XZyd&RelayState=https://us-west-2.console.aws.amazon.com/connect/federate/YOUR-INSTANCE-ID?
destination=%2Fconnect%2Fccp

9. Choose **Save**

Note: With the new Amazon Connect instance urls (*.my.connect.aws) you must put the full URL into the **Amazon Connect Instance** field in the AC CTI Adapter record for SSO to work. Ex: using **https://myinstance.my.connect.aws** instead of **my instance**.

10. **Refresh** your browser to make the changes take effect

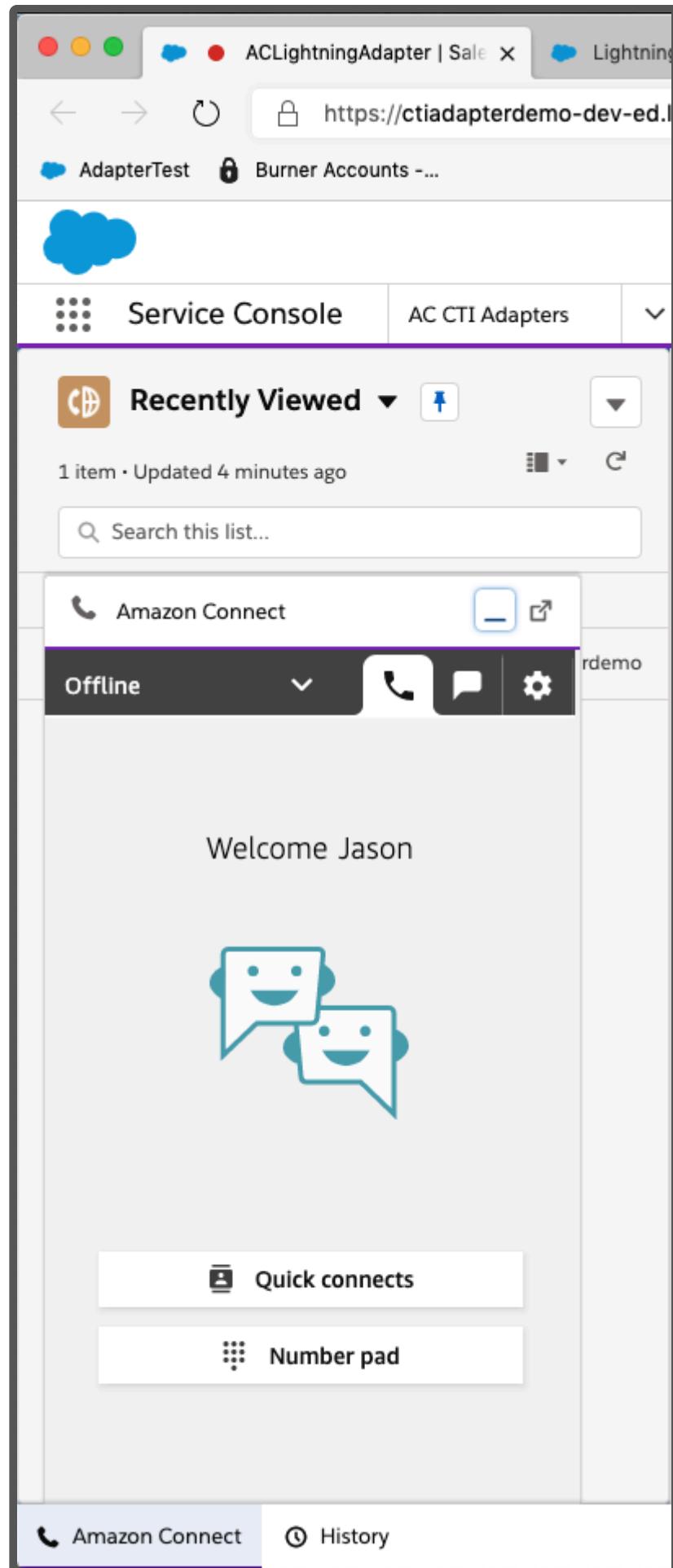
- NOTE:** If you receive a blocked popup warning, select the warning and change the setting to always allow popups from your Salesforce org, then refresh the browser again



11. Select the **phone icon** in the console toolbar to open the CCP Note: You may also receive popups to allow notifications and microphone access. Please accept both.

12. Click the Sign into CCP button

13. You should now see the authenticated and logged in CCP



14. Configuration is complete

Appendix C: CTI Flow Sources and Events

The following sources are defined in the adapter for use with CTI Flows:

- Initialization
 - onInit -- The CTI adapter has initialized.
- Amazon Connect Agent
 - onRefresh -- The Connect agent's data was updated.
 - onStateChange -- The Connect agent's state changed.
 - onRoutable -- The Connect agent became available for contacts.
 - onNotRoutable -- The Connect agent became unavailable for contacts.
 - onOffline -- The Connect agent's state was set to "Offline".
 - onError -- The Connect agent encountered a system error.
 - onAfterCallWork -- The Connect agent entered "After Call Work".
 - onInit -- The Connect agent has logged in.
- Amazon Connect Voice Contact
 - onIncoming -- The voice contact is incoming. Note: This event fires for queued callback contact only.
 - onConnecting -- The voice contact is connecting. Note. This event fires for inbound and outbound contacts except queued callback contacts.
 - onConnected -- The voice contact is connected.
 - onEnded -- The voice contact is ended or destroyed.
 - onRefresh -- The voice contact is updated.
 - onAccepted -- A voice contact is accepted.

- onPending -- The voice contact is pending.
 - onMissed -- The voice contact is / was missed.
 - onDestroy - The voice contact is destroyed.
- Amazon Connect Chat Contact
 - onConnecting -- The chat contact is connecting.
 - onConnected -- The chat contact is connected.
 - onEnded -- The chat contact ended.
 - onRefresh -- The chat contact is updated.
 - onAccepted -- The chat contact is accepted.
 - onPending -- The voice contact is pending.
 - onMessageReceived -- A message was received from the customer
 - onMessageSent -- A message was sent to the customer
 - onMissed -- The chat contact was missed.
 - onDestroy - The voice contact is destroyed.
- Amazon Connect Task Contact
 - onIncoming -- The tasks contact is incoming.
 - onConnecting -- The task contact is connecting.
 - onConnected -- The task contact is connected.
 - onEnded -- The task contact ended.
 - onRefresh -- The task contact is updated.
 - onAccepted -- The task contact is accepted.
 - onPending -- The voice contact is pending.
 - onMissed -- The task contact was missed.

- onDestroy - The voice contact is destroyed.
 - onTransferInitiated -- When the server has initiated the task transfer.
 - onTransferSucceeded -- When the task transfer has succeeded.
 - onTransferFailed -- When the task transfer has failed.
 - onTaskExpiring -- Triggers 2 hours before the task expires.
 - onTaskExpired -- When the task has expired.
- Salesforce Agent
 - onStateChange -- The Salesforce agent's state changed.
 - onWorkAccepted -- The Salesforce agent accepted work.
 - onWorkloadChanged -- The Salesforce agent's workload changed.
 - Salesforce UI
 - onClickToDial -- A phone number, within the Salesforce UI, was clicked.
 - onNavigationChange
 - onHvsWorkStart

Appendix D: CTI Flow Examples

Voice Contact Screenpop (Legacy Adapter Support)

Source: Amazon Connect Voice Contact

Event: onConnecting

[Download](#)

Chat Contact Screenpop

Source: Amazon Connect Chat Contact

Event: onConnecting

[Download](#)

Click-to-Dial

Source: Amazon Connect Chat Contact

Event: onClickToDial

[Download](#)

Screen Pop on Customer Phone Number

Source: Amazon Connect Voice Contact

Event: onConnecting

[Download](#)

Screen Pop a Case on Contact Attribute Data (if it exists) or Pop a New Case (if it does not)

Source: Amazon Connect Voice Contact

Event: onConnecting

[Download](#)

Create a Task (Call Activity) and Pop That Task

Source: Amazon Connect Voice Contact

Event: onConnecting

[Download](#)

Screenpop on Customer Email Address (in contact attribute data)

Source: Amazon Connect Chat Contact

Event: onConnecting

[Download](#)

Create a Task (Call Activity) and Pop That Task

Source: Amazon Connect Chat Contact

Event: onConnecting

[Download](#)

Create a Task (Call Activity) and Pop That Task using CTI Actions

Source: CTI Action

Event: N/A

[More details](#)

[Download](#)

Create a Record on Chat Connected and Screenpop

Source: Amazon Connect Chat Contact

Event: onConnected

[Download](#)

Screenpop Chat Contact on View

Source: Amazon Connect Chat Contact

Event: onViewContact

[Download](#)

Default CTI Flows

The following zip file includes default flows, which are automatically added and activated on new installations of the package. However, if you are upgrading from an earlier version you may need to replace your legacy script with the new flow.

Appendix E: Integration with Salesforce High Velocity Sales

What is High Velocity Sales?

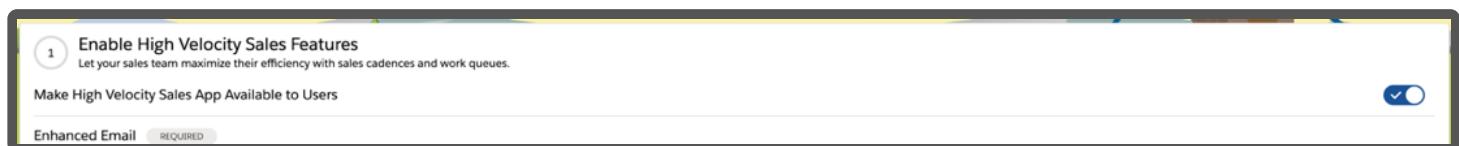
Salesforce HVS (HVS) is a process for your inside sales team to follow a repeatable pre-defined sales cadence for your business. It enables sales managers and representatives to work on a prioritized list of prospects and follow best sequence of sales outreach activities as defined by your sales process.

Enabling the Integration with High Velocity Sales

In order to make HVS works for your connect users, you must enable High Velocity Sales in your Salesforce Org.

Enable High Velocity Sales

1. From Setup, enter High Velocity Sales in the Quick Find box, then select High Velocity Sales.
2. Toggle "Enable High Velocity Sales Features" from disable to enable state



Call Outcomes for Branching

In this step, you can define call disposition values which can be used to branch sales cadence to define next best action for your sales process.

Define Call Outcomes for Branching

1. From Setup, enter High Velocity Sales in the Quick Find box, then select High Velocity Sales.
2. Edit the Define Call Outcomes for Branching.
3. Enter the call result values used by your org next to related call outcomes.

Configure High Velocity Sales

Define Call Outcomes for Branching RECOMMENDED

Call results are disposition values such as "Left Voicemail" that are captured when you log a call. Relate those values to call outcomes to display this data in reports and use it as branching criteria for sales cadences.

Call Outcomes	Call Result Values
Displays in reports and the Sales Cadence Builder.	Enter related call result values. If you have multiple, separate each value with a comma.
Call Back Later	Call Back later, No Answer
Left Voicemail	Left Voicemail
Meaningful Connect	Connected
Not Interested	Not Interested
Unqualified	Unqualified

Cancel Save

Assign HVS permission sets to Connect Users

For creating Sales Cadence, you need to have **High Velocity Sales Cadence Creator** permission set otherwise assign the **High Velocity Sales User** permission set to sales users.

Assign the permission set

1. From Setup, enter permission Sets in Quick Find box, and then select Permission Sets.
2. Select permission set, then click Manage Assignments to assign the permission set to users.

Create Sales Cadence

In HVS application, you will need to create a Sales Cadence based on Sales process

Create a Sales Cadence

1. Choose **Sales Cadence** from navigation menu.
2. Click the down arrow button then click **New**
3. Enter name and description. Click **Save** button which opens **Sales Cadence** builder screen.

Recently Viewed

3 items - Updated 9 minutes ago

New

Search this list...

RECENTLY VIEWED

Sales Cadence 1 10/10/2019 10:57 PM Active

New Sales Cadence 10/10/2019 3:53 PM Active

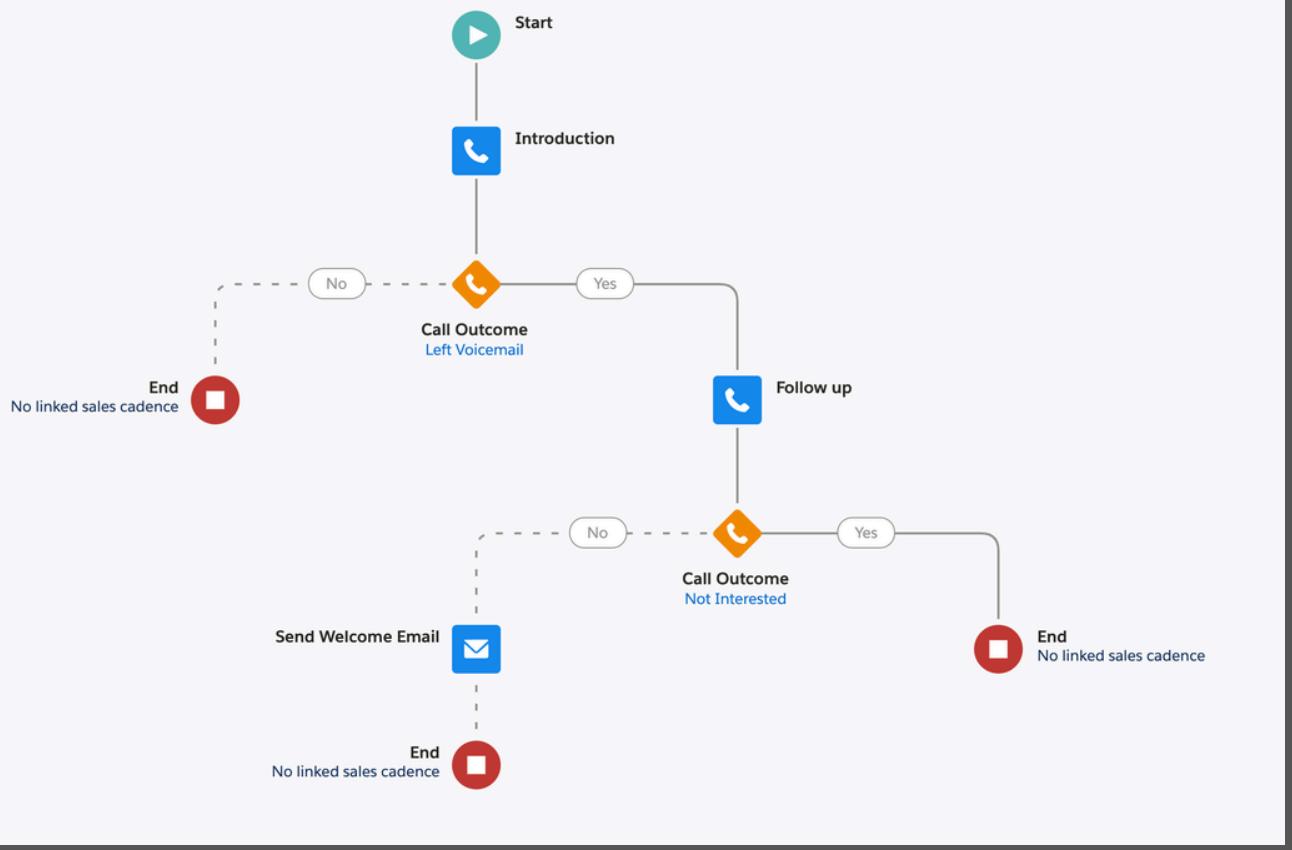
New Sales Cadence

Information

Name

Description

4. Click + sign in the builder to add a step. Choose a type of step you want to add for your sales cadence. Once you finish adding steps, click the **Activate** button. Once a sales cadence is active, you can add leads, contact, and personal accounts to Sales Cadence.



Assigning Prospects

You can assign a prospect to a Sales Cadence either on a prospect detail page or through an automated flow. In this example, using prospect detail page to assign a sales cadence.

The screenshot shows the Prospect Detail page for 'Jo Jim'. The top navigation bar includes 'Follow', 'Convert', 'Edit', and a dropdown menu. Below the name, there are fields for 'Title' and 'Company' (set to 'Test'). The 'Phone' field contains the number '(212) 121-2111'. The 'Email' field is empty. On the left, under 'Sales Cadence Steps', it says 'Jo Jim is not currently in a sales cadence.' and features a prominent blue button labeled 'Add to Sales Cadence'. To the right, the main content area shows the prospect's status as 'New' with a progress bar. It includes sections for 'Activity' (with tabs for 'Log a Call', 'New Task', and 'New Event'), 'Chatter', and 'Details'. A note says 'Email insights only Disabled'. Below this, there are sections for 'Upcoming & Overdue' activities and a message stating 'No next steps. To get things moving, add a task or set up a meeting.' A 'Show All Activities' button is at the bottom. The top right of the page has a 'Mark Status as Complete' button.

Click **Add to Sales Cadence** button to add this prospect to a Sales Cadence.

Create and Map Dispositions

In this step you need to add a disposition field on Activity object and map disposition options to what is defined in HVS call outcomes. In this example, I am going to create a picklist field and add it to default task page layout to track disposition value for each call.

Create and map disposition fields

1. Go to the Setup screen then click **Object Manager**
2. Click **Activity Object**
3. In Fields and Relationships section select **New**
4. Select a picklist field and choose **Next**
5. Enter require information and add HVS call outcomes as picklist options.
6. Select all default options and add this filed on Task page layout. (If there is already a field called **Call Result** on Task Page layout then remove it from the page layout.)
7. Choose **Save**

Custom Field Definition Detail		Edit	Set Field-Level Security	Where is this used?
Field Information				
Field Label	Call Result	Object Name	Activity	
Field Name	Call_Result	Data Type	Picklist	
API Name	Call_Result__c			
Description				
Help Text				
Data Owner				
Field Usage				
Data Sensitivity Level				
Compliance Categorization				
Created By	Sunil Sinha, 10/10/2019 11:04 PM	Modified By	Sunil Sinha, 10/10/2019 11:04 PM	
General Options				
Required	<input type="checkbox"/>			
Default Value	<input type="text"/>			
Picklist Options				
Restrict picklist to the values defined in the value set	<input checked="" type="checkbox"/>			
Controlling Field	[New]			
Field Dependencies				
No dependencies defined.				
Values				
Action	Values	API Name	Default	Chart Colors
Edit Del Deactivate	Completed	Completed	<input type="checkbox"/>	Assigned dynamically
Edit Del Deactivate	Connected	Connected	<input type="checkbox"/>	Assigned dynamically
Edit Del Deactivate	Left Voicemail	Left Voicemail	<input type="checkbox"/>	Assigned dynamically
Edit Del Deactivate	Not Interested	Not Interested	<input type="checkbox"/>	Assigned dynamically
Edit Del Deactivate	Unqualified	Unqualified	<input type="checkbox"/>	Assigned dynamically

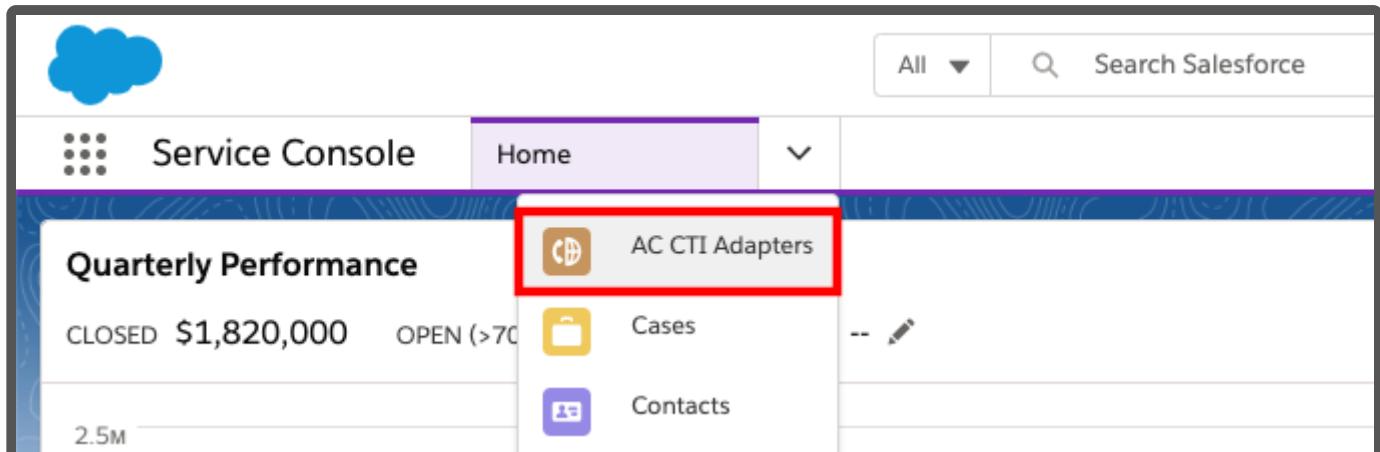
Setup CTI Flows for High Volume Sales

Next you will need to create a new set of CTI Flows for High Volume Sales.

Configuring the CTI Flow

1. Log in into your Salesforce org and go to the **Service Console**

2. Expand the **navigation menu** by selecting the down arrow and choose **AC CTI Adapters**.



3. Select **ACLightningAdapter**

4. Scroll down to the **Scripts** section

5. Select New to create a new CTI Flow



6. In the **CTI Flow Name** field, enter **Voice onHvsWorkStart**

7. Make sure the checkbox for **Active** is selected

8. For the **Source**, select **Salesforce UI**

9. For the **Event**, select **onHvsWorkStart**

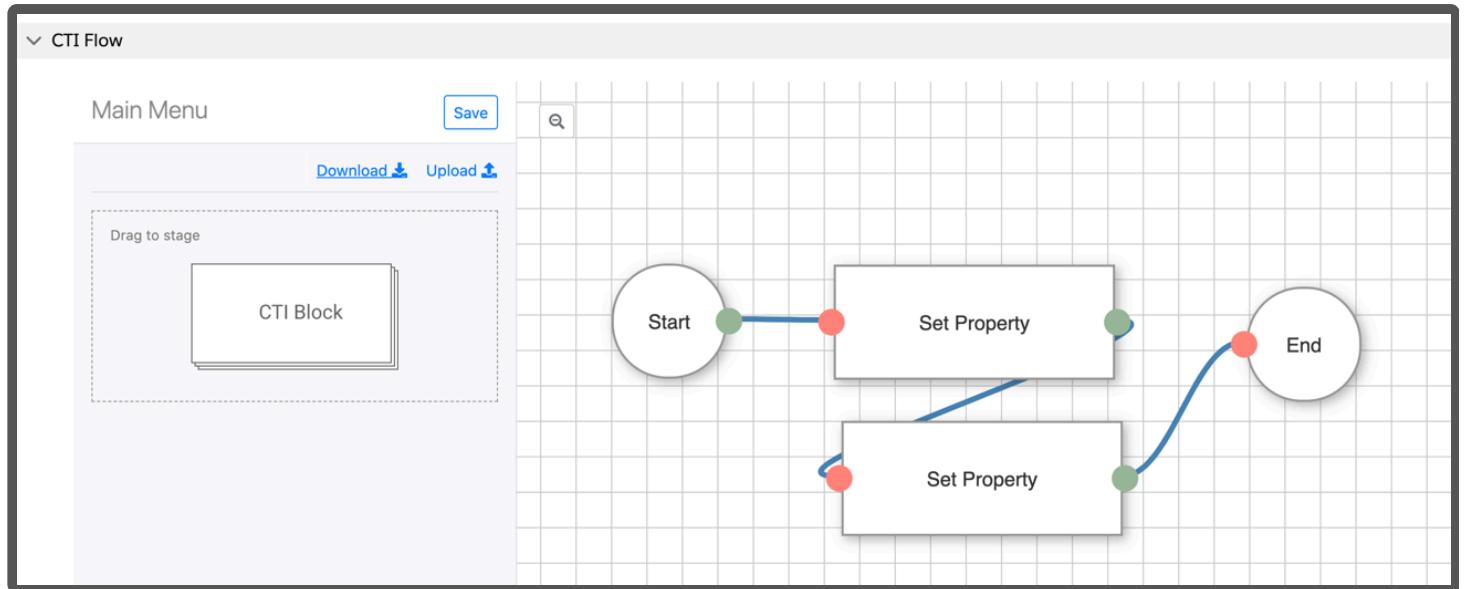
10. Provide a **Description**

11. Click **Save**.

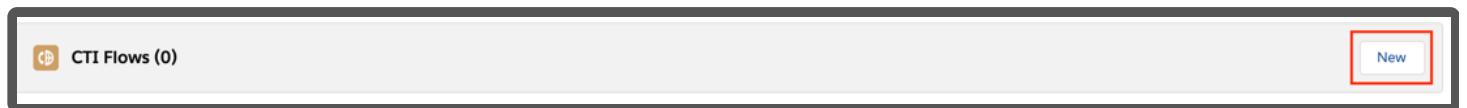
12. Scroll down and click on the link **Voice onHvsWorkStart**.

13. [Download this file](#)

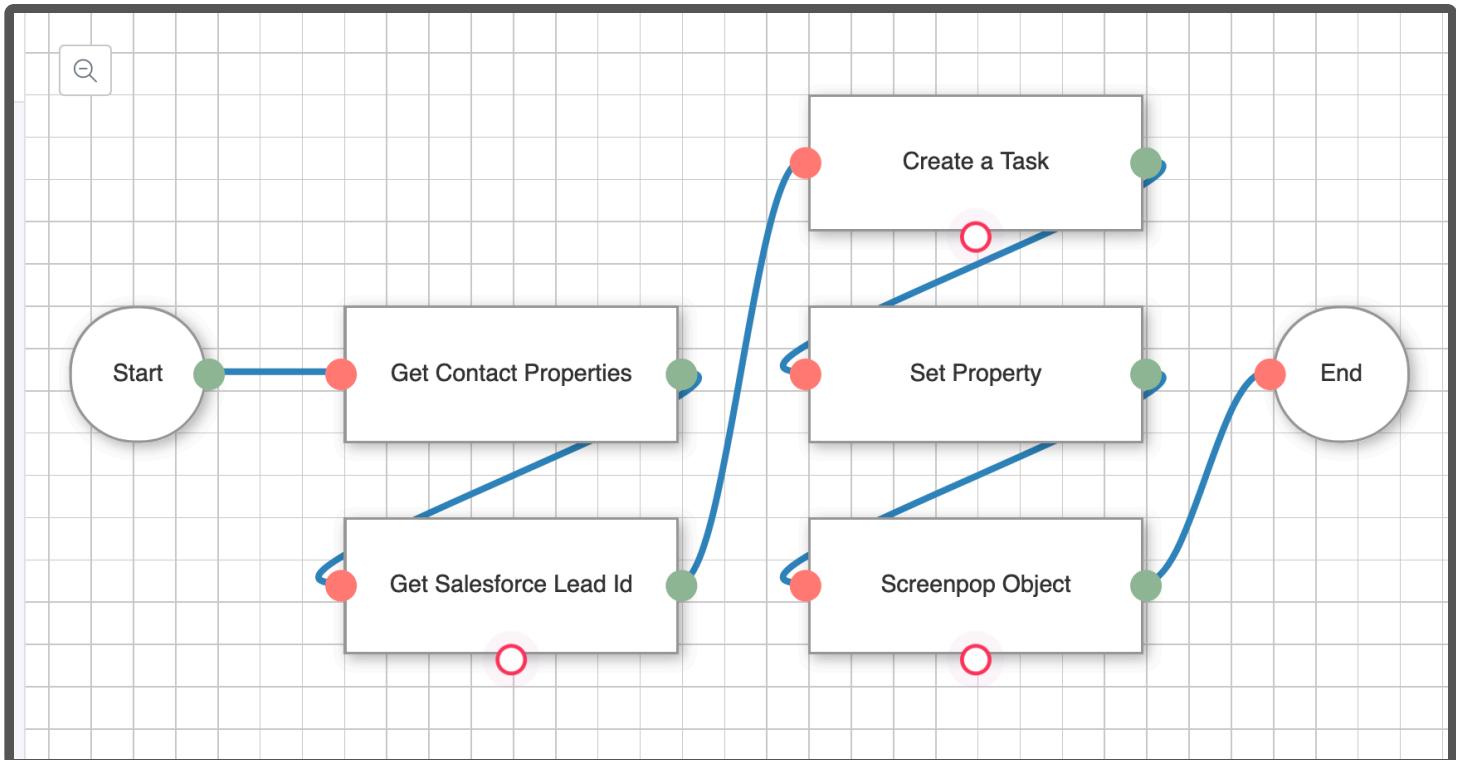
14. Click **Upload** and find the file you just downloaded. You should now see this:**



15. Click **Save**. This creates a CTI Flow that is invoked when you start a HVS work and capture the workId for the third CTI Flow below.
16. Go back to the CTI Adapter page and select **New** in CTI Flows section to create another CTI Flow.



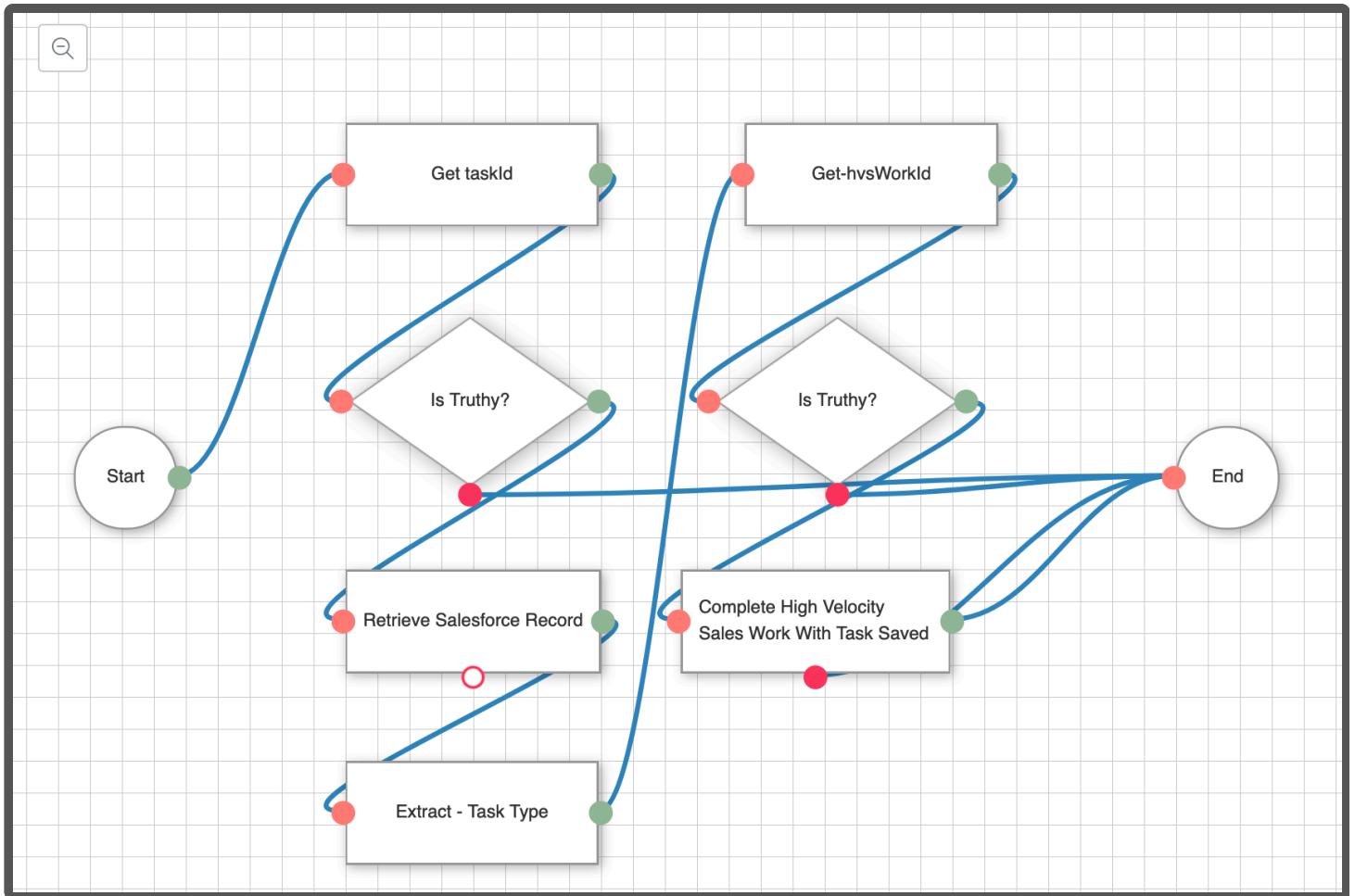
17. In the **CTI Flow Name** field, enter **HVS Voice onConnecting**
18. Make sure the checkbox for **Active** is selected
19. For the **Source**, select **Amazon Connect Voice Contact**
20. For the **Event**, select **onConnecting**
21. Provide a **Description** and Save
22. Scroll down and click on the link **HVS Voice onConnecting**.
23. [Download this file](#)
24. Click **Upload** and find the file you just downloaded. You should now see this:



25. Click **Save**. This creates a CTI Flow creates task for the voice contact and save the taskId for the third CTI Flow below. If you already have a CTI Flow that creates task for voice contact, you do not need to add this one but just need to add a **Set Property** CTI Block to save the taskId
26. Go back to the CTI Adapter page and select **New** in CTI Flows section to create another CTI Flow.



27. In the **CTI Flow Name** field, enter **HVS Voice onRoutable**.
28. Make sure the checkbox for **Active** is selected
29. For the **Source**, select **Amazon Connect Agent**
30. For the **Event**, select **onRoutable**
31. Provide a **Description** and Save
32. Scroll down and click on the link **HVS Voice onRoutable**
33. [Download this file](#)
34. Click **Upload** and find the file you just downloaded. You should now see this:



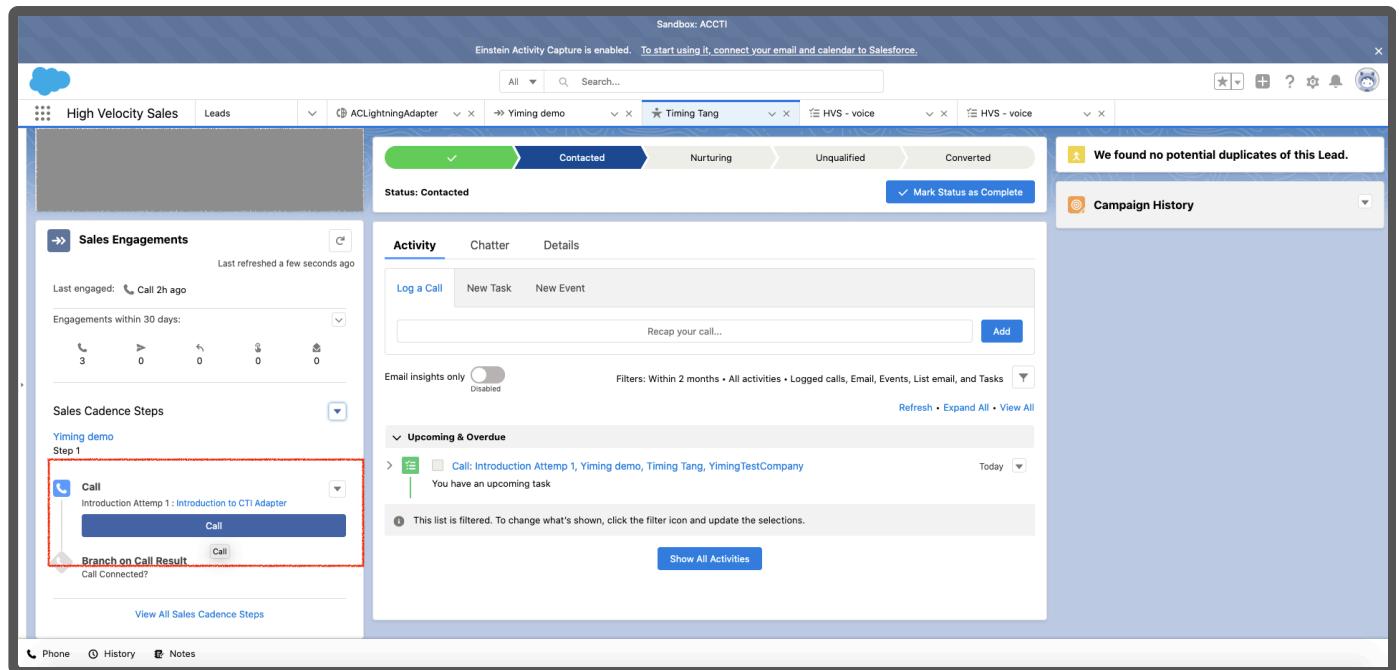
35. Click **Save**. This CTI Flow is executed before your agent is back to routable and retrieves the call result based on the task Id you set in the second CTI Flow, and use it to complete the HVS work

36. Once you've created the flows refresh your browser and the new scripts will take effect.

Expected Behavior

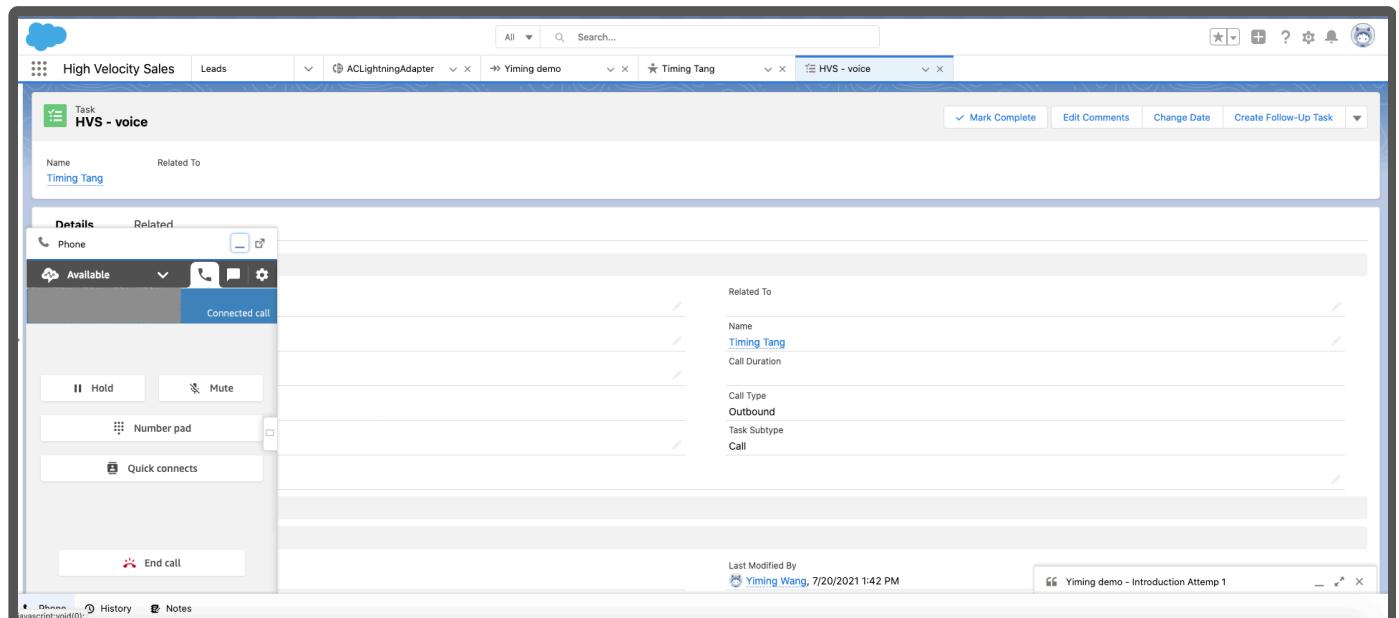
1. Adding Lead to the Sales Cadence you created

2. Make a call to the lead using the call button



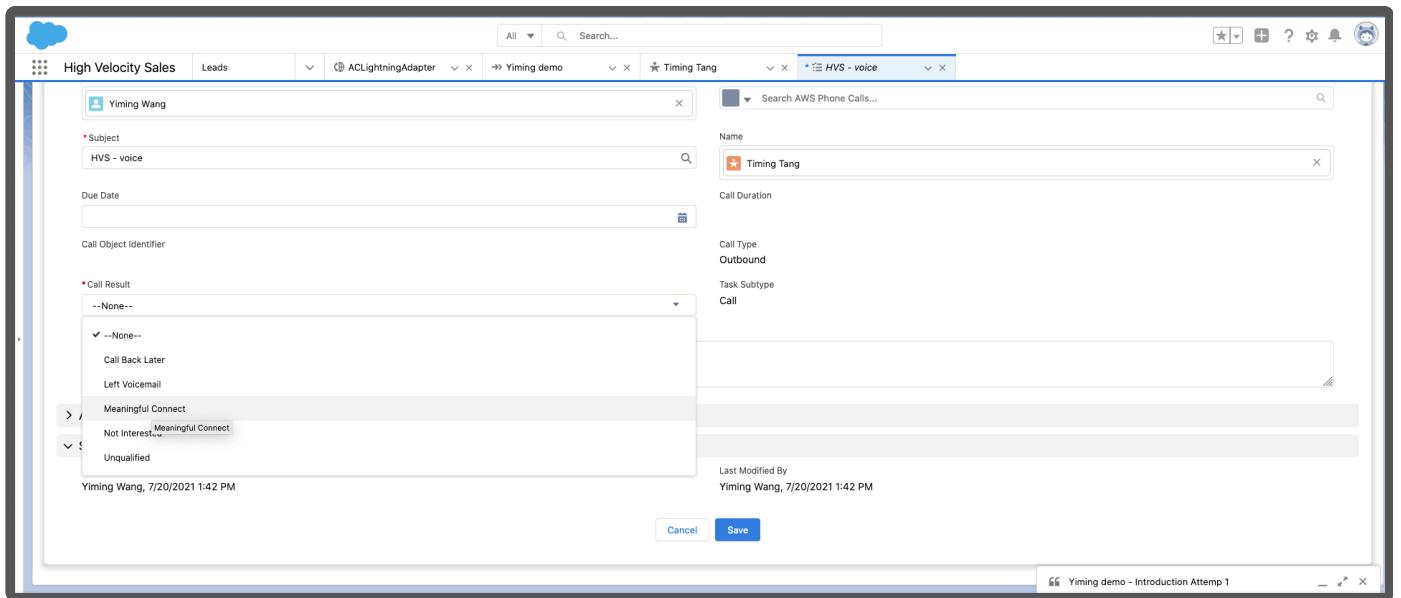
The screenshot shows the Salesforce Lightning Experience interface. The top navigation bar includes tabs for Leads, ACLightningAdapter, Timing demo, Timing Tang, HVS - voice, and HVS - voice. A message at the top right says "We found no potential duplicates of this Lead." On the left, there's a sidebar titled "Sales Engagements" with a summary of recent activity: "Last engaged: Call 2h ago" and "Engagements within 30 days: 3". Below this is a section for "Sales Cadence Steps" for "Yiming demo Step 1". A red box highlights the "Call" button under "Introduction Attempt 1: Introduction to CTI Adapter". The main content area shows an "Activity" tab with sections for "Log a Call", "New Task", and "New Event". It also displays an "Upcoming & Overdue" section with a task for "Call: Introduction Attemp 1, Yiming demo, Timing Tang, YimingTestCompany".

3. An outbound call is made and a task is created and popup

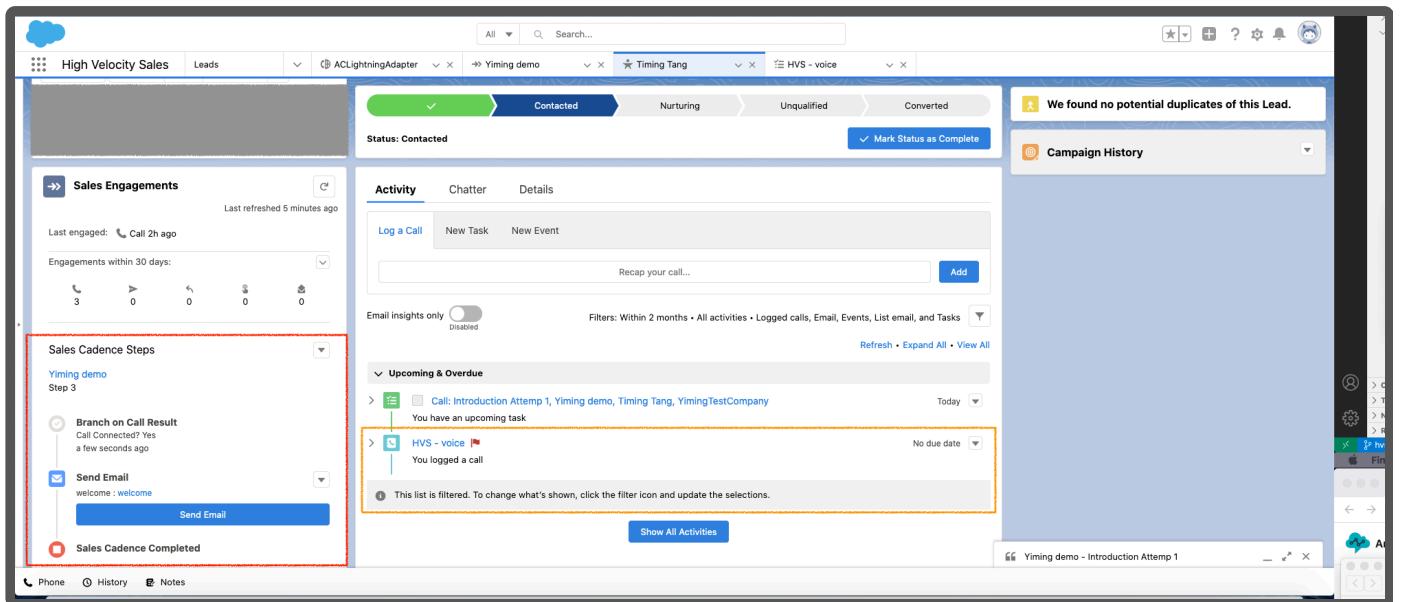


The screenshot shows the Salesforce Lightning Experience interface. The top navigation bar includes tabs for Leads, ACLightningAdapter, Timing demo, Timing Tang, HVS - voice, and HVS - voice. The main content area is a "Task" page for "HVS - voice" related to "Timing Tang". The task details show it is an "Outbound" call. On the left, there's a call control panel with buttons for Hold, Mute, Number pad, Quick connects, and End call. A red box highlights the "Connected call" status in this panel. The task details include fields for Name (Timing Tang), Related To (Timing Tang), Call Duration, and Call Type (Outbound). The task was last modified by Yiming Wang on 7/20/2021 at 1:42 PM.

4. While agent is in After Call Work status, Agent update the Call Result of the popup task and click Save.



5. Agent click Close Contact to be available for the next call. The third CTI Flow will be invoked to retrieve the call result and the Sales Cadence Steps for this lead will be updated (highlighted in red below). The popup task should be linked to the lead as well (highlighted in orange below).



Appendix F: CTI Flow Blocks

If-else

Change the flow of your script depending on value of fields you fetch or store. This is a simple "if-else" utility for your flow.

HTTP Request

Make an HTTP request.

Get Property

Fetches a property from the local data store. You can access a property you have retrieved from the local store by referring to the return value of this block.

Get All Properties

Returns all stored properties.

Format Phone Number

Formats a phone number for a country code.

Format Phone Number (E164)

Formats a phone number for a country code in E164 format.

Format a Date object

Returns a formatted date.

Is Truthy?

This is a utility to branch your flow depending on the truthiness of a value.

Set Property

Assigns a value to a property in the local data store.

Log to Console

Sends a static or dynamic value from an action to a logger.

Show Modal

The command to open modal.

Enable Click To Dial?

The query to determine whether Click to Dial should be enabled.

Enable Click To Dial

The command to enable Click to Dial.

Disable Click To Dial

The command to disable Click to Dial.

Get App View Info

The command to get App View information.

Get Softphone Layout

The query to get softphone layout.

Get Agent Workload on Salesforce

Returns the agent's current workload.

Complete High Velocity Sales Work With Task Saved

This methods allow your CTI implementation to communicate with High Velocity Sales (HVS) to handle HVS work.

Refresh View

The command to refresh the view.

Show Softphone Panel

The command to show softphone panel.

Hide Softphone Panel

The command to hide softphone panel.

Set Softphone Panel Height

The command to set the height of softphone panel.

Set Softphone Panel Width

The command to set the width of softphone panel.

Screenpop Object

The command to open a screenpop with information from object.

Screenpop Url

The command to screenpop a url in a new browser tab or browser window.

Screenpop Object Home

The command to screenpop to an object's home page.

Screenpop List

The command to screenpop a list view.

Screenpop Search

The command to screenpop search results based upon the search input. Not to be confused with "Search And Screenpop."

Screenpop New Record

The command to screenpop to a new record of the specified type with specified default field values.

Search And Screenpop

This command searches objects specified in the softphone layout for a given string. Returns search results and screen pops any matching records. Not to be confused with "Screenpop Search."

Run Apex

The command to run an apex method. Make sure the apex method is in a class that extends the AC_Utils class, and your class must be specified in the extensions list of `AC_CtiScript__c.page` Visualforce page. [See the Salesforce documentation for an example.](#)

Get Agent State from Salesforce

The command to get an agent's state.

Set Agent State on Salesforce

The command to set an agent's presence state on Salesforce.

Login Agent on Salesforce

The command to login an agent on Salesforce.

Logout Agent on Salesforce

The command to logout an agent on Salesforce.

Save (or Create) a Record

The command to save or create a Salesforce object.

Create a Task

The command to create a Task. (The Subject of the task will be a string made up of upto 3 field values.)

Is Contact "Do Not Call"?

The query to check if the Contact requested not to be called.

Dial Number

The command to dial a phone number or to conference to an endpoint.

Mute Agent

The command to mute the agent.

Unmute Agent

The command to unmute the agent.

Get Agent Status from Connect

The command to get the current presence status of the agent from Connect.

Set Agent Status on Connect

The command to set the current presence status of the agent on Connect.

Set Agent Status By Name on Connect

The command to set the current presence status of the agent on Connect by name of the state.

Set Agent as Available on Connect

The command to set the current state of the agent to "Available."

Get Quick Connection List

Gets the list of quick connects available to the current agent

Get Transfer Connection List

Gets the list of quick connects available to the current agent.

Get Endpoint by Phone Number

Generates and returns an endpoint for a provided phone number.

Get Available Agent States

Gets all of the available agent states including custom states.

Get Agent Name

Returns the agent's user friendly display name for the agent.

Get Agent Extension

Returns the phone number that is dialed by Amazon Connect to connect calls to the agent for incoming and outgoing calls, if softphone is not enabled.

Get Agent Deskphone Number

Returns the phone number that is dialed by Amazon Connect to connect calls to the agent for incoming and outgoing calls, if softphone is not enabled.

Is Agent Softphone Enabled?

Checks if agent softphone is enabled. Branches in different directions if it is or not.

Change Agent to Softphone

Changes the current agent to softphone mode.

Change Agent to Deskphone

Changes the current agent to desktop phone mode with the specified phone number.

Get Agent Configuration

Returns the phone number that is dialed by Amazon Connect to connect calls to the agent for incoming and outgoing calls, if softphone is not enabled.

Get Agent Dialable Countries

Returns the list of dialable countries for the current agent.

Create Task Contact

The command to create a task contact that is sent to the provided quick connect endpoint. The quick connect must be available to any queue the agent has access too.

Get Contact Attribute

The command to get value of an attribute from the contact in the current session.

Is Voice Contact?

The command to determine if the contact is a voice contact.

Is Chat Contact?

The command to determine if the contact is a chat contact.

Is Task Contact?

The command to determine if the contact is an amazon connect task contact.

Is Contact Inbound?

The command to determine if the contact is inbound.

Is Contact Transfer?

The command to determine if the contact is transferred.

Is Callback?

The command to determine if the contact is a queue callback.

Get Contact Properties

The command to get properties of a contact.

Get Customer Phone Number

The command to get customer phone number of a contact.

Get Contact Interaction Metadata

The command to get metadata about a contact interaction.

Pop Task Contact's Reference Urls

The command to pop any reference urls if the contact is a task. Returns the number of urls popped.

Query value

The query to execute an arbitrary SOQL statement and returns the results.

Get Salesforce Lead Id

The command to get a salesforce lead id using a formatted phone number.

Get Salesforce Contact Id

The query to get the Id of the Salesforce Contact with the formatted phone number.

Retrieve Salesforce Record

Retrieves a Salesforce record by Id. Note use this block in with the "Extract Value" block to obtain the value of a specific field.

Extract Value

This utility allows you to access the inner value of an object or an array at the designated key path

Open Salesforce Primary Tab

Opens a new primary tab to display the content of the specified URL.

Open Salesforce Sub Tab

Opens a new subtab (within a primary tab) that displays the content of a specified URL.

Get Focused Primary Tab Object Id

Returns the object ID of the primary tab on which the browser is focused.

Get Focused Subtab Object Id

Returns the object ID of the subtab on which the browser is focused.

Call jQuery Method

Perform a method call on a jQuery selection with your arguments.

Replace String

Perform a .replace() method on an input string.

Text Starts With Value

Checks whether a text input starts with one of the values.

Text Ends With Value

Checks whether a text input ends with one of the values.

Join Strings

Concatenates 2 values into a string.

Multiply

Multiply two numbers.

Divide

Divide two numbers.

Get Tab Object Map

Returns a map of all visible primary tabs and their associated objects (if available).

Close Salesforce Tab

Closes the Salesforce with a given id.

Delay

Delays execution for a period of time. (Keep in mind that your flow may be stopped if it runs longer than the maximum allowed execution window of 60 seconds.)

Get Primary Tab Ids

Returns all of the IDs of open primary tabs.

Get Tabs With Matching Url

Returns the ids of the primary tabs with the url matching a provided string.

Length

Returns the length of a value.

Slice

Returns the slice of a value.

Cast a Value to a Type

Cast an input value to a Javascript type, such as Number or String.

Get CCP Logs

The command to get the logs of agent from Connect.

Clear All Properties

Clears all stored properties.

Unset Property

Removes the value assigned to a property in the local data store.

Is Task Contact?

Check if the contact is a task

Create Task Contact

Creating a new task contact with certain inputs.

Start Recording

Use the contact recording API to start recording the call.

Stop Recording

Use the contact recording API to stop recording the call.

Update Contact Attributes

Use the Connect API to update the attributes of the current contact.

Get Payload

Retrieve the payload of the CTI Flow. (The payload can be configured by CTI Actions.)

Send Data to CCP Overlay

Send an object to Data panel of CCP Overlay.

Leave a Voicemail

Use Voicemail drops to leave a voicemail.

Destroy Agent Connection to Live Contact

Destroys destroy the agent's connection to any live contact that is currently being handled by the CTI Flow. This is being deprecated for contacts in ACW. Use the ClearContact block for Clear ACW functionality.

Clear Contact

Clears a contact that is no longer being worked on - i.e. it's one of ERROR, ACW, MISSED, REJECTED.