Installing and Configuring Avaya CRM Connector 2.2 for AACC

Release 2.2.7.4
Issue 2
September 2021

Notice

While reasonable efforts have been made to ensure that the information in this document is complete and accurate at the time of printing, Avaya assumes no liability for any errors. Avaya reserves the right to make changes and corrections to the information in this document without the obligation to notify any person or organization of such changes.

Documentation disclaimer

“Documentation” means information published by Avaya in varying mediums which may include product information, operating instructions and performance specifications that Avaya may generally make available to users of its products and Hosted Services. Documentation does not include marketing materials. Avaya shall not be responsible for any modifications, additions, or deletions to the original Published version of documentation unless such modifications, additions, or deletions were performed by Avaya. End User agrees to indemnify and hold harmless Avaya, Avaya's agents, servants and employees against all claims, lawsuits, demands and judgments arising out of, or in connection with, subsequent modifications, additions or deletions to this documentation, to the extent made by End User.

Link disclaimer

Avaya is not responsible for the contents or reliability of any linked websites referenced within this site or documentation provided by Avaya. Avaya is not responsible for the accuracy of any information, statement or content provided on these sites and does not necessarily endorse the products, services, or information described or offered within them. Avaya does not guarantee that these links will work all the time and has no control over the availability of the linked pages.

Warranty

Avaya provides a limited warranty on Avaya hardware and software. Refer to your sales agreement to establish the terms of the limited warranty. In addition, Avaya’s standard warranty language, as well as information regarding support for this product while under warranty is available to Avaya customers and other parties through the Avaya Support website: http://support.avaya.com or such successor site as designated by Avaya. Please note that if you acquired the product(s) from an authorized Avaya Channel Partner outside of the United States and Canada, the warranty is provided to you by said Avaya Channel Partner and not by Avaya.

HostedService

THE FOLLOWING APPLIES IF YOU PURCHASE A HOSTED SERVICE SUBSCRIPTION FROM AVAYA OR AN AVAYA CHANNEL PARTNER (AS APPLICABLE), THE TERMS OF USE FOR HOSTED SERVICES ARE AVAILABLE ON THE AVAYA WEBSITE, HTTP://SUPPORT.AVAYA.COM/ LICENSEINFO UNDER THE LINK “Avaya Terms of Use for Hosted Services” OR SUCH SUCCESSOR SITE AS DESIGNATED BY AVAYA, AND ARE APPLICABLE TO ANYONE WHO ACCESSES OR USES

THE HOSTED SERVICE. BY ACCESSING OR USING THE HOSTED SERVICE, OR AUTHORIZING OTHERS TO DO SO, YOU, ON BEHALF OF YOURSELF AND THE ENTITY FOR WHOM YOU ARE DOING SO (HEREINAFTER REFERRED TO INTERCHANGEABLY AS “YOU” AND “END USER”), AGREE TO THE TERMS OF USE. IF YOU ARE ACCEPTING THE TERMS OF USE ON BEHALF A COMPANY OR OTHER LEGAL ENTITY, YOU REPRESENT THAT YOU HAVE THE AUTHORITY TO

BIND SUCH ENTITY TO THESE TERMS OF USE. IF YOU DO NOT HAVE SUCH AUTHORITY, OR IF YOU DO NOT WISH TO ACCEPT THESE TERMS OF USE, YOU MUST NOT ACCESS OR USE THE HOSTED SERVICE OR AUTHORIZE ANYONE TO ACCESS OR USE THE HOSTED SERVICE. YOUR USE OF THE HOSTED SERVICE SHALL BE LIMITED BY THE NUMBER AND TYPE OF LICENSES PURCHASED UNDER YOUR CONTRACT FOR THE HOSTED SERVICE, PROVIDED, HOWEVER,

THAT FOR CERTAIN HOSTED SERVICES IF APPLICABLE, YOU MAY HAVE THE OPPORTUNITY TO USE FLEX LICENSES, WHICH WILL BE INVOICED ACCORDING TO ACTUAL USAGE ABOVE THE CONTRACT LICENSE LEVEL. CONTACT AVAYA OR AVAYA’S CHANNEL PARTNER FOR MORE INFORMATION ABOUT THE LICENSES FOR THE APPLICABLE HOSTED SERVICE, THE AVAILABILITY OF ANY FLEX LICENSES (IF APPLICABLE), PRICING AND BILLING INFORMATION,

AND OTHER IMPORTANT INFORMATION REGARDING THE HOSTED SERVICE.

Support Tools:

"AVAYA SUPPORT TOOLS"MEAN THOSE SUPPORT TOOLS PROVIDED TO PARTNERS OR CUSTOMERS IN CONNECTION WITH MAINTENANCE SUPPORT OF AVAYA EQUYIPMENT (E.G., SAL, SLA MON, AVAYA DIAGNOISTIC SERVER, ETC.) AVAYA SUPPORT TOOLS ARE INTENDED TO BE USED FOR LAWFUL DIAGNOSTIC AND NETWORK INTEGRITY PURPOSES ONLY. The customer is responsible for understanding and complying with applicable legal requirements with regard to its network. The Tools may contain diagnostic capabilities that allow Avaya, authorized Avaya partners, and authorized customer administrators to capture packets, run diagnostics, capture key strokes and information from endpoints including contact lists, and remotely control and monitor end-user devices. The customer is responsible for enabling these diagnostic capabilities, for ensuring users are aware of activities or potential activities and for compliance with any legal requirements with respect to use of the Tools and diagnostic capabilities on its network, including, without limitation, compliance with laws regarding notifications regarding capture of personal data and call recording.

Licenses

THE SOFTWARE LICENSE TERMS OR SUPPORT TOOLS LICENSE TERMS AVAILABLE ON THE AVAYAWEBSITE, HTTP://SUPPORT.AVAYA.COM/LICENSEINFO

OR SUCH SUCCESSOR SITE AS DESIGNATED BY AVAYA, ARE APPLICABLE TO ANYONE WHO DOWNLOADS, USES AND/OR INSTALLS AVAYA SOFTWARE, PURCHASED FROM AVAYA INC., ANY AVAYA AFFILIATE, OR AN AVAYA CHANNEL PARTNER (AS APPLICABLE) UNDER A COMMERCIAL AGREEMENT WITH AVAYA OR AN AVAYA CHANNEL PARTNER. UNLESS OTHERWISE AGREED TO BY AVAYA IN WRITING, AVAYA DOES NOT EXTEND THIS LICENSE IF THE SOFTWARE WAS OBTAINED FROM ANYONE OTHER THAN AVAYA, AN AVAYA AFFILIATE OR AN AVAYA CHANNEL PARTNER; AVAYA RESERVES THE RIGHT TO TAKE LEGAL ACTION AGAINST YOU AND ANYONE ELSE USING OR SELLING THE SOFTWARE WITHOUT A LICENSE. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE, OR AUTHORIZING OTHERS TO DO SO, YOU, ON BEHALF OF YOURSELF AND THE ENTITY FOR WHOM YOU ARE INSTALLING, DOWNLOADING OR USING THE SOFTWARE (HEREINAFTER REFERRED TO INTERCHANGEABLY AS "YOU" AND "END USER"), AGREE TO THESE TERMS AND CONDITIONS AND CREATE A BINDING CONTRACT BETWEEN YOU AND AVAYA INC. OR THE APPLICABLE

AVAYA AFFILIATE ("AVAYA").

Avaya grants you a license within the scope of the license types described below, with the exception of Heritage Nortel Software and Support Tools, for which the scope of the license is detailed below Where the order documentation does not expressly identify a license type, the applicable license will be a Designated System License. The applicable number of licenses and units of capacity for which the license is granted will be one (1), unless a different number of licenses or units of capacity is specified in the documentation

or other materials available to you. "Designated Processor" means a single stand-alone computing device. "Server" means a Designated Processor that hosts a software application to be accessed by multiple users.

License type(s)

Designated System(s) License (DS). End User may install and use each copy or an Instance of the Software only on a number of Designated Processors up to the number indicated in the order. Avaya may require the Designated Processor(s) to be identified in the order by type, serial number, feature key, Instance, location or other specific designation, or to be provided by End User to Avaya through electronic means established by Avaya specifically for this purpose.

Concurrent User License (CU). End User may install and use the Software on multiple Designated Processors or one or more Servers, so long as only the licensed number of Units are accessing and using the Software at any given time. A "Unit" means the unit on which Avaya, at its sole discretion, bases the pricing of its licenses and can be, without limitation, an agent, port or user, an e-mail or voice mail account in the name of a person or corporate function (e.g., webmaster or helpdesk), or a directory entry in the administrative database utilized by the Software that permits one user to interface with the Software. Units may be linked to a specific, identified Server or an Instance of the Software.

Shrinkwrap License (SR). You may install and use the Software in accordance with the terms and conditions of the applicable license agreements, such as "shrinkwrap" or "clickthrough" license accompanying or applicable to the Software ("Shrinkwrap License").

Heritage Nortel Software

"Heritage Nortel Software" means the software that was acquired by Avaya as part of its purchase of the Nortel Enterprise Solutions Business in December 2009. The Heritage Nortel Software currently available for license from Avaya is the software contained within the list of Heritage Nortel Products located at http://support.avaya.com/LicenseInfo/ under the link "Heritage Nortel Products", or such successor site as designated by Avaya. For Heritage Nortel Software, Avaya grants Customer a license to use Heritage

Nortel Software provided hereunder solely to the extent of the authorized activation or authorized usage level, solely for the purpose specified in the Documentation, and solely as embedded in, for execution on, or (in the event the applicable Documentation permits installation on non-Avaya equipment) for communication with Avaya equipment. Charges for Heritage Nortel Software may be based on extent of activation or use authorized as specified in an order or invoice.

Support Tools: Avaya Support Tools are provided as an entitlement of Avaya Support Coverage (e.g., maintenance) and the entitlements are established by Avaya. The scope of the license for each Tool is described in its License terms and/or the applicable service description document.

Copyright

Except where expressly stated otherwise, no use should be made of materials on this site, the Documentation, Software, Hosted Service, or hardware provided by Avaya. All content on this site, the documentation, Hosted Service, and the Product provided by Avaya including the selection, arrangement and design of the content is owned either by Avaya or its licensors and is protected by copyright and other intellectual property laws including the sui generis rights relating to the protection of databases. You may

not modify, copy, reproduce, republish, upload, post, transmit or distribute in any way any content, in whole or in part, including any code and software unless expressly authorized by Avaya. Unauthorized reproduction, transmission, dissemination, storage, and or use without the express written consent of Avaya can be a criminal, as well as a civil offense under the applicable law.

Third Party Components

"Third Party Components" mean certain software programs or portions thereof included in the Software or Hosted Service may contain software (including open source software) distributed under third party agreements ("Third Party Components"), which contain terms regarding the rights to use certain portions of the Software ("Third Party Terms"). As required, information regarding distributed Linux OS source code (for those Products that have distributed Linux OS source code) and identifying the copyright holders of the Third Party Components and the Third Party Terms that apply is available in the Documentation or on Avaya's website at: http://support.avaya.com/Copyright or such successor site as designated by Avaya. You agree to the Third Party Terms for any such Third Party Components.

THIS PRODUCT IS LICENSED UNDER THE AVC PATENT PORTFOLIO LICENSE FOR THE PERSONAL USE OF A CONSUMER OR OTHER USES IN WHICH IT DOES NOT RECEIVE REMUNERATION TO (i) ENCODE VIDEO IN COMPLIANCE WITH THE AVC STANDARD ("AVC VIDEO") AND/OR (ii) DECODE AVC VIDEO THAT WAS ENCODED BY A CONSUMER ENGAGED IN A PERSONAL ACTIVITY AND/OR WAS OBTAINED FROM A VIDEO PROVIDER LICENSED TO PROVIDE AVC VIDEO. NO LICENSE IS GRANTED OR SHALL BE IMPLIED FOR ANY OTHER USE. ADDITIONAL INFORMATION MAY BE OBTAINED FROM MPEG LA, L.L.C. SEE HTTP://WWW.MPEGLA.COM.

Note to Service Provider

The Product or Hosted Service may use Third Party Components subject to Third Party Terms that do not allow hosting and require a Service Provider to be independently licensed for such purpose. It is your responsibility to obtain such licensing.

Preventing Toll Fraud

"Toll Fraud" is the unauthorized use of your telecommunications system by an unauthorized party (for example, a person who is not a corporate employee, agent, subcontractor, or is not working on your company's behalf). Be aware that there can be a risk of Toll Fraud associated with your system and that, if Toll Fraud occurs, it can result in substantial additional charges for your telecommunications services.

Avaya Toll Fraud intervention

If you suspect that you are being victimized by Toll Fraud and you need technical assistance or support, call Technical Service Center Toll Fraud Intervention Hotline at +1-800-643-2353 for the United States and Canada. For additional support telephone numbers, see the Avaya Support website: http://support.avaya.com or such successor site as designated by Avaya. Suspected security vulnerabilities with Avaya products should be reported to Avaya by sending mail to: securityalerts@avaya.com.

Trademarks

The trademarks, logos and service marks (“Marks”) displayed in this site, the Documentation, Hosted Service(s), and Product(s) provided by Avaya are the registered or unregistered Marks of Avaya, its affiliates, or other third parties. Users are not permitted to use such Marks without prior written consent from Avaya or such third party which may own the Mark. Nothing contained in this site, the Documentation, Hosted Service(s) and Product(s) should be construed as granting, by implication, estoppel, or otherwise,

any license or right in and to the Marks without the express written permission of Avaya or the applicable third party.

Avaya is a registered trademark of Avaya Inc.

All non-Avaya trademarks are the property of their respective owners. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

All non-Avaya trademarks are the property of their respective owners, and “Linux” is a registered trademark of Linus Torvalds.

Downloading Documentation

For the most current versions of Documentation, see the Avaya Support website: http://support.avaya.com, or such successor site as designated by Avaya.

Contact Avaya Support

See the Avaya Support website: http://support.avaya.com for Product or Hosted Service notices and articles, or to report a problem with your Avaya Product or Hosted Service. For a list of support telephone numbers and contact addresses, go to the Avaya Support website: http://support.avaya.com (or such successor site as designated by Avaya), scroll to the bottom of the page, and select Contact Avaya Support.

Table of Contents

Chapter 1: Introduction

About this guide

This document is intended for administrators who want to install and configure Avaya CRM Connector version 2.2.

Business usage scenario

Avaya CRM Connectors 2.2 benefits Avaya customers who wish to use a CRM application together with one or more Avaya platforms, such as Avaya Aura® Contact Center (AACC).

The solution supports only inbound/outbound use case for voice-only communication for AACC.

Solution overview

Avaya CRM Connector 2.2 is a new modular software that allows Avaya to quickly integrate voice-only contact center features into Customer Relationship Management (CRM) applications and at the same time provide a highly scalable and robust solution to Avaya contact center customers. A key part of Avaya CRM Connector 2.2 is its modular architecture built as containerized micro-services running on Docker, composed by a common core and adapters to connect to different Avaya platforms and different CRM applications.

Chapter 2: Avaya CRM Connector 2.2 Prerequisites

Important:

It is recommended that you use a thin client version to deploy the Avaya CRM Connector 2.2 application.

Pre-deployment checklist

Accesses and Permissions

No.	Accesses	Permissions to:	Notes	✔
1	Avaya Aura® Communication Manager	a. Trace b. Verify configuration	(MST) Trace in case troubleshooting is needed.	
2	SSH access to Avaya CRM Connector™ servers	Check logs		
3	Salesforce.com	a. Install Managed Packages b. Create custom fields c. Import and configure Call Center Definitions		

Required External platform configuration

No.	Configuration required	Notes	✔
1	• AACC system is configured properly and the machine is up and running. • Security settings must be active, and the agents must have CCT licenses.	If the CRM Connector is the sole controller of the logged in agent, multiple applications simultaneously controlling the same agent is not supported. For more information on configuring AACC, refer the AAAC documents mentioned in the *Appendix D - Resources* section.	
2	Test elements on the AACC server, which include: a. Agents b. Skillsets c. CDNs as required		
3	Salesforce.com test account		
4	Root CA Certificate for WebLM		

Data gathering

No.	Data required	Notes	✔
1	AACC server IP address		
2	Salesforce test user account credentials		
3	Connector server IP or FQDN		

Note:

A Salesforce Administrator resource provided by Avaya's Customer is required to perform and support Salesforce deployment and configuration related tasks.

System requirements

- **Avaya CRM Connector™ Release 2.2 Cluster Profile**
  - **Minimal Profile**

    Each Avaya CRM Connector™ node in a cluster is a single vAppliance package with the following characteristics:

    - Operating system: RHEL 7.9 64–bit
    - CPU Core(s): 8 floating cores CPU reservation 18960MHz = 8x2370MHz
    - Memory reservation: Minimum 8.0 GB
    - Storage reservation: Minimum 30 GB
    - Shared NIC(s): Two at 1000 Mbps, used for management interface and security module/public access.
  - **Standard Profile**

    Each Avaya CRM Connector™ node in a cluster is a single vAppliance package with the following characteristics:

    - Operating system: RHEL 7.9 64–bit
    - CPU Core(s): 8 floating cores CPU reservation 18960MHz = 8x2370MHz
    - Memory reservation: Minimum 16.0 GB
    - Storage reservation: Minimum 50 GB
    - Shared NIC(s): Two at 1000 Mbps, used for management interface and security module/public access.

  Note

  1) Memory and storage reservations are optional on lab deployments.
  2) All the Avaya CRM Connector™ servers in a cluster must have the same memory reservation. Resource requirements may increase depending on the number of agent using Avaya CRM Connector™ 2.2.
  3) Avaya CRM Connector™ 2.2 requires a licensed VMware instance (standard edition or better) and any the following versions of the VMware hypervisor and products:
     - ESXi 6.7 and its updates; or
     - ESXi 7.0 and its updates;

     Important:

     You must use thin client to deploy ova due to limitation of encryption key.

Avaya platform requirements

- Avaya Aura ® Communication Manager Releases 6.3, 7.0.1, 7.1 and 8.0.
- Avaya Web License Manager 7.0 and 7.1
    - Standalone or provided by System Manager 7.0 and 7.1
    - Installed, configured, and contains the proper licenses deployed
- Avaya Aura Contact Center 7.1.0.2 or later
    - AACC login - AgentId must have the following format: userId@domain.com

    Important:

    Only ACD-Only login mode is currently supported for AACC.

Software requirements

CRMs supported:

CRM	Version	Presentation Modes
Salesforce.com	Latest	Lightning Experience, Classic Console, and Classic Standard

Browsers supported:

- Chrome version 89, 90 and 91
- Firefox version 87, 88 and 89
- Microsoft Edge version 89, 90 and 91 (Chromium version)

Note

- Only HTML 5 compliant browsers are supported.
- The maximum number of concurrent softphone tabs supported for Salesforce Classic mode is 5.

Network port requirements

- Network ports are configurable and can be changed in .yml files for each component during the installation.

Avaya CRM Connector Solution port usage

Application	Source	Destination Port	Purpose
Avaya CRM Connector Service	HTTP Client / Browser	8484	HTTPS
Interaction Endpoint Controller Service	HTTP Client / Browser	8483	HTTPS
Avaya Web LM	Avaya CRM Connector	52233	WebLM
AACC	HTTPS	443	HTTPS

Note:

All ports use only TCP.

The port numbers listed above are the default ports, adjust them according to environment configuration

Endpoint compatibility

Hardphones

- Avaya recommends that you use CRM Connector to perform all telephony tasks, such as logging on or off, changing your status to Ready or Not Ready, accepting or rejecting a call, placing a customer on hold, transferring a customer, calling a supervisor, and releasing a call.
- Softphones supported type:
  - one-X® Communicator 6.2.6
  - Avaya Workplace Client softphone

    Prior to Release 3.7.x, this product is Avaya Equinox®. Avaya Aura® Contact Center supports Avaya Workplace Client as a softphone only. Avaya Aura® Contact Center does not support using Avaya Workplace Client for call control functionality, or for any other advanced features such as IM, Presence, or Conferencing.
  - Avaya one-X® Agent

Important:

Avaya one-X® Agent is supported, but with limitations. The limitations are as follows:

- The agent state may display incorrectly in Avaya one-X® Agent.
- The agent state reason codes may display incorrectly in Avaya one-X® Agent.

Certificate requirements

- External Certificates
  - Root CA for WebLM
- Hosting Server Certificates
  - Certificate for Server to deliver secure content over HTTPS

    To know more about Server certificate deployment, see *Deploying the Server Certificate*.
- AACC root certificate and identity certificate

License requirements

The licenses are managed by WebLM where the WebLM URL is now configured in the `application.yml` file.

- Application Licenses:
  - License file to be deployed on Avaya Web LM

  Note

  Contact your Avaya sales representative for more information.
- Avaya Platform Licenses
  - CM station port licenses (1 per logged agent)
  - CM agent licenses (1 per agent created on CM).

  Note

  These are the same standard licenses needed to implement a contact center.

List of features

Used By	Feature/Counter Keyword name	Feature Keyword Description	Use Description
AACC3PCC-Driver	FEAT_AESO_CALLCENTER	CRM Connector AACC Voice Adapter	• Checked on application startup
IEC	VALUE_AESO_OPEN_USERS	CRM Connector Inbound Users	• Acquired on Station Registration Request

Used By	Feature/Counter Keyword name	Feature Keyword Description	Use Description
			• Released on Station Logout Request

Environment configuration

Communication Manager Configurations

IMS

Communication Manager must not be configured as a Feature Server. On page 1 of the relevant signaling group, ensure that IMS enabled is set to **N**.

```
display signaling-group 13                                    Page   1 of   3
                              SIGNALING GROUP

 Group Number: 13              Group Type: sip
 IMS Enabled? n          Transport Method: tcp
        Q-SIP? n
     IP Video? y          Priority Video? y       Enforce SIPS URI for SRTP? y
  Peer Detection Enabled? y  Peer Server: SM
 Prepend '+' to Outgoing Calling/Alerting/Diverting/Connected Public Numbers? y
Remove '+' from Incoming Called/Calling/Alerting/Diverting/Connected Numbers? n
Alert Incoming SIP Crisis Calls? n
   Near-end Node Name: procr                Far-end Node Name: sm48dot13
 Near-end Listen Port: 5060               Far-end Listen Port: 5060
                                       Far-end Network Region: 1

Far-end Domain: sipccgal.com
                                            Bypass If IP Threshold Exceeded? n
Incoming Dialog Loopbacks: eliminate               RFC 3389 Comfort Noise? n
        DTMF over IP: rtp-payload          Direct IP-IP Audio Connections? y
Session Establishment Timer(min): 3                  IP Audio Hairpinning? n
        Enable Layer 3 Test? y                Initial IP-IP Direct Media? y
H.323 Station Outgoing Direct Media? n       Alternate Route Timer(sec): 6
```

Figure 1: Disable IMS feature

For more details, see the *Configuring a SIP Signaling Group for the first Session* section in Avaya *Aura® Contact Center and Avaya Aura® Unified Communications Platform Integration* guide for release 7.1 or later version available at https://support.avaya.com.

The Avaya *Aura® Contact Center and Avaya Aura® Unified Communications Platform Integration* guide can be found at https://support.avaya.com under Documents section → Product AACC → Version 7.1 → Content Type Installation, Upgrades & Config.

Reason Codes

Setting up the reason codes for Not Ready and ACW in AACC

You must refer to the *Configuring activity codes* section of the *Avaya Aura® Contact Center Client Administration* guide.

The guide is located at https://support.avaya.com under Documents section → Product AACC → Version 7.1 → Content Type User Guides.

Conference member limitation in Avaya Aura® Communication Manager

The conference member limit in Avaya Aura® Communication Manager is six, therefore call recorders must be configured with a service observing strategy on all extensions being used by the connector. This recorder strategy will prevent exceeding the conference member limit and therefore allow transfer or conference operations to complete. The call recording strategies requiring more than six conference members across the main call leg and the consultation call leg at completion of the transfer or the conference call cannot be supported by this solution.

Environment prerequisites

System Clocks

The systems clocks on the servers and clients must be properly set. This is required to store call logs with correct date and time.

Chapter 3: Installation and configuration

Product Artifacts

The following artifacts must be available in order to be able to continue the installation process.

Name	Type	Description
Avaya CRM Connector 2.2	ova	Avaya Connector VM Image containing all the services needed by the connector.
SFDC CCD	xml	Salesforce Call Center definition files to contain the configuration options for the Salesforce UI.

Deploying VMWare OVA

About this task

The following procedure must be performed using VSphere VCenter Client to deploy VMWare OVA.

Important:

It is recommended that you use a thin client to deploy the Avaya CRM Connector 2.2 OVA file.

Procedure

1. Download the OVA file from PLDS.
2. Import the OVA into VMWare.

   A virtual machine is created following the specs described in the Avaya Platform Requirements section.
3. Access the newly created virtual machine console.
4. Log into the VMWare using username as **root** and password as **Avaya@123.**
5. Use the **Network Configuration Text User Interface** (**nmtui)** to set the IPs and FQDN designated for the new virtual machine.

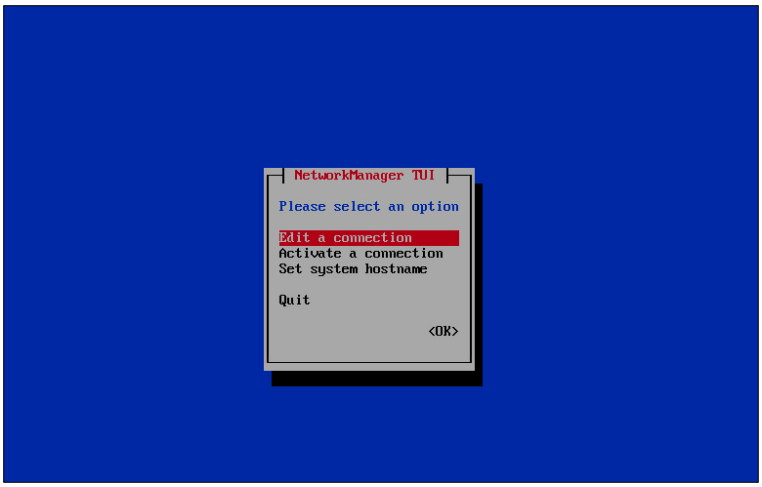

Figure 2: Network Manager

Select the configuration suited for the network from the given options: **Manual Static IP assignment** or **Automatic DHCP IP assignment**.

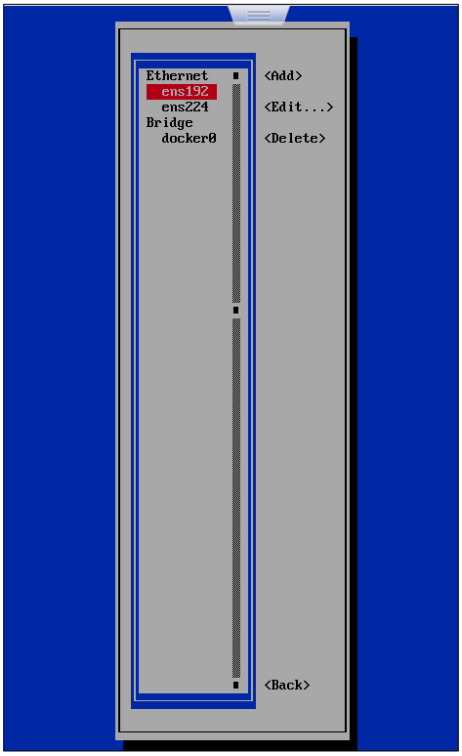

Figure 3: Ethernet selection

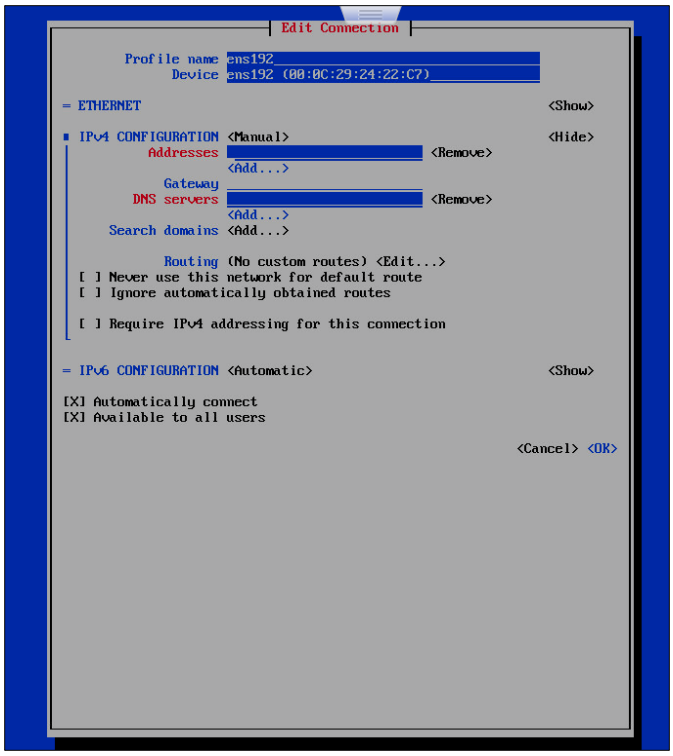

Figure 4: Edit Connection details

Set the host name with full qualified domain name (FQDN).

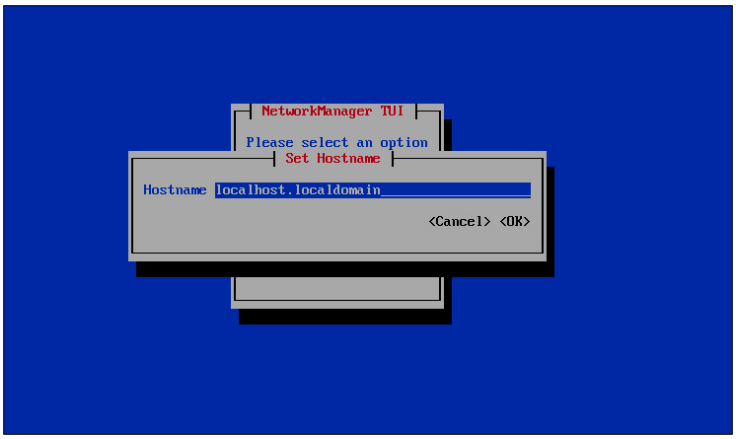

Figure 5: Set Hostname.

Activate the connections before exiting the window.

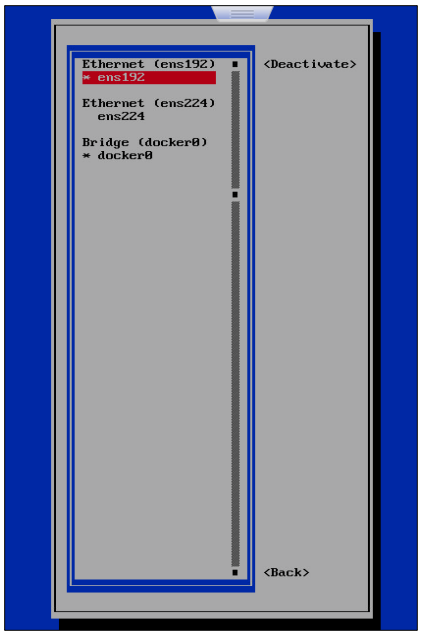

Figure 6: Activate connections

Click **OK** to exit.

Note:

1. ens192 is the primary NIC intended for public access.

2. ens224 is the secondary NIC intended for Out of Bound Management.

3. Based on the customer networking requirements, additional NICs can be added and configured.

4. Change or delete the password for root login using the following procedure:

   - Run the following command to set a stronger password:

     `passwd root`

   - Run the following command to delete the password-based access for root:

     `passwd --delete root`

Note:

a. Deleting the password will allow only EASG based access.

b. SSH access for **`root`** and **`sroot`** accounts is blocked.

c. SSH access will need to be performed with the craft user and later **`su - sroot`** will need to be performed to get the root access.

d. Both **`sroot`** and craft uses EASG to generate challenge responses.

e. **`root`** access is only possible when you are directly accessing the console.

f. SSH requires EASG.

Deploying the Server Certificate

The CRM Connector requires a server certificate for the Docker applications to be able to use HTTPS connections. You can install any of the following types of certificates on the server:

- Self-signed certificate
- Third-party or signed certificate from a request file
- Signed certificate with a private key

Whichever certificate is chosen, it will then be used by all the applications on the server.

Self-Signed Certificate

A self-signed certificate is created and signed by itself. Self-signed certificates do not contain a valid signature. As a result, whenever the client uses the CRM Connector, a security warning is displayed. Alternatively, to avoid the security warning, the certificate can be imported by each user's certificate store as a trusted root certificate. While acceptable during initial testing, this is not recommended for long term or production use. It is expected that any self-signed certificate used will be replaced with a signed certificate.

Creating a self-signed certificate and the associated files does not require root privileges. The self-signed certificate can be created using the default user.

Procedure

1. Create a new directory.
   For example, **`/opt/avaya/certs`**

1. Run the following command to navigate to the newly created directory:

   `cd /opt/avaya/certs`

2. Create a file called **openssl.cnf** with the following content:

```
[ req ]
default_bits = 2048
distinguished_name = dn
prompt = no
default_md = sha256
x509_extensions = v3_req
req_extensions = v3_req
extensions = v3_req
copy_extensions = copy

[ dn ]
countryName = Organization Country
stateOrProvinceName = Organization State
localityName = Organization Location
organizationName = Organization Name
commonName = server.domain.com
```

```
[ v3_req ]
basicConstraints = CA:FALSE
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
subjectAltName = @alt_names

[ alt_names ]
DNS.1 = server.domain.com
DNS.2 = loadbalancer.domain.com
```

3. Please make sure you change the bold selected text to what you need. The commonName is used to be able to identify the certificate when loaded into a client's certificate store. Without this, it can be hard to identify. The subjectAltName DNS value, must be set to the server's FQDN.

   Note

   The SAN (subjectAltName) is very important. It must match across the server and the corporate DNS, and the URL used to access the server. The server must recognize the name as its own. The name must be registered in the corporate DNS, so that it can be resolved by all of the user's workstations. The URL configured in the call center definition must use this name. All these names must match the SAN configured in the certificate.

4. Create the private key (with no password) and the self-signed certificate in a single command:

   openssl req -config openssl.cnf -new -extensions v3_req -x509 -sha256 -newkey rsa:2048 -nodes -keyout private_key.pem -days 365 -out certificate.crt

   To change the amount of time the certificate will be valid, adjust the value for `days`.

   This command creates the key (`private_key.pem`) and creates the self-signed certificate (**certificate.crt**).

5. Verify the certificate by running the following command:

   openssl x509 -text -noout -in certificate.crt

   This will print out the text of the certificate. When examining the output, the third line should show:

   Version: 3 (0x2)
   And farther down we should see:

   X509v3 Subject Alternative Name:

   DNS:server.domain.com

6. Run the following command to generate the `keystore.p12` file:

   openssl pkcs12 -export -in certificate.crt -inkey private_key.pem -out keystore.p12 -name avaya_crm_connector

Note

-name value must equal the value of the **server.ssl.key-alias** field set in all yml files.

7. Type a password and type the same password as a confirmation password to create the file.
8. Run the following command to copy the keystore file into the security directory:

   cp keystore.p12 /opt/avaya/avayacrmconnector/security

   Note:

   The keystore.p12 file already exists and must be overwritten when the command in step 8 is executed.

Now the certificate (in the keystore) is ready for use by Docker. The name of the keystore file (if different from above) and the password used must be retained and entered into the configuration files when designated below.

You must copy a self-signed certificate into each user's workstation. The self-signed certificate should be imported into the Windows Certificate Store for Internet Explorer and Google Chrome or imported as an exception directly into Firefox. This allows the browser to treat the self-signed certificate as a genuine certificate from a trusted entity.

For the self-signed certificate to be properly recognized when imported into the Windows Certificate Store, the certificate must be installed in the Trusted Root Certification Authorities folder. Placing the self-signed certificate in any other store will not allow it to be recognized.

These actions are not required for a signed certificate.

Signed Certificate

Creating a signed certificate and the associated files does not require root privileges. A signed certificate can be created using the default user.

This process assumes a private key and certificate request file for the signed certificate are being created. Even if a self-signed certificate has been used prior to this, it is highly recommended that a completely new private key is used.

Note that this process has two components. First is the creation of the private key and certificate request. Second is the use of the returned signed certificate. How long it takes to get to the component depends on the IT security organization and process involved.

Procedure

1. Create a new directory.
   For example, **/opt/avaya/certs**
2. Run the following command to navigate to the newly created directory:

   cd /opt/avaya/certs

3. Create a file called **openssl.cnf** with the following content:

```
[ req ]
default_bits = 2048
distinguished_name = dn
prompt = no
```

```
default_md = sha256
x509_extensions = v3_req
req_extensions = v3_req
extensions = v3_req
copy_extensions = copy

[ dn ]
countryName = Organization Country
stateOrProvinceName = Organization State
localityName = Organization Location
organizationName = Organization Name
commonName = server.domain.com

[ v3_req ]
basicConstraints = CA:FALSE
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
subjectAltName = @alt_names

[ alt_names ]
DNS.1 = server.domain.com
DNS.2 = loadbalancer.domain.com
```

Please make sure you change the bold selected text to what you need. The commonName is used to be able to identify the certificate when loaded into a client's certificate store. Without this, it can be hard to identify. The subjectAltName DNS value, must be set to the server's FQDN.

Note

The SAN (subjectAltName) is very important. It must match across the server, the corporate DNS, and the URL used to access the CTI Engine. The server must recognize the name as its own. The name must be registered in the Corporate DNS, so it can be resolved by all the user's workstations. The URL configured in the call center definition must use this name. These names must match the SAN configured in the certificate.

4. Create the private key (with no password) and the certificate request file in a single command:

   openssl req -config openssl.cnf -new -extensions v3_req -sha256 -newkey rsa:2048 -nodes -keyout private_key.pem -out certificate_req.csr

   This command creates the key (**private_key.pem**) and creates the certificate request file (**certificate_req.csr**).

   It is recommended to make the name of the certificate request file be based on the server (e.g. **servername_req.csr**). That way when the request file is sent to get the certificate, it can be easily identified as to which server it applies.

5. Send the certificate request file to the IT security department so that the signed certificate can be issued. Depending on the processes involved, this could take anywhere from minutes to multiple days.

6. Once the certificate is received, place the file into the **/opt/avaya/certs** directory on the server. Make a copy of the file and name it **certificate.crt.**

   Note

   Use the Base64 encoding when creating the signed certificate. If the DER encoding is used, the certificate will not match up with the private key. To switch the encoding, use the following command:

   ```
   openssl x509 -inform der -in signed_cert.der -out certificate.crt
   ```

   Also, if the signed certificate is a compound certificate (meaning it includes the server certificate, plus one or more intermediate certificates), it is very important to make sure the certificates are ordered correctly. The first certificate must be the server certificate, followed by the intermediate certificates in order.

7. Run the following command to generate the **keystore.p12** file :

   ```
   openssl pkcs12 -export -in certificate.crt -inkey private_key.pem
   -out keystore.p12 -name avaya_crm_connector
   ```

Note

-name value must equal the value of the **server.ssl.key-alias** field set in all yml files.

8. Type a password and type the same password as a confirmation password to create the file.
9. Run the following command to copy the keystore file into the security directory:

   ```
   cp keystore.p12 /opt/avaya/avayacrmconnector/security
   ```

Now the certificate (in the keystore) is ready for use by Docker. The name of the keystore file (if different from above) and the password used must be retained and entered into the configuration files when designated below.

Signed Certificate with Key

In case where IT security teams do not allow you to create keys or certificate request files, then you must send the Server FQDNs to the security team to perform these actions. The IT security team will provide both private key and a signed certificate in the form of a single key store file. When sent this way, the key store file must be sent in the PKCS12 format. Such files will usually have the “pfx” extension. The key store will also have a password assigned to it, and that must also be provided to be used.

Fortunately, this is also the format we want the key store to be in, so all we must do is copy the file to the **/opt/avaya/avayacrmconnector/security** directory and then enter the name of the key store file, its password and its key alias into the configuration files when requested below.

Deploying and configuring Avaya CRM Connector

Avaya CRM Connector Services Configuration mechanism

The configuration of the Avaya CRM Connector services, such as aacc3pcc, iec, and aa4salesforce is based on YML (format) files. The files are maintained in a Git repository also provided as a service by Avaya CRM Connector. In this model, Git can be used as a backup store, an audit trail, and access control mechanism tool for configuration changes.

Consul service acts as a configuration manager and configuration delivery system to other Avaya CRM Connector services, automatically loading configuration from git, offering it to the services and refreshing it upon detecting changes.

Managing configuration in GIT

The configuration of Avaya CRM Connector services, is maintained in a Git repository provided as a service by Avaya CRM Connector which must be up and running to allow the configuration to be gotten, changed and updated. To verify the Git (git-server) service status `/opt/avaya/avayacrmconnector/avayacrmconnectorctl services-status`, look for the service name that contains git-server and check if the status is `RUNNING`

Once the service is up and running, steps required by the Git client used to interact with the Git repository must have been performed. These steps are listed in the next sections.

Generating the SSH keys

About this task

SSH keys/security identities required by the git-server service to permit client access must be placed in the security directory. This is required before a cloning/getting the configuration from the repository.

Note:

git clone will not work until Docker is started.

Procedure

1. Run the following command to generate the security identity files:

   ```
   ssh-keygen -f /root/.ssh/id_rsa -t rsa -N ''
   ```

2. Run the following command to copy security identity files to Avaya CRM Connector:

   ```
   sh /opt/avaya/avayacrmconnector/keygen.sh
   ```

3. Run the following to reload the security configuration:

   ```
   /opt/avaya/avayacrmconnector/avayacrmconnectorctl reload
   ```

Note:

1. In case new Security Identities and SSH Keys need to be re-generated, the following command must be executed:
   ```
   ssh-keygen -f /root/.ssh/id_rsa -t rsa -N ''
   ```

2. In case the security identity files need to be updated, the following commands must be executed to clean up the previous security identities:

   a. `rm -rf /opt/avaya/avayacrmconnector/security/.ssh`

   b. `/opt/avaya/avayacrmconnector/keygen.sh`

   c. `/opt/avaya/avayacrmconnector/avayacrmconnectorctl reload`

Cloning the GIT repository

About this task

Cloning the configuration repository allows access to the configuration files. Usually this procedure is performed once, making the files available to be changed multiple times. After changing the configuration files, commit and push commands must be executed to update the configuration repository which will leverage other application services to perform the changes on the running services.

Procedure

1. Run the following command to clone the config Git repository:

   ```
   git clone ssh://git@localhost:2222/git-server/repos/config.git
   ```

   OR

   ```
   git clone ssh://git@<<HOST_IP>>:2222/git-server/repos/config.git
   ```

2. You may encounter the following error while cloning the repository that indicates missing SSH Key for your user account. Ensure that you have sufficient access rights to perform the cloning actions. For more information, see Generating the SSH keys.

   ```
   git clone ssh://git@localhost:2222/git-server/repos/config.git
   Cloning into 'config'...
   git@localhost: Permission denied (publickey,keyboard-
   interactive).
   fatal: Could not read from remote repository.
   Please make sure you have the correct access rights
   and the repository exists.
   ```

   Note:

   A new directory containing a cloned working copy of configuration files will be created.

Changing the configuration

About this task

Changing configuration allows you to configure the application to match the environment needs.

Procedure

To make any configuration changes, do the following:

1. Access the git repository directory created by the git clone command:

   ```
   cd config
   ```

2. Edit the corresponding *.yml config file (details in Configuration files).
3. Run the following command to add the changes:

   ```
   git add .
   ```

4. Run the following command to commit the changes:

   ```
   git commit -m 'Your commit message'
   ```

5. Run the following command to push the changes to the remote Git server:

   ```
   git push
   ```

   Note:

   If the remote Git destination is not configured, you will receive the 'No configured push destination' error.

Note:

The changes made directly on the Consul UI will be lost as they are not synchronized to Git. However, you can use the Consul UI to view the current configuration on the K/V tab.

Configuration Mechanism

The configuration for the Docker services (aacc3pcc, iec, aa4salesforce) is file based. The YAML format configuration files are committed to a Git repository, and are loaded to Consul automatically upon changes detected. Using this model Git can be treated as the backing store, audit trail, and access control mechanism for configuration changes and Consul as the delivery mechanism.

In order to ensure a secure communication between the services, TLS connections to the rabbitmq is used. In order for this to work, you must specify the certificate and the private key for the RabbitMQ server. These are configured in the rabbitmq servers configuration file which is located at /opt/avaya/avayacrmconnector/rabbitmq/rabbitmq.conf where the following fields should be modified to contain the server certification:

Configuration Key	Sample Configuration Value	Description
ssl_options.password	password	The password for the private key file
ssl_options.keyfile	/security/rabbitmq.key	The server private key
ssl_options.certfile	/security/rabbitmq.cer	The server certificate in PEM format
ssl_options.cacertfile	/security/rabbitmq.cer	The certification authority for server certificate in PEM format

Configuration files

Find below the description of the configuration files for Avaya CRM Connector Services. There is a dedicated configuration file for each service, and a configuration file -*application.yml,* which contains common configuration for all services**.**

Note

The configuration files are stored in git. The steps to get access to them, prepare them to be changed and the actions to be performed after changing them are described in the section Managing configuration in GIT.

Note

The default configuration files used by the Avaya CRM Connector services on the first boot and whenever reloaded (system service unit reload) are stored at /opt/avaya/avayacrmconnector/config/git-server/default-config.

aa4salesforce-prod.yml

Configuration Key	Sample Configuration Value	Description
jhipster.cors.allowed-origins	"force.com, salesforce.com"	The list of allowed origins for CORS. * for all.
jhipster.cors.allowed-methods	"*"	The list of allowed methods for CORS. * for all.
jhipster.cors.allowed-headers	"*"	The list of allowed headers for CORS. * for all.
jhipster.cors.allow-credentials	true	Flag to control the Access-Control-Allow-Credentials header.
server.ssl.key-store	/security/keystore.p12	The keystore location that contains the server certificate to secure connections. Defaults to /security/keystore.p12. Keystore is created above. Update the file name if different.
server.ssl.key-store-password	password	Keystore password. Use the password created in the keystore creation process.
server.ssl.key-store-type	PKCS12	Keystore type.
server.ssl.key-alias	avaya_crm_connector	Alias for the server certificate.
server.ssl.cyphers	• TLS_ECDHE_RSA_WITH_	Cyphers for TLS communication.

Configuration Key	Sample Configuration Value	Description
	AES_256_GCM_SHA384 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 • TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 • TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	
enabled-protocols	TLSv1.2	TLS version

iec-prod.yml

Configuration Key	Sample Configuration Value	Description
jhipster.cors.allowed-origins	"*"	The list of allowed origins for CORS. * for all.
jhipster.cors.allowed-methods	"*"	The list of allowed methods for CORS. * for all.
jhipster.cors.allowed-headers	"*"	The list of allowed headers for CORS. * for all.
jhipster.cors.allow-credentials	true	Flag to control the Access-Control-Allow-Credentials header.
server.ssl.key-store	/security/keystore.p12	The keystore location that contains the server certificate to secure connections. Defaults to /security/keystore.p12 . Keystore is created above. Update the file name if different.
server.ssl.key-store-password	password	Keystore password. Use the password created in the keystore creation process.
server.ssl.key-store-type	PKCS12	Keystore type.
server.ssl.key-alias	avaya_crm_connector	Alias for the server certificate.
server.ssl.cyphers	• TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 • TLS_ECDHE_RSA_WITH_AES_	Cyphers for TLS communication.

Configuration Key	Sample Configuration Value	Description
	128_GCM_SHA256 • TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 • TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 • TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 • TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 • TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	
server.ssl.enabled-protocols	TLSv1.2	TLS version
application.inactivityTimeout	90	The time (in minutes) of user's inactivity after which a session is automatically closed. The default value is 90 minutes.
application.agentPassKey		A cipher key (seed) to encrypt or decrypt an agent's password.
application.aniMasking	A	An ANI masking configuration for the

Configuration Key	Sample Configuration Value	Description
		make call and the consult call features. Possible values: • A: This is available to any CM version but does not support SIP stations. • B: This is available to CM 6.x only and supports SIP stations. • C: This is available to CM 6.3.x and 7.x only and supports SIP stations.
application.smtpHost	172.29.64.153	IP address of the SMTP server to be used for sending emails.
application.senderEmail	TestEmail01@172.29.64.153	Email address used to send the emails.
application.senderEmailPass	123456	Password for 'senderEmail' email address configured in the previous field. If the user is created without password, then this property is not mandatory.
application.destinations	TestEmail02@172.29.64.153, TestEmail03@172.29.64.153	Contains the users who must receive the emails. The email addresses must be

Configuration Key	Sample Configuration Value	Description
		separated by comma’,’).
application.expirationDays	60	The number of days to receive notifications after which the certificate expires. If the value is lower than 30 days, then the value is overridden with the minimum value, that is 30 days.
application.keystorePath	/security/keystore.p12	The keystore location that contains the server certificate to secure connections. The path must also contain the filename. E.g. /security/keystore.p12
application.keystorePass	password	keystore password
application.alias	avaya_crm_connector	Alias for the server certificate. Value must be copy of server.ssl.key-alias
application.loginTimeout	8000	In case the connector does not receive response from AACC server in the time configured, the login will be considered failed. **Note:** **It is suggested to add login timeout value for AACC as 16000.**

Configuration Key	Sample Configuration Value	Description
		8000 is the default value.
application.completeBlindTransferOffset	400	The amount of time after which the server will send the complete transfer request
application.waitForAcwInterval	400	The maximum amount of time the server waits for the ACW event before checking the agent state again
application.maxInterval	10000	The maximum period of time, in milliseconds, that the server waits for a new /meta/connect message from a client before that client is considered invalid and is removed.
application.survivingFirstCall	true	Indicates whether the first call is the one that survives when completing a consult call as conference or as transfer. For AACC server the surviving call is the first one.

aacc3pcc-prod.yml

Configuration Key	Sample Configuration Value	Description
application.aaccIp	10.10.1.10	The Avaya Aura Contact Center IP address. This can be a list of comma-separated IP addresses.

Configuration Key	Sample Configuration Value	Description
acwCode	acwDefaultCode	After Call Work default code. This value needs to be an existing configured value in AACC.

application.yml

Configuration Key	Sample Configuration Value	Description
application.weblmUrl	https://smgr.avaya.com/WebLM/LicenseServer	WebLM URL to check the license. For example: https://<weblm-server>:52233/WebLM/LicenseServer
application.trustStore	See *Sample configuration value: application.trustStore* for details.	List of certificates that must be treated as trusted ones during the secure connection attempts to the external server components: WebLM and AACC server. The definition must follow the syntax rules, otherwise the components will fail to read the configuration and fail to start as well: • Indentation must be the same for the keys and values on the same level. • *alias* must start with a dash (-) character, *certificate* does not. • *alias* value must be unique. • *Certificate* must be indented to start at the same horizontal position as *alias*. • The first line of the *certificate* value must contain: >- • The *certificate* content must be indented one level to the left from *certificate* (and *alias*). See *Sample configuration value: application.trustStore* for reference that follows all rules properly.
snmp.enabled	false	Enables SNMP supporting SNMP v2 and v3

Configuration Key	Sample Configuration Value	Description
snmp.trapDestinations	172.18.0.1/162	The address and port number of the SNMP listener
snmp.oid	.1.3.6.1.4.1.6889.2.97.1.0	SNMP message OID base for the Avaya Connector
snmp.community	avaya_crm_connector	The SNMP community advertised on the SNMP traps
snmp.username	username	The username to connect to the SNMP listener in case SNMP v3 is used
snmp.password	password	The password for the username mentioned in the previous field
snmp.passphrase	passphrase	The passphrase for the SNMP v3 secure connection

Sample configuration value: application.trustStore

```
application:
    truststore:
        - alias: aacc_cert
          certificate: >-
            -----BEGIN CERTIFICATE-----
            MIICQjCCAaugAwIBAgIIXTQ6zKLDOCMwDQYJKoZIhvcNAQEFBQAwMTEQMA4GA1UE
            AwwHZGVmYXVsdDENMAsGA1UECwwETUdNVDEQMAwGA1UECgwFQVZBWUEwHhcNMTQw
            NTA4MTUwNzQ3WhcNMjQwNTA1MTUwNzQ3WjAxMRAwDgYDVQQDDAdkZWZhdWx0MQ0w
            CwYDVQQLDARNR01UMQ4wDAYDVQQKDAVBVkFZQTCBnzANBgkqhkiG9w0BAQEFAAOB
            jQAwgYkCgYEAnqiMDutxsXSEIx8LsgWvHk13UFw265wu1H38lUSnrwu6rt3kxkSF
            2XMem1w+m8XLXsXakfbEKkiOK/eHP/cHoca+t5WzQwauL7z7LTnXy8+xaaRB9P4Y
            mUMzg10SUwqpbdtTAbhMvN6Tilqd3IesL4nCQNng4ToYLO2+RuNZuP8CAwEAAaNj
            MGEwHQYDVR0OBBYEFImNYn6bO0A7vb2GQZSoOTd+Dd9GMA8GA1UdEwEB/wQFMAMB
            Af8wHwYDVR0jBBgwFoAUiY1ifps7QDu9vYZBlKg5N34N30YwDgYDVR0PAQH/BAQD
            AgGGMA0GCSqGSIb3DQEBBQUAA4GBADKZX+TdXN768nReHgvnB0WWD28oaGw4sQc4
            ar8Ge5OqUXY60dFPTdavcOAakmRPoBQ7bG0oceC61AX/r+sftWuV5k2VPNsn+T35
            3xoFVd+6HykySeCTtSl5wYisfdohpZcPXmyzGudUtTeB9+Mj81jMIDvqoe6yebQO
            1RxZ0nrg
            -----END CERTIFICATE-----
```

Figure 7: Truststore certificate

Certificate Management using YML files

Each component in the solution that has an image will have a corresponding YML file which contains the initial configuration. These YML files are loaded into the Consul component which itself runs as a container in the solution when Docker compose is started. The initial

configuration which includes CORS filters, configuration for SSL, and certificates can be configured in the YML files.

Three components in the solution expose an HTTP interface and which must be secured:

- The AA4salesforce which is the softphone component
- The IEC

HTTPS must be enabled for these three components. The configuration pointing to a keystore file that resides in the security directory is defined in the YML file. The system retrieves the certificate from the defined path in the YML file.

The Consul loader also retrieves the contents of the YML files. The certificates are loaded after the installation in the Consul UI.

Enabling log handling

About this task

WARNING!

Starting from 2.2.7.1 release, all application logging is disabled by default. To benefit from log diagnosis, the logging feature must be re-enabled.

Procedure

1. Open the **`docker-compose.yml`** file located in **`/opt/avaya/avayacrmconnector`** directory.
2. For each service in the file, such as **`aa4salesforce-app`**, **`iec-app`**, **`oec-app`**, **`aes3pcc-app`**, **`pomdriver-app`**, **`rabbitmq1`**, **`rabbitmq2`**, **`consul`**, **`consul-config-loader`**, and **`git-server`** there is a section called **logging** which looks like this:

   ```
   logging:
         driver: "none"
   ```

3. To re-enable the logging, delete these two lines or comment them for each of the services mentioned above.

Avaya CRM Connector for AACC Mode

Avaya CRM Connector for AACC supports only AACC-Only mode. In this mode, the connector acts as an endpoint that performs third party control of a device on behalf of the agent or unified communication user.

To set the running mode as aacc-only, you must run the **`setmode.sh`** script located at **`/opt/avaya/avayacrmconnector/`**

Avaya CRM Connector Service

Avaya CRM Connector is provided as a System D Service Unit, which is enabled by default.

To disable the service, run the following command (single line):

- `systemctl disable /opt/avaya/avayacrmconnector/avayacrmconnector.service`

To enable the service, run the following command (single line):

- `systemctl enable /opt/avaya/avayacrmconnector/avayacrmconnector.service`

Standard Start, Stop, Restart and Reload operations can be performed by running one of the options depicted in the examples below:

- `systemctl stop avayacrmconnector`
- `systemctl stop avayacrmconnector.service`
- `service avayacrmconnector stop`

In addition, Avaya CRM Connector offers a utility command line interface which is the preferred way to interact with the application and can be used to control the application services and collect status and logs. Details about usage are shown below:

`/opt/avaya/avayacrmconnector/avayacrmconnectorctl [command][options]`

where commands can be:

Command	Description
start	Start the services
stop	Stop the services
restart	Restart the services
reload	Reload the service after a configuration change
status	Display services status from System (systemctl)
services-status	Display micro services status from Docker
processes-status	Display processes status from Docker
print-logs [microservice-name] [options]	Print logs on the screen. • If no parameter is passed, the system prints all micro services logs. • If a micro service name is passed, the system prints the logs for the micro service. • Additionally, other docker-compose logs options can be passed, such as: • --no-color: Produce monochrome output

Command	Description
	• -f, --follow: Follow log output • -t, --timestamps: Show timestamps • --tail="all" or --tail="<number>": Number of lines to show from the end of the logs. For example : ./avayacrmconnectorctl print-logs –tail= “100” iec-app
save-logs [microservice-name] [options]	Saves the logs in **`/var/log/avaya/avayacrmconnector/`** in a file named following the pattern ***avayacrmconnector-[microservice-name | all]-timestamp.log*** • If no parameter is passed, the system saves all micro services logs. • If a micro service name is passed, the system saves the logs for the micro service. • Additionally, other docker-compose logs options can be passed, such as: • -f, --follow: Follow log output • -t, --timestamps: Show timestamps --tail="all" or --tail="<number>": Number of lines to show from the end of the logs. For example: ./avayacrmconnectorctl save-logs –tail= “100” iec-app

Updating Docker images on an existing OVA

Procedure

4. Copy the package artifact (CRM-Connectors-2.2-SNAPSHOT-XY.tar.gz) from Avaya support website to the server as shown in the following example:

   ```
   scp CRM-Connectors-2.2-SNAPSHOT-XY.tar.gz
   craft@10.130.89.89:/home/craft
   ```

5. Back up the current images so they can be reloaded if required.
6. As a privileged user (sroot), run the following commands:

   a. `systemctl stop avayacrmconnector.service`

```
b. tar -xzvf CRM-Connectors-2.2-SNAPSHOT-XY.tar.gz -C
   /opt/avaya/avayacrmconnector/images/ images
c. cd /opt/avaya/avayacrmconnector/images/
d. docker container prune (delete old containers from docker)
e. docker image prune –a (delete old images from docker)
f. service docker restart (restart docker)
g. cd ..
h. ./load-images.sh
i. update configuration files(.yml files).
j. systemctl start avayacrmconnector.service
```

Note: For the upgrade from any 2.1.0.x or 2.1.1.x to 2.1.6.x or 2.2, the following steps are required:

- Update all the docker-compose.yml files from /opt/avaya/avayacrmconnector/mode-templates. After updating all docker-compose files, run /opt/avaya/avayacrmconnector/setmode.sh
- Update rabbitmq.conf file from /opt/avaya/avayacrmconnector/rabbitmq to add security settings (ssl_options.cacertfile, ssl_options.certfile, ssl_options.keyfile, ssl_options.password)
- Create rabbitmq certificate and key with the path and name specified in rabbitmq.conf file(ssl_options.cacertfile and ssl_options.keyfile properties) (Check here) and add it into application.yml file.

Customization through JavaScript files

Avaya CRM Connector design can be used for the customization needed in the connector to support the following:

- Load external JavaScript files.
- Migrate Screen pop.
- Frequently dialed numbers.
- VIP icon in the call card.

Load external JavaScript files

The loading mechanism in the AA4SFDC module will be modified to allow specific external javascript files to be loaded when the connector UI is requested to the server.

Extensions directory

The external JavaScript files are stored in a new directory (called “extensions”) located at the following connector installation path:

- /opt/avaya/avayacrmconnector/extensions

The custom files have a fixed name, as mentioned in the following list:

- FrequentlyDialedNumbers.js
- IconCallCardCustom.js
- iconCallCardCustom.png
- PopupProxyExternal.js

Creating SSH users without privileged access on CRM Connector servers

Starting with 2.2.7.4, users with fewer rights can be created for the production servers using **make-standard-user.sh** file.

The script from this file can be run after uploading it on the CRM Connector server and providing to it full rights. This can be done using the following command by a user with privileged rights or root: **`chmod 777 make-standard-user.sh`**).

This script must be run using **`./make-standard-user.sh`** in the file location followed by the wished USERNAME, PASSWORD and GROUPNAME as arguments.

The users created with this script will be able to:

- save logs
- to start, stop, restart or reload the application using the docker-compose commands (./avayacrmconnectorctl will not work for users without privileges)

These users will not be able to change, edit or remove the project folder and its elements.

Salesforce Configuration

Prerequisites

To start using Avaya CRM Connector in Salesforce, you must first perform the following tasks:

No.	Task	Reference	Notes	✔
1	Install the APEX package	See Installing APEX Package		
2	Configure the Call Center Detail components	See Configuring the Call Center Definition settings		

Installing the APEX Package

Before you begin

Log in as a Salesforce administrator.

Procedure

1. Install the package **for all users** for a production organization by using the following URL:

   https://login.salesforce.com/packaging/installPackage.apexp?p0=04t2M000002ilx3

   OR

   https://test.salesforce.com/packaging/installPackage.apexp?p0=04t2M000002ilx3

   for sandbox.

   If you are not logged in as a Salesforce administrator, the system displays the Salesforce login page.
2. Provide the Salesforce administrator login credentials when prompted.

   The system displays the package page.
3. Ensure that the **Install for All Users** option is selected and click **Install**.
4. After the installation is completed, click **Done**.

   The system installs the APEX package which contains the following component:

   Apex Class: CustomCallLogField

   Apex Class: DataRetrieval

   Apex Class: UserRetrieval

   Apex Class: ObjectDetailsRetrieval

   Apex Class: SalesforceDirectory

   Apex Class: CustomCallLogFieldUnitTest

   Apex Class: DataRetrievalUnitTest

   Apex Class: UserRetrievalUnitTest

   Apex Class: ObjectDetailsRetrievalUnitTest

   Apex Class: SalesforceDirectoryUnitTest

Grant access to Apex Classes

1. Go to **Setup** > **Apex Classes**.
2. Open the apex class that we want to modify.

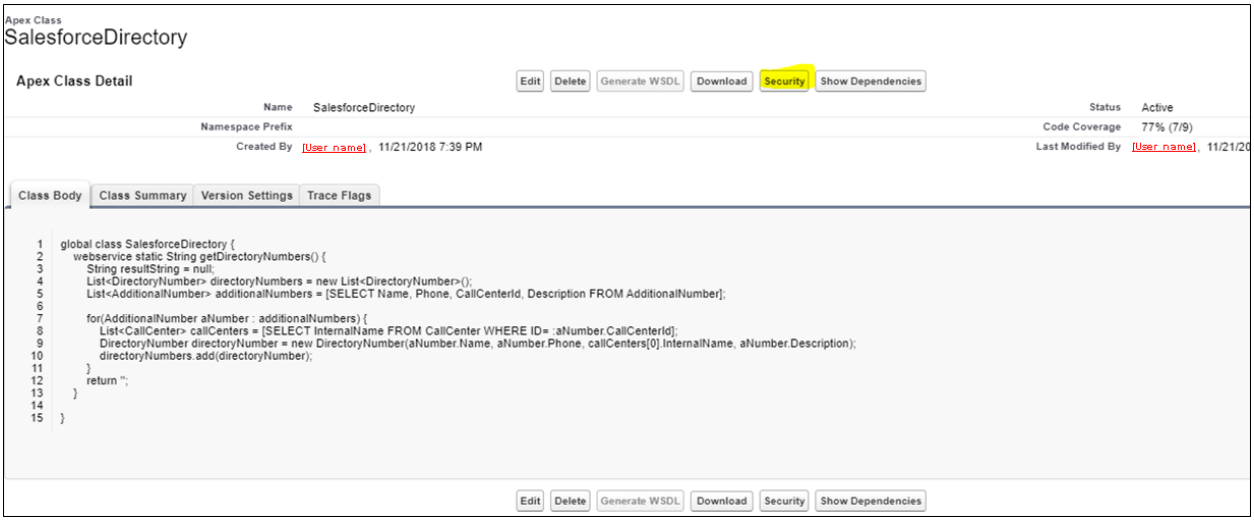

Figure 8: Apex Class Detail page

3. Click **Security**.

   The system displays the Enable Profile Access for Apex Class page.

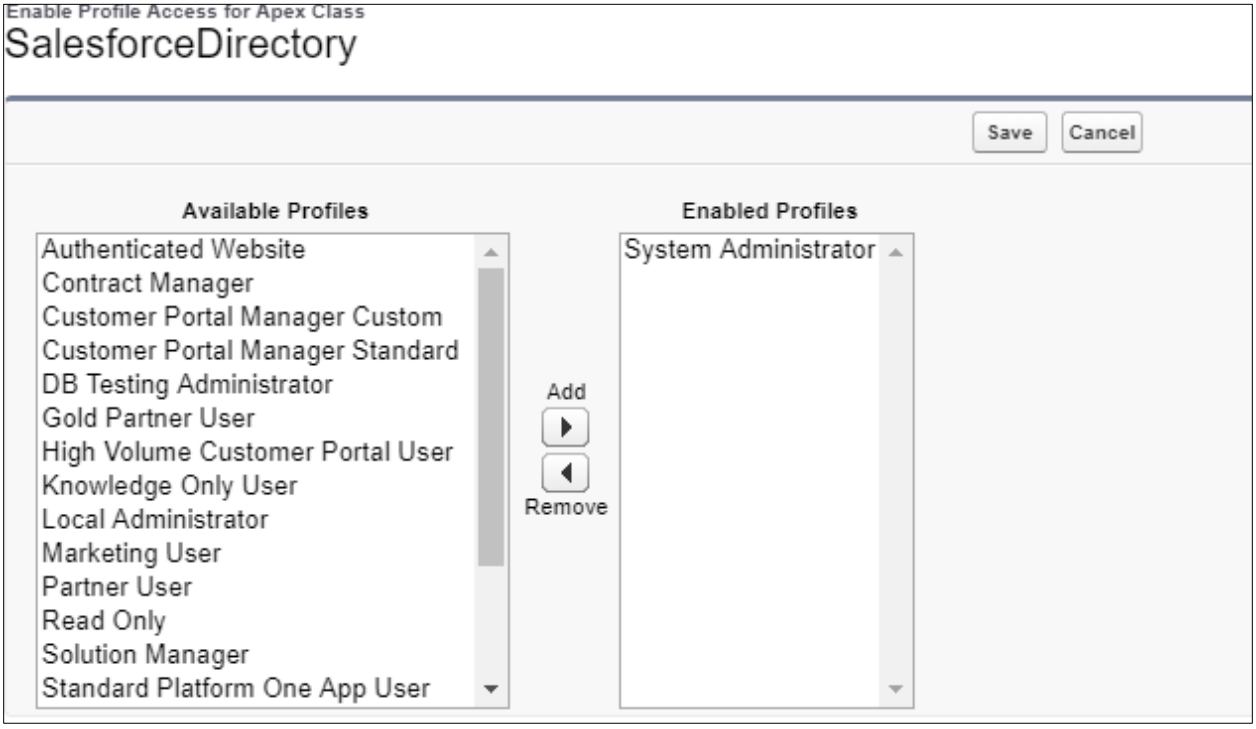

Figure 9: Enable Profile Access for Apex Class page

4. Add the required profiles to the **Enabled Profiles** list.
5. Click **Save**.
   These steps must be repeated for all APEX classes.

Call Log Fields visibility

The call log fields must be visible to the agent that is logged in on Salesforce.

1. Go to **Setup** > **Object Manager** > **Task** > **Fields&Relantionships** > <Call_log_field>(in this case Type) > **Set Field-Level Security**
2. Select the **Visible** column check box next to the desired profile.
3. Ensure that the **Read-Only** field is cleared for the profiles selected in the previous step.

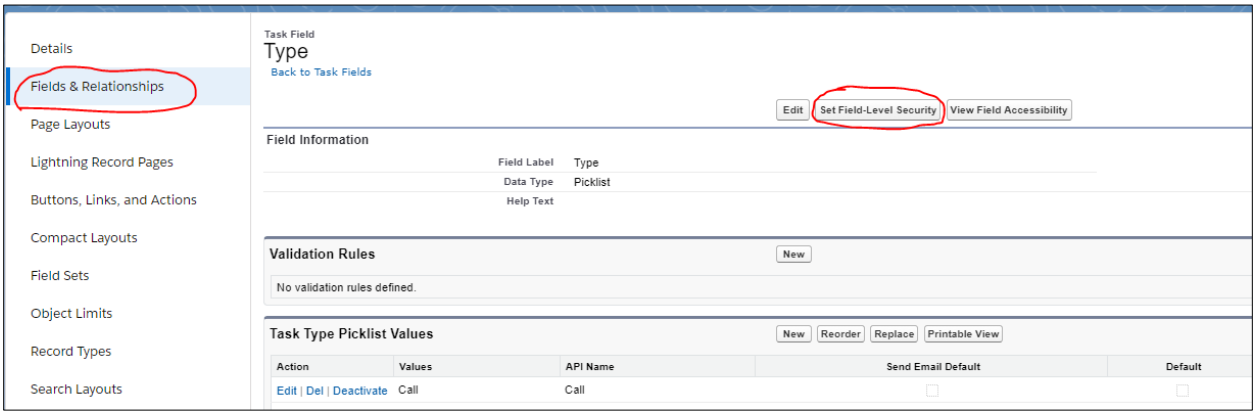

Figure 10: Type field page

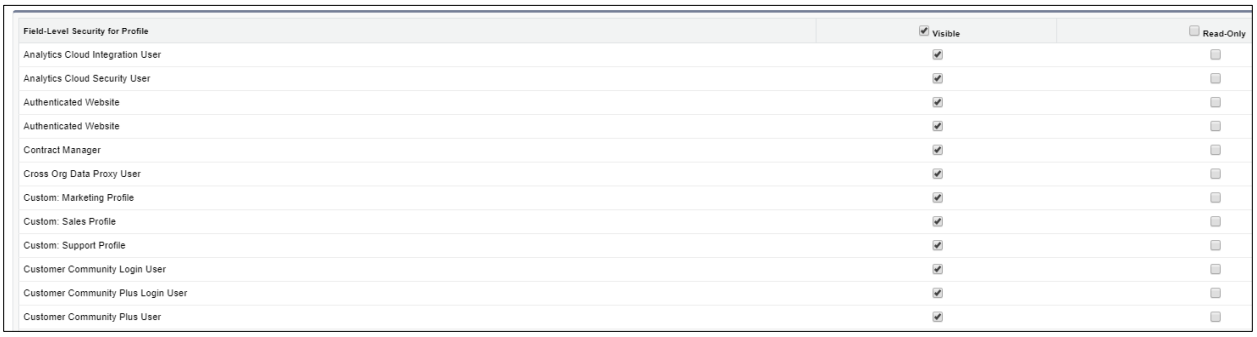

Figure 11: Field-Level Security page

Creating the activity custom fields for call logs

About this task

Using the Custom Call Log fields, you can enable enhanced call logging feature which saves additional information into the call log record. Salesforce handles these fields as Activity Custom Fields. The following procedure is optional in context of Avaya CRM Connectors 2.2 and must be performed only in case the customer wants to save the activity information in the call logs.

Procedure

1. Navigate to **<Username> > Setup > App Setup > Customize > Activities > Activity Custom Fields**.

2. Click **New** and create the activities for each of the following custom fields:

Purpose	Suggested field name and label Name	Suggested description and help text	Data type	Suggested API name
Caller ID (ANI)	Caller	Caller's Phone Number (ANI)	Phone	Caller__c
Called Number (DNIS)	Called	Called Phone Number (DNIS)	Phone	Called__c
UUI (User To User Information)	UUI	User to User Information (UUI)	Text(96)	UUI__c
Queue (for ACD Calls)	Queue	ACD Queue	Text(20)	Queue__c
Queue Name (for ACD Calls)	Queue Name	ACD Queue Name	Text(96)	QueueName__c
Original UCID	Original Call Object ID	Original Call Object ID	Text(20)	OriginalCallObject Id__c
Call Start Time	Call Start Time	Call Start Time	Date/Time	CallStartTime__c
Agent ID	Agent ID	Agent ID	Text(20)	AgentId__c
Call ID	Call ID	Call ID	Text(9)	CallId__c
Old Call ID	Old Call ID	Old Call ID	Text(9)	OldCallId__c

3. Follow the table and the on-screen instructions to complete the settings.
4. Click **Save** and repeat steps for all custom fields in the table above.

The Custom Call Log fields allow users to specify up to three log details fields in the configuration. Users can define the name of the custom Activity field, and whether it is a text box or Picklist.

These fields must be created in Salesforce (Setup>Object Manager>Activity>Fields & Relationships). The template contains "Field Label", "Field Name", and "Data Type", where in "Data Type" must be specified if the custom field is a text or a picklist.

If it is a picklist, then the configuration also stores the available options. When the call log is saved, the value is saved into the specified custom Activity field. The custom field type text is also saved in logs.

For picklist is important to have its options activated in Salesforce Fields & Relationships (Setup>Object Manager>Activity>Fields & Relationships) to be able to use it in CCDef.

Example for settings in CCDef:

- Text field example:

  Custom Call Log Field 1: Description:Description__c

- Pick List example:

  Custom Call Log Field 2: Category:Category__c

Configuring the Call Center Definition settings

About this task

Salesforce Call Centers can be created by importing and configuring the Call Center Definition (CCD) XML file into Salesforce.

Procedure

1. Navigate to **<Username> > Setup > App Setup > Customize > Call Center > Call Centers** OR

   Type **Call Centers** in the search field on the left pane. (Do not click the Enter key.)

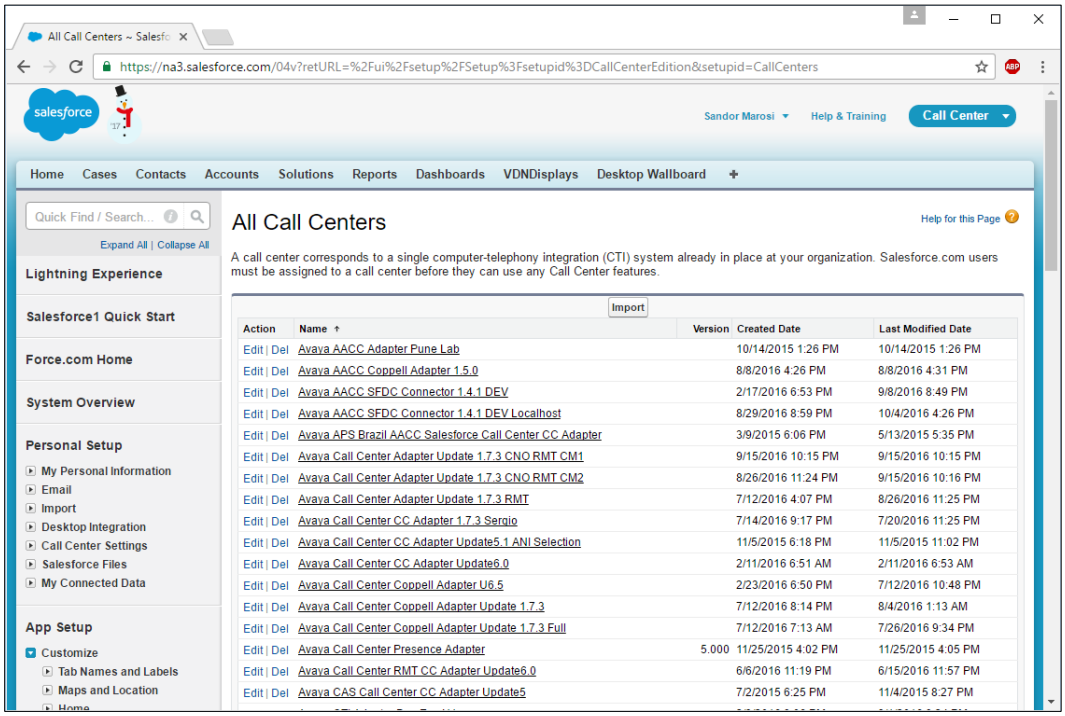

Figure 12: All Call Centers

2. (**Optional**) If the system displays a help page, click **Continue**.

   The system displays the CCD Import page.

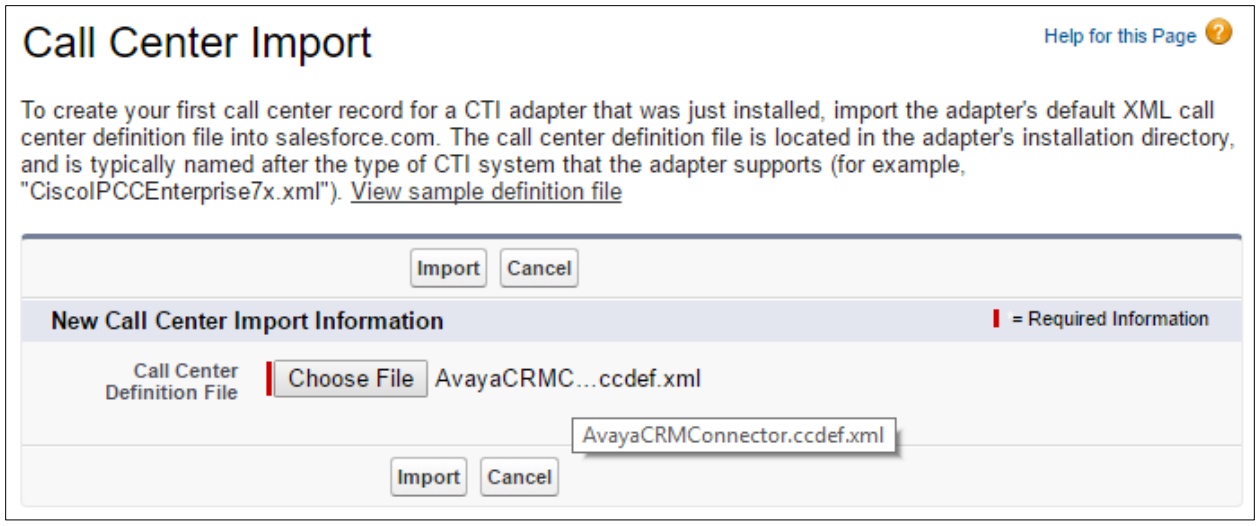

Figure 13: CCD Import

3. Click **Import**.
4. Click **Choose File** and select the desired CCD file.
5. Click **Import** again.
   The system adds the new CCD file in the list of Call Centers.

Note:

Sometimes the import action might fail due to the following two reasons:

1. There is no more room in the table that holds the Call Center Definition. In this case, you must delete an old Call Center Definition and import a new one.
2. The Internal Name of the Call Center Definition being imported matches an already existent Call Center Definition. Either edit the XML file to have a different Internal Name or change the conflicting name in the existent Call Center Definition.

Call Center components general descriptions

You must configure the following settings of the Call Center Detail component:

- General Information
- Dialing Options
- Softphone Options
- Call Log Options
- Label Options
- Screen Pop Options
- Reason Codes
- CometD Server Configuration

You must add information in the required fields as explained in the field descriptions sections of this guide.

General Information settings field descriptions

Name	Default	Description
Internal Name and **Display Name**	CRMConnector21	The fields to uniquely identify the Call Center definition in Salesforce. If multiple Call Center definitions are present in a Salesforce instance, each name must have a unique Internal Name and Display Name.
Display Name	Avaya CRM Connector 2.2	The field to define the name of a CCD for identifying the CCD in the list of CCD files.
Web Agent Widget URL	https://<Server IP address>:8484	The Web Agent Widget URL indicates the entry point for the web application letting the browser fetch it from the Web Agent Salesforce Connector Server. The URL pattern follows: *https://<server>:<port>* Where <*server*> is the Server Cluster IP or fully qualified domain name (FQDN).
Use CTI API	True	An option to use CTI API in Call Center Definition. You must always keep this value as True.
Softphone Height	600	These properties are used by Salesforce to define the pixels required by each dimension to properly display the Avaya CRM Connector web page. **Note** • In Salesforce.com, **Standard View** and **Console mode**, the recommended height to properly display the softphone is 560 pixels. In Lightning mode, the recommended height is 600 pixels. • In Salesforce.com, **Standard View**, the width is fixed as 200 pixels and cannot be changed. The height value can be adjusted
Softphone Width	260	

Name	Default	Description
		to allow the widget to display properly.
Salesforce Compatibility Mode	Classic_and_Lightning	Determines the settings where the softphone is visible. **Note** • To display the softphone in Lightning Experience, select **Lightning**. • To display the softphone in Salesforce Classic, select **Classic**. • To display the softphone in both user interfaces, select **Classic_and_Lightning**.

Dialing Options field descriptions

Name	Default	Description
Outside Prefix	9	**Outside Prefix** is the number used to get external dial tone. **Long Distance Prefix** is the number needed to indicate a long-distance call. **International Prefix** is the number needed to indicate an international call.
Long Distance Prefix	1	
International Prefix	011	
Internal Number Length	4	A field to specify the type of number: The options are: • **Internal number**: If the number of digits in a phone number is equal to or less than the given value, then the number is treated as an internal number. • **Inbound** or **outbound external number**: If the number is greater than the given value, the number is treated as an inbound or outbound external number.

Name	Default	Description
		Note If this value is 0, then all numbers are treated as inbound or outbound external numbers. If the local dial plan uses extensions as long as the local number length, set this value to 0.
Country Code	1	A field to specify the dialing country code of the current location. This value is used for the enhanced outbound number processing as the default number processing follows US country code standard.
Local Number Length	7	A field to define the length of an external phone number. If 10-digit local numbers are used, then set this value to 9 and use the Communication Manager routing tables to finish handling the number.
Long Distance Length	10	A field to define the length of a long distance external phone number. These values are used to examine an outbound number to determine which prefixes to use.
Do Not Call Prefix		A field to define the Do Not Call Prefix for a number. If there is no default value added, this feature is not enabled. This feature is applicable for click-to-dial calls only.

Softphone Options field descriptions

Name	Default	Description
Is Call Center? (Y/N)	Y	A field to determine whether the ACD is used. The options are: • Y: ACD is used. The ACD agent ID and password fields are included in the login form. The default value is Y. • N: ACD is not used. The ACD agent ID and password fields are not included in

Name	Default	Description
		the login form and only the extension information is requested.
Transfer Button Enabled? (Y/N/C)	Y	A field to disable the option to transfer. The options are: • Y: The Transfer call capability is enabled. • N: The Transfer call capability is disabled. • C: The blind transfer button is hidden, but the user can perform a consultative transfer and the complete transfer button is still shown on the call card for the new call. • If the value is not Y, N or C, then default value is set on Y
Conference Button Enabled? (Y/N/C)	Y	A field to disable the option to conference. The options are: • Y: The Conference call capability is enabled. • N: The Conference call capability is disabled. • C: The blind conference button is hidden, but the user can perform a consultative conference and the complete conference button is still shown on the call card for the new call. • If the value is not Y, N or C, then default value is set on Y
Call Log Report URL	<Default_URL>	A field to define the URL to use when someone selects My Report Label. If no value is present, then the option is not shown. This is a partial URL. A default value is included in the call center definition XML file.
Click-to-Dial Enabled? (Y/N/P/I)	Y	A field to allow click-to-dial functionality. The options are: • Y: Click-to-dial is enabled.

Name	Default	Description
		• N: Click-to-dial is disabled. This will deactivate all the links in the Salesforce display and will not allow any click-to-dial functionality to work. If this is selected, the click-to-consult option will be ignored and set to "N". • P: Click-to-dial is enabled. However, when this message is received from Salesforce, we will **not** automatically place the call. Instead, the number received from Salesforce will be placed into "Type a Phone Number" field in the softphone. • I: Click-to-dial is enabled. When click-to-dial is used, a pop up with be displayed. The pop-up is just a secondary confirmation. • If the value is not Y, N, P or I, then default value is set on Y
Click-to-Consult Enabled? (Y/N/T/C)	Y	A field to extend the basic click-to-dial functionality to allow for click-to-consult. Click-to-dial only works if the phone is currently idle, or if all calls present are held. If there is a single active line, and click-to-dial is invoked, click-to-consult will automatically initiate a consultative call. Once initiated, the consultation can be completed as either a conference or a transfer, or it can be backed out to the original call. The options are: • Y: The option to enable click-to-consult call. If there is a single active line, and click-to-dial is used, then click-to-consult is automatically started. • N: The option to disable click-to-consult call. • T: Instead of doing a "consult" operation, attempt to do a full blind transfer operation.

Name	Default	Description
		• C: Instead of doing a "consult" operation, attempt to do a full blind conference operation.
Available Type (A/M)?	A	A field to specify the Available type. The options are: • Auto-In (A): Using this option will automatically move the user to an Available state after an ACD call. • Manual-In (M): Using this option will place the user in a Wrap-up state after an ACD call. The user must then manually set them to an Available state to receive another call.
Auxiliary Enabled? (Y/N)	Y	A field to allow users to set the Auxiliary state. The options are: • Y: The option to set Auxiliary state is enabled. • N: The option to set Auxiliary state is disabled.
After Call Work Enabled? (Y/N)	Y	A field to allow a user to manually set the After Call Work state. The options are: • Y: Allows user to set After Call Work manually. • N: Do not allow user to manually set the ACD state to After Call Work. It is possible for the user to end up in the After Call Work state automatically. For example, if the Available type used is Manual-In, or if Timed ACW is used with Auto-In, the user will automatically change to After Call Work even when set to N.
Auto Login Enabled? (Y/N)	N	A field to automatically log the user into the softphone. The options are: • Y: Automatic login is enabled. This option works only when the login credentials are already stored. If the credentials are not stored, the user is

Name	Default	Description
		not logged in automatically even when this setting is Y. • N: Automatic login is disabled.
Default Language	en_US	A field which defines the default language of the softphone application.
Password Enabled? (Y/N)	Y	Determines whether a Password field is presented in the login form. If set to N, the login form will only show Extension and Agent Id.
Extension from Salesforce User Profile? (Y/N)	N	A field to take the extension from Salesforce user profile. The options are: • Y: Extension is taken to Salesforce user profile and cannot be changed. If the setting in Salesforce is wrong or missing, the user will not be able to log in. • N: Extension must be entered manually into the login screen.
Enable Console Logout? (Y/N)	Y	A field to logout the extension from the Softphone in Salesforce Classic Console Mode logout. The options are: • Y: The extension will be automatically logged out of the ACD when the user logs out of the Classic Console. • N: The extension will remain logged in into the ACD even after logging out of Salesforce.
Display login time information? (Y/N)	N	A field to display the login time information. The options are: • Y: The login time information will be showed in the Softphone. • N: The login time information won't be showed in the Softphone
Drop Selected Party Enabled? (Y/N)	N	A field to handle the drop of parties in a conference. The options are:

Name	Default	Description
		• Y: The user must indicate which party wants to remove from the conference. • N: The user can remove only the last added party.
Automatically answer incoming calls? (Y/A/N)	N	A field to enable auto answer for incoming calls. The options are: • Y: All the incoming calls will be auto answered. • A: Only the ACD calls will be auto answered. • N: No calls will be auto answered.
Display hold timer? (Y/N)	N	A field to display a hold timer into the softphone. The options are: • Y: The hold timer will be shown in the softphone. • N: The hold timer won't be shown in the softphone.
Display Agent State timer? (Y/N)	N	A field to display a timer for the agent's current state. The options are: • Y: The Agent State timer will be showed in the softphone. • N: The Agent State timer will not be showed in the softphone.
Omnichannel Enabled? (C/S/N)	N	Enable integration with Salesforce Omnichannel when using Console mode. The options are: • C: Omnichannel complimentary mode is enabled. • S: Omnichannel synchronized mode is enabled. • N: Omnichannel integration is disabled.
Omnichannel Ready Status Id	<EMPTY>	In case of Salesforce Omnichannel Enabled, the system allows to select the id for Ready status. Accepts values separated with comma.

Name	Default	Description
Omnichannel Not Ready Status Id	<EMPTY>	In case of Salesforce Omnichannel Enabled, the system allows to select the id for Auxiliary status. Accepts values separated with comma.
Omnichannel Wrapup Status Id	<EMPTY>	In case of Salesforce Omnichannel Enabled, the system allows to select the id for WRAPUP status. This field can contain only a single value.
Omnichannel Not Ready Reason Code	21=Digital Ready	In case of Salesforce Omnichannel Enabled, the system allows to select the id and the text for the Reason Code to be used on Auxiliary status.
Show Device Name? (ADQ)	ADQ	If A/a-ANI or D/d-DNIS or Q/q-QUEUE is present, the corresponding device name is displayed on the call card when a call is either performed or received. If none is configured the ANI, DNIS, and QUEUE will be displayed instead of the name.
Append Agent ID to UUI? (Y/N)	Y	A field to append agent's id to UUI when the agent drops the call.
Outbound ANI Replacement	<EMPTY>	If this field is set for outbound calls, it replaces the ANI with the set value.
ANI Replacement WS URL	<EMPTY>	A field to define the URL of the application that returns the ANI replacement options. The options can be used to replace the ANI when performing an outbound call. This field will not have any effect if 'Outbound ANI Replacement' field is not empty.
ANI Masking on Consult? (Y/N)	N	If field is set to Y, it enables ANI replacement for consult, blind transfer, and blind conference calls. The 'Outbound ANI Replacement' or 'ANI Replacement WS URL' field must be set for this field to have effect.
Apex User Retrieval Class	AvayaConnector.UserRetrieval	Name of the User Retrieval APEX class.

Name	Default	Description
Apex Data Retrieval Class	AvayaConnector.DataRetrieval	Name of the Data Retrieval APEX class.
Apex Object Retrieval Class	AvayaConnector.ObjectDetailsRetrieval	Name of the Data Retrieval APEX class.
Apex Salesforce Directory Class	AvayaConnector.SalesforceDirectory	Name of the Salesforce Directory APEX class.
Apex Picklist Values Class	AvayaConnector.CustomCallLogField	Name of the Apex Class that works with the picklist options, to be able to use picklist as Custom field without mentioning all its options.
Warning Banner Message	<EMPTY>	A field to define a warning banner message that must be displayed to users on the login screen.
Show Softphone when click-to-dial? (Y/N)	Y	Show soft-phone in lightning mode when agent perform a click-to-dial
Configure number	40158	The number configured for voicemail. **Note:** Not relevant for CRM Connector with AACC.
Logout on timeout (Y/N)	N	If the option has the value set to Y and an agent is leaving without logging out their station will be automatically logged out when their inactivity timeout expires from the AACC. If the option is set to N the logout will be simulated on UI and IEC side and kept logged in AACC.

Call Log Options field descriptions

Name	Default	Description
Call Log Enabled?(Y/A/E/N)	Y	A field to enable call logging. The options are: • Y: Call logs are saved for all calls. • A: Call logs are saved for only ACD calls.

Name	Default	Description
		• E: Call logs are saved for only inbound and outbound external calls. • N: Call logs are not saved.
Show Call Log Y/N/S/A)?	Y	A field to allow users to view the call logs. The options are: • Y: Allows users to view the call logs. • N: Disallows users to view the call logs. • S: Display call logs details car expanded at the beginning of the call. Agent will be able to collapse/expand. • A: Call logs detail always expanded. Agent will not be able to collapse it.
Save Call Log on Call Start? (Y/N/V)	N	A field to enable call logging when the call status is in progress at the beginning of the call. The logs are later updated when the call is dropped. The options are: • Y: Call logs are saved at the beginning of the call and later updated. • N: Call logs are saved at the end of the call. The default value is N. • V: Call logs are saved at the beginning of the call and updated when the call is answered and at the end of the call. **Note** Unless this feature's capability is explicitly used, you must always set this to N.
Call Log on Incomplete Calls	N	A field to enable call logging for unanswered calls. The options are: • Y: Call logs are saved for unanswered calls. • N: Call logs are not saved for unanswered calls
Make Call Log Related Data Sticky	Y	Determines how the call log references are saved. The options are:

Name	Default	Description
		• Y: The application sets the reference to be saved to the first matching resource browsed. The user can always override the selection manually, but continued browsing does not update the selection. Only user intervention can change the selection. The last selection is saved. • N: The application sets the reference to be saved to the latest resource browsed. Every time the user browses to a new resource, the reference selection is updated. The user can override the selection, but any further browser will again update the reference.
Caller Field API Name	Caller__c	A field to define the Activity Custom Field API Name to save the Caller ID (ANI). If it is empty, this parameter will not be saved in the call log.
Called Field API Name	Called__c	A field to define the Activity Custom Field API Name to save the Called Number (DNIS). If it is empty, this parameter will not be saved in the call log.
UUI Field API Name	UUI__c	A field to define the Activity Custom Field API Name to save the UUI (User to User Information). If it is empty, this parameter will not be saved in the call log.
Queue Field API Name	Queue__c	A field to define the Activity Custom Field API Name to save the Queue for ACD Calls. If it is empty, this parameter will not be saved in the call log.
QueueName Field API Name	QueueName__c	A field to define the Activity Custom Field API Name to save the Queue Name for ACD Calls.
Original Call Object Id	OriginalCallObjectId__c	A field to define the Activity Custom Field API Name to save the Original UCID for transfer and conference call. In case is empty, this parameter won't be saved in the call log
Call Start Time API	CallStartTime__c	A field to define the Activity Custom Field API Name to save the time when the call start.

Name	Default	Description
Call History? (Y/N/0-20)	Y	A field to enable access to a button to show information about the last calls in which the user was involved. The options are: • Y: The user can view call history. • N: The user is not able to view call history. • 0: The user is not able to view Call History. • any number greater than 0, will enable the call history, and the number of calls will be equals with the number configured. Anything that is greater than 20, will be considered 20, this is the max value
Show Comments (Y/N)	Y	A field to indicate if the comments entry is displayed in the Log Details. The options are: • Y: The user can view and edit the comments for the call log. • N: The user is not able to view the comments entry and it will be saved empty.
Show Name (Y/N)	Y	A field to store the name of the caller in the logs. • Y: The user can view the name of the caller. • N: The user is not able to view the name of the caller.
Show Related To (Y/N)	Y	A field to store the name of the group/category the caller belongs to in the logs. • Y: The user can view the group/category name of the caller. • N: The user is not able to view the group/category name of the caller.
Enable Task/Call Log link (Y/N)	N	A field to indicate if a new call log will be created for a call placed from a click-to-call from a task. The options are: • Y: The task is reused for call log saving. • N: The task is not reused for call log saving; a new call log will be created.

Name	Default	Description
Update Task Status (Y/N)	N	If the previous option is enabled, this will determine if the Status is automatically set as Completed or if the user will have to manually set the Status. The options are: • Y: The call log status is set as Completed. • N: The call log status is not set. The user is responsible for later completing the task.
Enable UTC time (Y/N)	N	A field to allow the call log time to be saved in a UTC format instead of a local time. The options are: Y: The time in the UTC format is used to be saved in the call log. N: The local time is used to be saved in the call log.
Call Log Subject	<EMPTY>	This is the default subject shown in the Log Interactions display. If this field is empty, the default subject shown is “Call”.
Enable Interaction Log? (Y/N)	N	A field to integrate with the Salesforce Classic Console interaction log and not the call log. The options are: • Y: Enables integration with the Salesforce Console interaction log. This setting must be only when both Call Log Enabled and Show Call Log are set to N. • N: Disables integration with the Salesforce Console interaction log. **Note** Unless it is known that the Interaction Log is being used, you must always set this to N.
Custom Call Log Field 1 to 3	<EMPTY>	A set of fields to activate three custom fields in the call log. You can define two different types of fields:

Name	Default	Description
		• Text Field: Format = “Label:Variable”. This type of field shows textbox free form field. Example: “Description:Description__c” • Pick List: Format = “Label:Variable”. This type of field shows a drop-down list with a list of items to choose. Example: “Type:Type__c Where: • Label is the text to display at left of the field. • Variable is the Salesforce API name for the Task/Activity field. Its value is represented by the Field Name of the variable associated with the custom field created in Salesforce Fields & Relationships. The customer can define up to three custom call log fields in the call center definition. These custom fields have the following structure: <Field Label>:<Variable_Name> **Note:** By design, no value will be implicitly selected for the pick-list. It is important to have the pick list defined and its options activated in Salesforce Fields & Relationships (Setup > Object Manager > Activity > Fields & Relationships). The fields value can contain only the Variable. In this case, the label of this field will be automatically taken from Salesforce Fields & Relationships, being the Field Label of the variable associated with the custom field name.
Call Log Completed Status	Completed	A field to define the default text for a completed call log status.
Call Log In Progress Status	In Progress	A field to define the default text for a call log status which is in progress.

Name	Default	Description
		Note: The value of this field is used only when *Save Call Log on Call Start? (Y/N/V)* is set to Y and V.
Call Log Answered Status	Answered	A field to define the default text for a call log status which is answered. **Note:** The value of this field is used only when *Save Call Log on Call Start? (Y/N/V)* is set to V.
Prioritize Related To (Y/N)	N	The possible values are: • Y: If the Name value is a Lead, then the Lead value will not be saved, and the Related To value will be saved instead. • N: If the Name value is a Lead, then the Lead value will be saved in Name and the Related To value will not be saved.

Label Options field descriptions

Name	Default	Description
Login Extension	<EMPTY>	If provided, it will override the “Extension” label on the login form.
Login Agent ID	<EMPTY>	If provided, it will override the “Agent Id” label on the login form.
Login Agent Password	<EMPTY>	If provided, it will override the “Agent Password” label on the login form.
Login Accept Button	<EMPTY>	If provided, it will override the “Accept” button name on the login form.
Login Reset Button	<EMPTY>	If provided, it will override the “Accept” button name on the login form.
Available Label	<EMPTY>	If provided, it will override the “Available” option in the agent state drop-down list.

Name	Default	Description
Auxiliary Label	<EMPTY>	If provided, it will override the “Auxiliary” option in the agent state drop-down list.
After Call Work Label	<EMPTY>	If provided, it will override the “After Call Work” option in the agent state drop-down list.
Busy Call Label	<EMPTY>	If provided, it will override the “Busy Call” option in the agent state drop-down list.
Pending Auxiliary Label	<EMPTY>	If provided, it will override the “Pending Auxiliary” option in the agent state drop-down list.
Pending After Call Work Label	<EMPTY>	If provided, it will override the “Pending After Call Work” option in the agent state drop-down list.
On Call Label	<EMPTY>	If provided, it will override the “On Call” option in the agent state drop-down list.
Log Out Label	<EMPTY>	If provided, it will override the “Log Out” option in the agent state drop-down list.
Agent Mode Label	<EMPTY>	If provided, it will override the “Agent Mode” option in the drop-down list.
Zone	<EMPTY>	If provided, it will override the “Zone” option in the drop-down list.

Call ScreenPop Options field descriptions

Name	Default	Description
Pop on ANI? (Y/N)	Y	A field to enable screen pops for ANI. The options are: • Y: The screen pop for ANI is enabled. • N: The screen pop for ANI is disabled. **Note** Even if Pop on ANI is set to N, all other screen pop capabilities are still enabled.

Name	Default	Description
Pop on DNIS (Y/N)	N	A field to enable screen pops for DNIS. The options are: • Y: The screen pop for DNIS is enabled. • N: The screen pop for DNIS is disabled. **Note** Even if Pop on DNIS is set to N, all other screen pop capabilities are still enabled.
Pop on Transfer and Conference (Y/N/X/UUI Fields)	N	A field to enable screen pops on a transfer or a conference call. The options are: • Y: The screen pop is enabled. The transfer or the conference call recipient gets the last browsed salesforce object that the original users have browsed. • N: The screen pop is disabled. • X: This option will disable all pops on transfer and conference. Both Context Transfer and the base way of doing the screen pop is disabled. There is simply no screen pop for the recipient of a transfer or conference • "1", "2", "3", "4", "5" (in any combination): This enables Context Transfer, but when the Salesforce object id is added into the UUI, one or more UUI fields are eliminated at the same time. The configuration parameter defines which field or fields are to be eliminated. So, for example, if the configuration parameter is "1", then we will make sure that any data that was in UUI1 is eliminated. If the configuration parameter is "145", then we make sure that any data that was in UUI1, UUI4, and UUI5 is eliminated. If the configuration parameter is "12345" then all of the UUI data is cleaned out. Note that in this case, it will only do the deletion if necessary. If adding the Salesforce object id still fits under 96 bytes, then no replacement is done.

Name	Default	Description
		Note This feature uses Original Call Information, and if the transfer/conference is not done through the Connector, the expected pop will not happen.
Use E164 format for ANI search	N	A field to enable searching for ANI Number using E164 format. The options are: • Y: E164 ANI search is enabled. • N: E164 ANI search is disabled.
UUI1 to **UUI5**	<EMPTY>	Each of these components can be handled differently. In general, a given UUI element can be mapped to a given object and column in the Salesforce database (using the object.column form) or in all the columns using *. It can also be used to replace the ANI or DNIS values (by specifying ani or dnis), or it can be simply used for display (designated by a value starting with a). For more information, see the following table.
UUI Start and **UUI Stop**	; for start : for stop	A special character that denotes the end of the UUI as the Open CTI Adapter cares about it. The purpose of this character is to allow an arbitrary number of characters at the end of the UUI to be ignored for whatever reason.
UUI Separator	!	A special character that separates each of the UUI fields. It can occur up to four times in the UUI field and marks the separation between each field.
Suppress Screen Pop (C/A/W/D)		A filed to determine when a screen pop should be suppressed. The options are: • C: If an agent is on a call, the screen pop is suppressed. • A - If an agent is in AUX state, the screen pop is suppressed. • W - If an agent is in ACW state, the screen pop is suppressed.

Name	Default	Description
		• D – If the agent is on a direct call, the screen pop is suppressed. **Note:** We can have any combination of the above values in the configuration. If no value is provided, then this feature is disabled. When a screen pop is suppressed, the search action can still be performed, and the results will still be shown in the softphone. However, the active screen will not be changed to the results of the search.
End Pop VisualForce Page Name		A field to define the name of the VisualForce page that will be triggered at the end of a call.

Example UUI1 to UUI5

There are three delimiter characters for the UUI field: Start, Stop, and Separator. Start denotes where the Open CTI Adapter will start reading the UUI values. Anything prior to Start is completely ignored. Likewise, Stop denotes where the Open CTI Adapter will stop reading the UUI values. Anything after Stop is also completely ignored. Separator is used to break the fields apart. You only need enough Separator delimiters for the values present.

In general, the solution requires the use of Start and Stop. There is only one exception. If there is no Start delimiter present in the UUI, the entire UUI value will be used as the first element (UUI1).

All examples below assume that Start is ";", Stop is ":", and Separator is "!"

"00022"

";00022:"

";00022!!!!:"

These result in the same action: the value "00022" is placed into UUI1, and the other fields are left empty.

";!00001017:"

";!00001017!!!:"

These result in the value "00001017" placed into UUI2, and the other four fields left empty.

";!00001017!!CSR:"

";!00001017!!CSR!:"

These result in the value "00001017" placed into UUI2, the value "CSR" placed into UUI4, and the other three fields left empty.

Note that all three of the above examples show that any Separator delimiters ("!" in our examples) are not needed after the last field with a value, but can exist if so desired.

";A!B!C!D!E:"

In this example, the value "A" is placed into UUI1, the value "B" is placed into UUI2, the value "C" is placed into UUI3, the value "D" is placed into UUI4, and the value "E" is placed into UUI5. While this would rarely ever be used, it shows all the fields being filled with a simple value.

Reason Codes Enabled settings field descriptions

Name	Default	Description
Auxiliary Reason Enabled (Y/N)	N	A field to activate Auxiliary Reason codes. The option explicitly lists all the available Auxiliary Reason Codes. There are 20 slots for Auxiliary reason codes in the XML file. The format of the entry is ID=Label, where ID is the reason code number that is used by CM and Label is the string that is shown to the user. The ID values must not be consecutive. If the value is set to Y, you need to select the available Auxiliary Reason Code for selecting the state as Auxiliary. **Note:** If more than 20 slots are required, an alternative XML file can be provided that has all 99 slots.
Logout Reason Enabled (Y/N)	N	A field to activate logout reason codes. The option explicitly lists all the available logout reason codes. There are 9 slots for logout reason codes in the XML file. The format of the entry is ID=Label, where ID is the reason code number that is used by CM and Label is the string that is shown to the user. The ID values must not be consecutive. If the value is set to Y, you need to select the available Logout Reason Code while logging out of a station.
After Call Work Reason Enabled (Y/N)	N	A field to activate after call work reason codes. This option explicitly lists all after call work reason codes. There are 20 slots for after call work reason codes in the XML file. The format of the entry is just the reason code label, as there are no codes or IDs for after call work reason codes. **Note** If more than 20 slots are required, an alternative XML file can be provided that has more slots.

Keyboard shortcuts

The following are the list of keyboard shortcuts that can be used in the Connector for keyboard navigation. The default values can be changed. If you are defining a combination of keys for an action, the keys should be separated by the + sign.

Name	Keyboard shortcut default
Open Dialpad	Ctrl+Alt+m
Focus Make Call Textfield	Ctrl+Alt+8
Access Directory Popup	Ctrl+Alt+s
Redial	Ctrl+Alt+n
Open My Calls Today Report	Ctrl+Alt+t
Open Call History Popup	Ctrl+Alt+l
Drop Call	Ctrl+Alt+d
Answer Call	Ctrl+Alt+f
Hold Call	Ctrl+Alt+h
Retrieve Call	Ctrl+Alt+r
Consult Call	Ctrl+Alt+c
Blind Transfer	Ctrl+Alt+i
Blind Conference	Ctrl+Alt+o
Complete Transfer	Ctrl+Alt+j
Complete Conference	Ctrl+Alt+k
Drop Last Party	Ctrl+Alt+p
Focus on Subject Field	Ctrl+Alt+9
Focus on Name Dropdown	Ctrl+Alt+z
Focus on Related To Dropdown	Ctrl+Alt+x
Focus on Comments Text Area	Ctrl+Alt+v
Focus on Custom Field 1	Ctrl+Alt+4

Name	Keyboard shortcut default
Focus on Custom Field 2	Ctrl+Alt+5
Focus on Custom Field 3	Ctrl+Alt+6
Focus on Reason Dropdown	Ctrl+Alt+b
Focus First After Call Work Reason	Space
Finish After Call Work	Ctrl+Alt+w
Finish After Call Work And Set Available	Ctrl+Alt+q
Previous Cal	[
Next Call	]
Set Agent to Ready State	Ctrl+Alt+1
Set Agent to Auxiliary State	Ctrl+Alt+2
Set Agent to After Call Work State	Ctrl+Alt+3
Close Call History	Esc
Close Parties List	Esc
Cancel Ani Replacement	Esc
Close Ani Replacement	Enter

Server Configuration field descriptions

Name	Default	Description
Host FQDN/IP	<Default server / host name>	The FQDN of the Server Host Name setting points to the server (cluster) on which the IEC is running.
IEC Port	8483	The port number for inbound calls.
Use HTTPS? (Y/N)	Y	The Use HTTPS setting must be set to Y or N and indicates whether the connection used supports HTTPS (SSL/TLS) protocol.
Enable Debug Log Level? (Y/N)	N	If this field is set Y, the logs for CometD library (3rd party) will be set to DEBUG.

Important

Salesforce.com does not provide Call Center Definition validation capabilities and the user is able to enter any text in those fields. If the user enters a value different than the valid one for each configuration parameter, the softphone behavior is not guaranteed.

Login Mode Configuration field descriptions

Name	Default	Description
Agent Mode?(ACD-Only/Blended/Outreach-Only)	ACD-Only	A field to define the agent modes. **Note:** **For AAAC, only ACD-Only mode is supported**.

Call Center Agent Assignment

Salesforce users can be assigned to the Call Center that was defined by the Call Center definitions. This way the Salesforce users will be considered to be agents of the Call Center, and Salesforce will display the Salesforce UI for them.

Assigning Bulk Users

Procedure

1. Navigate to **<Username> > Setup > App Setup > Customize > Call Center > Call Centers**.
2. In **All Call Centers** list, select the desired **Call Center** name.
3. Scroll to the bottom of the page till **Call Center Users** and click **Manage Call Center Users**.
4. Use the following screen to assign users to the call center.

Note

Only those Salesforce users will be listed on these configuration pages that are not currently assigned to any Call Centers. If you want to change an existing Call Center agent assignment check the **Individual Assignment** below.

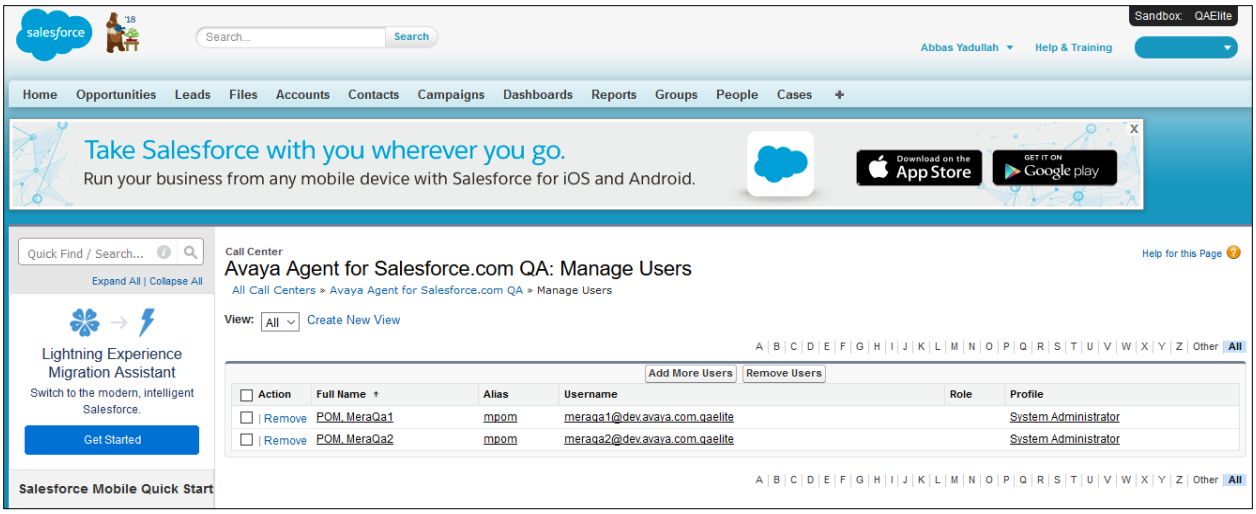

Figure 14: Manage users

Assigning Individual Users

1. Navigate to **<Username> > Setup > Administration Setup > Manage Users > Users**.
2. Select the user to edit from the list and click **Edit**.
3. Use the **Call Center** text field to set the desired Call Center for the user.

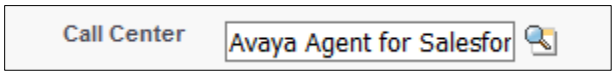

Figure 15: Call center selection

Configuring Softphone Layout

Softphone Layout is used to determine how information is displayed within a softphone.

Procedure

6. Navigate to **App Setup > Call Center > Softphone Layouts**.
   The system displays a list of all Softphone Layouts. Initially there will be only a single one. Use the **New** button to create new softphone layouts. If there are multiple softphone layouts, use the **Softphone Layout Assignment** button to assign which groups of users use which softphone layout.
7. To see the settings for a softphone layout, click the **Edit** link.
   The system displays the settings for the selected softphone layout.

Softphone Layout Settings

Name	Description
Select Call Type	The **Select Call Type** drop-down list in the black bar determines the type of the call. The settings vary based on the call type selected. The options are: • **Inbound**: The options are the most varied and are presented in to main sections. The sections are **Softphone Layout** and **Screen Pop Settings**. • **Outbound** and **Internal** calls: There is no **Screen Pop Settings** for Outbound and Internal calls.
Softphone Layout	The **Softphone Layout** section determines the options shown automatically in the softphone display. The options are: • **Display these call-related fields**: This option determines the call data elements that are always shown. By default, the system shows **Caller ID** and **Dialed Number**. There are two other options available: **Queue** and **Segment**. The Open CTI Adapter does not make use of **Segment**, but **Queue** can be selected and used if desired. • **Display these Salesforce.com objects**: This option determines which objects are searched to find a match. Only the mentioned collection of objects is searched when searching with ANI. So, for example, if the set of Objects defined are just **Contact** and **Lead**, then **Account** cannot be matched. The objects in the list may be custom objects, too. In the case of UUI, any objects can be searched, as the UUI setting will define the object required. • Underneath this is the darker gray box where you can define the elements of a matched object. This list can be expanded as desired but having too many elements can affect readability in the softphone.
Screen Pop Settings	The Screen Pop Settings determines the search results. The options are: **Screen Pop Settings**: • **Screen pops open with**: This can be set for either **Existing browser window** or **New browser**

Name	Description

	window or tab. The default option is **Existing browser window**. This setting only applies to Classic Standard. It is ignored in Classic Console and Lightning modes.
	• **No matching records**: This option determines what happens when a screen pop attempt fails to find any records. The possible values are:
	  ○ **Don't pop any screen**
	  ○ **Pop to new**
	  ○ **Pop to Visualforce page**.
	The default and recommended setting for simple screen pops is **Don't pop any screen**.
	• **Single-matching record**: This option determines what happens if a unique match is found. The possible values are:
	  ○ **Don't pop any screen**
	  ○ **Pop detail page**
	  ○ **Pop to Visualforce page**
	The default and recommended setting for simple screen pops is **Pop detail page**.
	• **Multiple-matching records**: This option determines what happens if multiple records are returned by the search attempt. The possible values are:
	  ○ **Don't pop any screen**
	  ○ **Pop to search page**
	  ○ **Pop to Visualforce page**.
	The default setting is **Don't pop any screen**, but the recommended setting for simple screen pops is **Pop to search page**.
	• In all three scenarios, for advanced search capabilities, select **Pop to Visualforce page** for all three and have those all pop to the same Visualforce page. This provides the maximum flexibility and ease in providing screen pops when showing a simple detail page.
	• Also, when using both UUI and Visualforce page screen pops, it is not necessary to use the parsing capabilities set up in the call center definition.

Name	Description
	Instead UUI is passed through the CTI Adapter and provided to the Visualforce page. This allows the Visualforce page to parse the UUI data, providing more flexibility than the standard five elements.

Visualforce Page

When the screen pop is configured to pop to a Visualforce page, the following elements are sent to the Visualforce page as query parameters:

- ANI (Caller Id)
- DNIS (Dialed Number)
- UUI (User-to-user Information)
- UCID (Universal Call Id)
- Queue (Skill Extension Number)

These values are always passed but could be empty. (For example, if a call arrives with no UUI value, then the UUI parameter will be left empty.) The Visualforce page will have to pull the values from the query parameters to get them and them operate on them.

When passing UUI into the Visualforce page, the entire unparsed UUI is provided. As a result, when passing the UUI, the “five fields” limitation and the requirements for the separator characters does not have to be observed, as the Connector is not trying to parse the UUI; it is simply passing it on to the Visualforce page. As such, it is entirely possible to encode the UUI differently in this case. It is also possible to still pass in fields the Connector can parse (for example providing an alternative ANI value or one or more display values), but still pass additional information the Visualforce page will read that is outside (for example, after the UUI Stop Character) what the Connector will parse.

Configuring Salesforce Directory

This section is used to fill out the directory provided in the CTI Adapter.

Procedure

8. Navigate to **App Settings > Call Center > Directory Numbers**.
   Each entry is simplistic with just a simple name and phone number. The user interface provides only the ability to provide one entry at a time. To perform a bulk addition, other Salesforce tools must be used. When selected, the current list of directory numbers is shown.
9. To add new entries, click **New**.

When adding entries to the Directory, keep these in mind:

- When adding the directory entries, all that is required is the Name and Phone entry. If the **Call Center** field is left blank, the directory entry will apply to all call center definitions. If you wish to restrict a directory entry to a single call center definition, then it can be selected using the search icon.
- The **Directory Numbers** section will only allow directory entries to be added one at a time. To do a bulk addition, you must use other Salesforce tools.

Configuring the Console

When using the Salesforce Classic Console, it is critically important to configure the "Whitelist Domains" setting in the Salesforce administration. If this is not done, then none of the interactions with the Classic Console will work.

Procedure

1. Navigate and click the **Setup** option.
   The system displays the setup page.
2. On the left menu, navigate to **App Setup > Create > Apps**.
   The system displays the list of applications.

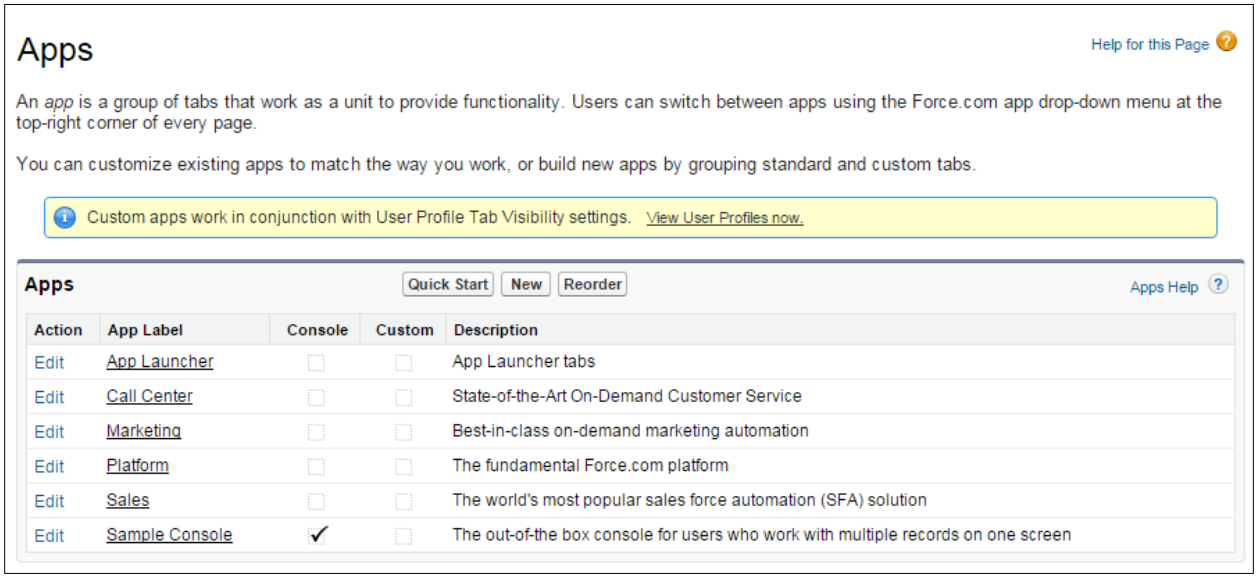

Figure 16: The application list page

3. Click the **Edit** link next to the application you are using.
4. When editing the Console application, go to the **Whitelist Domains** section as shown below:

Figure 17: The Whitelist Domains option

5. Type the name and port of the Open CTI Server in the following format:
   `<server address>:8443`

6. You can type multiple entries separated by commas.

Note

It is important that the same FQDN used in the other configuration entries is used. It is critical that the port be included. If the port is not included, then the entry will have no effect.

Setting up Avaya CRM Connector in Lightning Experience App Manager

About this task

The following steps must be performed in Salesforce.com setup to enable the Utility Bar which holds the Avaya CRM Connector softphone button and the panel which displays the Avaya CRM Connector in the Lightning Experience view.

Procedure

1. Navigate to **<Username> > Setup**.
2. In the *Quick Find* field, search for **App Manager** as shown in the following screenshot:

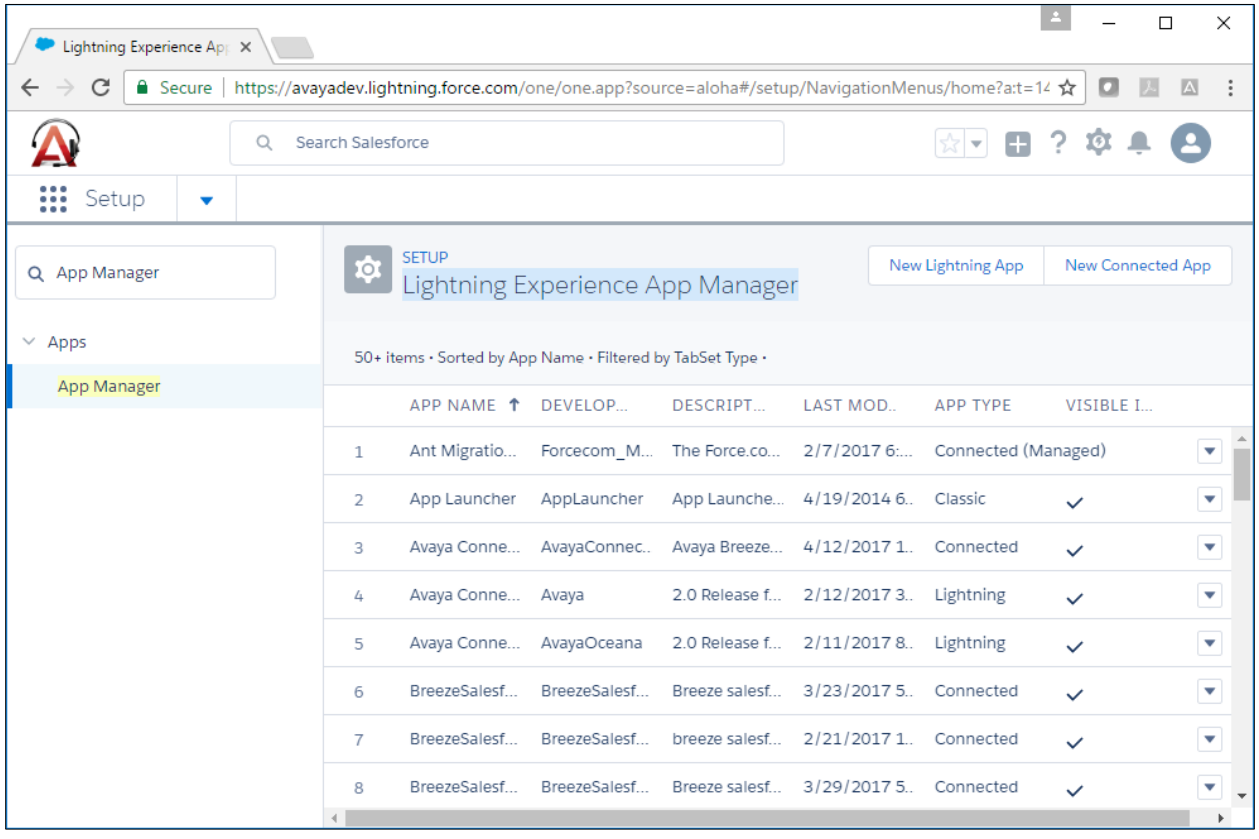

Figure 18: Quick Find – App Manager

3. While either creating or editing a **Lighting App**, navigate to **Utility Bar** > **Add** (**Utility Bar Items**).

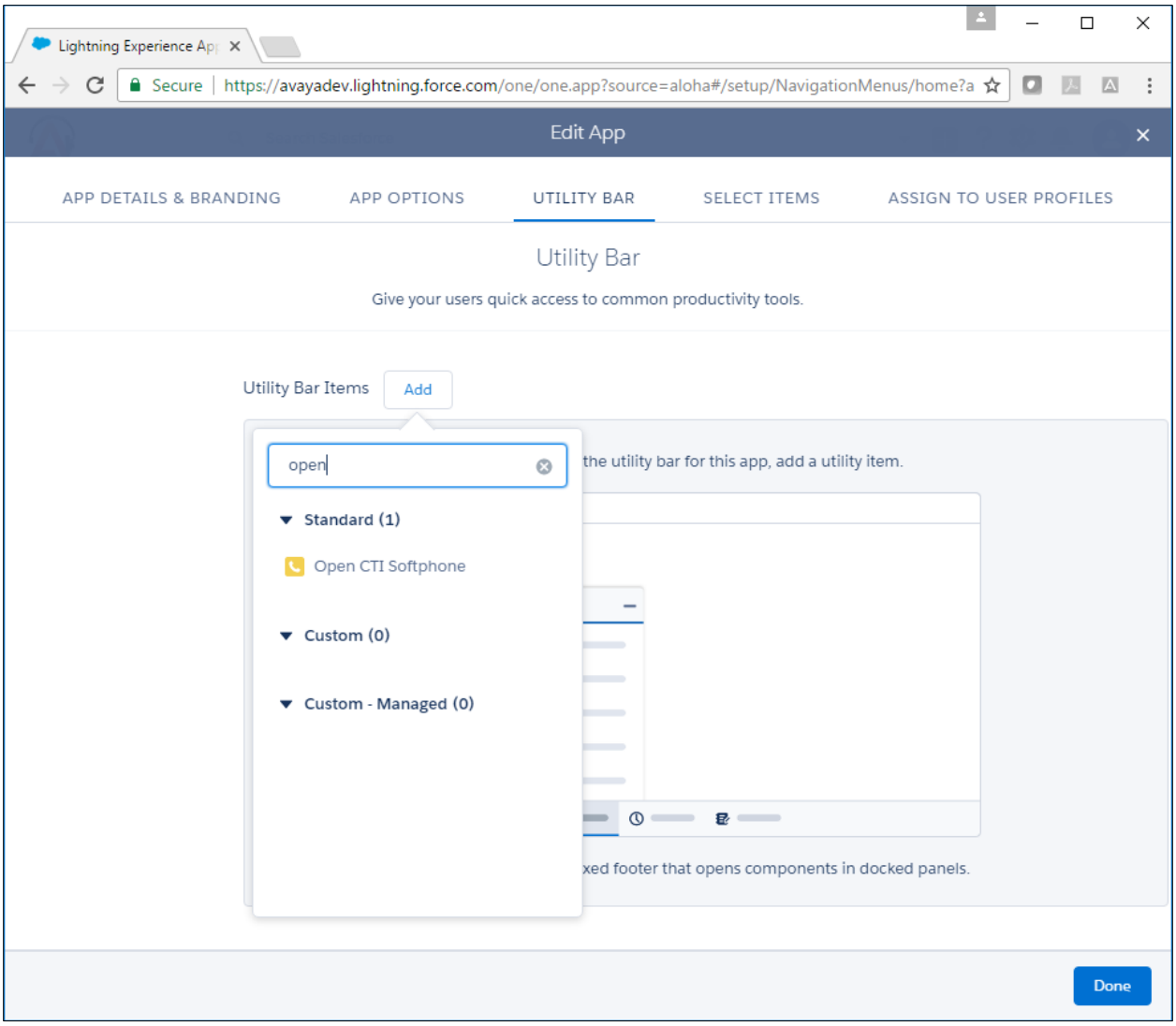

Figure 19: Add Utility Bar

4. Search for **Open CTI Softphone** and click the entry to add it in the **Utility Bar** pane. The system displays the Phone tab configuration properties.

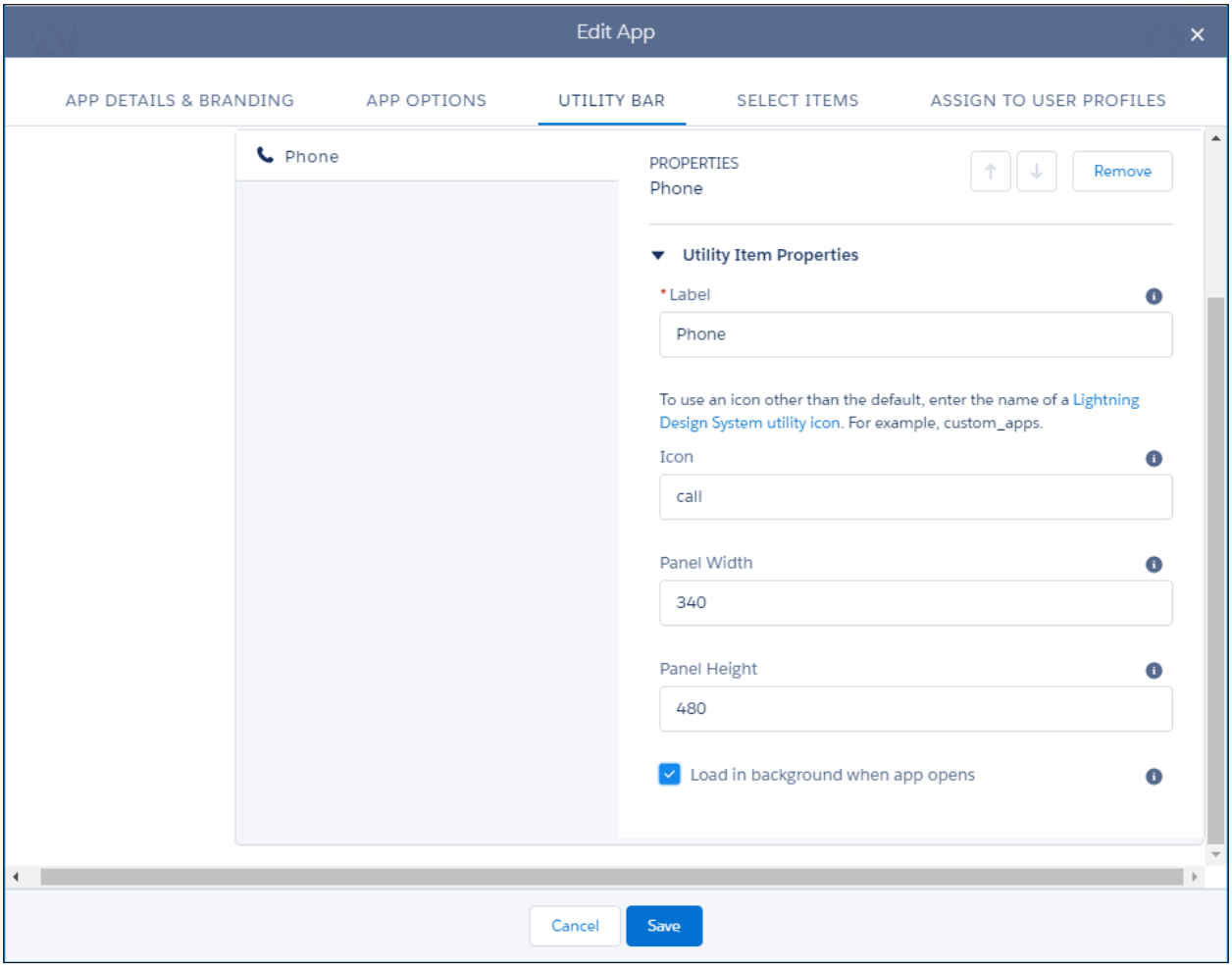

Figure 20: Utility Bar

5. In the **Panel Width** field, set the value to *260*.
6. In the **Panel Height** field, set the value to *600.*
7. If editing a Lightning App, click **Save the Changes**. OR

   If creating a new Lightning App, follow the wizard and complete the details.

   The Utility Bar and the Avaya CRM Connector button are displayed after a new user logs in the Salesforce application as shown in the following screenshot:

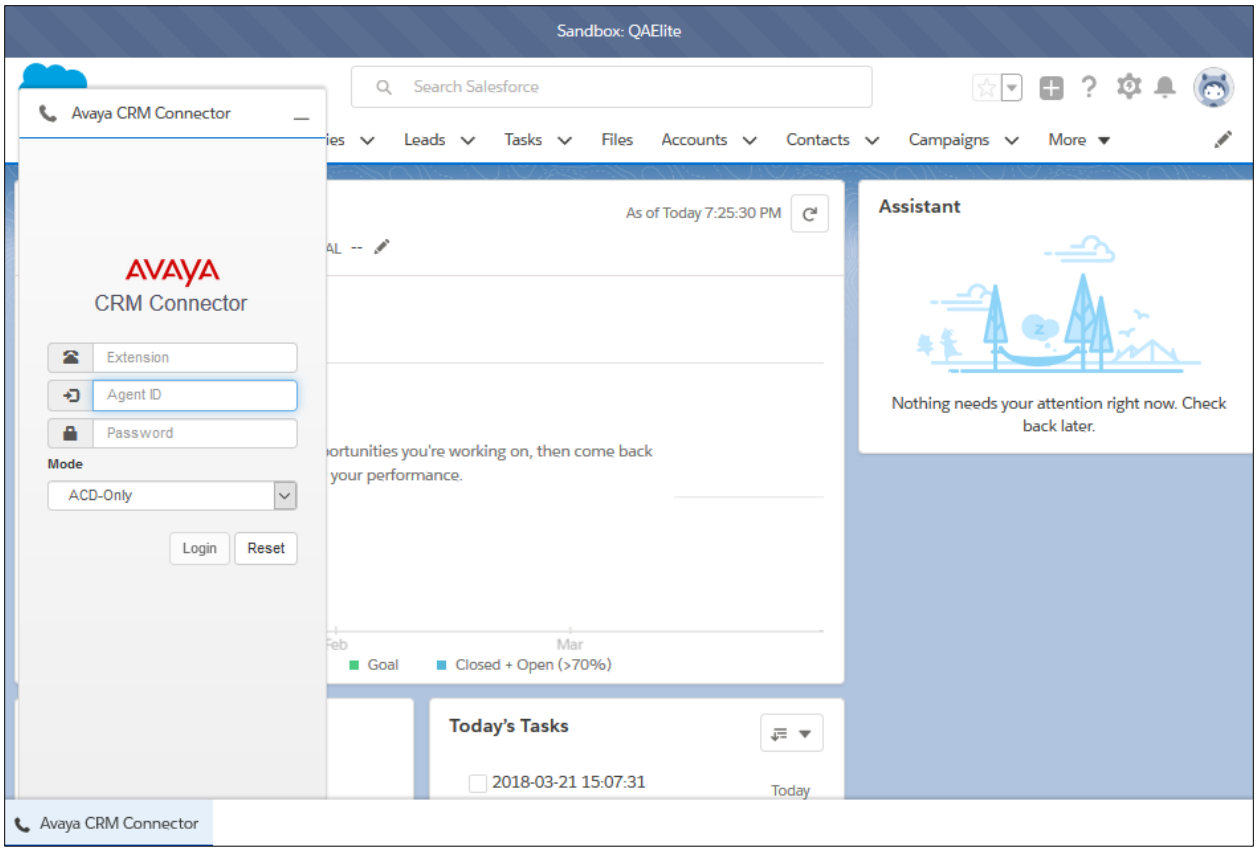

Figure 21: Avaya CRM Connector softphone panel and button

Query parameters on Visualforce page

Salesforce Lightning Standard and Salesforce Classic console modes does not allow the query parameters to be displayed in the URL. You must perform the following steps to test the query parameters that are sent to a VisualForce page.

Visualforce page configured in Softphone Layout for Inbound Screen Pop

Note:

To see the Visualforce page in the application, the name of the Visualforce page must be the same as configured in the **Screen Pop Settings** section of the **Softphone Layout**.

Procedure

1. Navigate to **Setup** > **Build** > **Develop** > **Apex Classes**.
2. Click **New** to create an Apex Class.
3. Add the following lines here:

```
global class VisualForceQueryParamsController {
        public String ani {get; private set;}
```

```
        public String dnis {get; private set;}
        public String uui {get; private set;}
        public String ucid {get; private set;}
        public String queue {get; private set;}
        public VisualForceQueryParamsController() {
                ani =
ApexPages.currentPage().getParameters().get('ANI');
                dnis =
ApexPages.currentPage().getParameters().get('DNIS');
                uui =
ApexPages.currentPage().getParameters().get('UUI');
                ucid =
ApexPages.currentPage().getParameters().get('UCID');
                queue =
ApexPages.currentPage().getParameters().get('Queue');


      }
}
```

4. Navigate to **Setup** > **Build** > **Develop** > **Visualforce pages**.
5. Click **New** to create the VisualForce page.
6. Add the following lines here:

```
<apex:page controller="VisualForceQueryParamsController">
   <apex:pageBlock title="Inbound Call Information">
       <style type="text/css">
       .zui-table {
       border: solid 1px #DDEEEE;
       border-collapse: collapse;
       border-spacing: 0;
       font: normal 13px Arial, sans-serif;
           }
.zui-table thead th {
    background-color: #DDEFEF;
    border: solid 1px #DDEEEE;
    color: #336B6B;
    padding: 10px;
```

```
    text-align: left;
    text-shadow: 1px 1px 1px #fff;
}
.zui-table tbody td {
   border: solid 1px #DDEEEE;
   color: #333;
   padding: 10px;
   text-shadow: 1px 1px 1px #fff;
}
</style>
      <table class="zui-table">
           <tr style="background-color: #DDEFEF;">
               <td>Parameter Name</td>
               <td>Parameter Value</td>
           </tr>
            <tr>
               <td>ANI</td>
               <td>{!ani }</td>
            </tr>
            <tr>
                <td>DNIS</td>
                <td>{!dnis}</td>
            </tr>
            <tr>
                <td>UUI</td>
                <td>{!uui }</td>
            </tr>
            <tr>
                <td>UCID</td>
                <td>{!ucid }</td>
            </tr>
            <tr>
                <td>Queue</td>
                <td>{!queue }</td>
```

```
            </tr>
          </table>


    </apex:pageBlock>
</apex:page>
```

Visualforce page configured in Call Center definition for Outbound Screen Pop

Note:

To see the Visualforce page in the application, the name of the Visualforce page must be the same as the Outbound Screen POP Visual Force Page URL field in the Call Center Definition.

Procedure

7. Navigate to **Setup** > **Build** > **Develop** > **Apex Classes**.
8. Click **New** to create an Apex Class.
9. Add the following lines here:

```
global class VisualForcePOMQueryParamsController {
        public String campaign {get; private set;}
        public String dialedNumber{get; private set;}
        public String firstName {get; private set;}
        public String lastName {get; private set;}
        public String phone {get; private set;}
        public String timeZone {get; private set;}
        public String email {get; private set;}
        public String zipCode {get; private set;}
        public VisualForcePOMQueryParamsController() {
                campaign =
ApexPages.currentPage().getParameters().get('CampaignName');
                dialedNumber =
ApexPages.currentPage().getParameters().get('Dialed');
                firstName =
ApexPages.currentPage().getParameters().get('First Name');
                lastName =
ApexPages.currentPage().getParameters().get('Last Name');
```

```
                phone =
ApexPages.currentPage().getParameters().get('Phone 1');
                timeZone =
ApexPages.currentPage().getParameters().get('Time Zone');
                email =
ApexPages.currentPage().getParameters().get('E-Mail');
                zipCode =
ApexPages.currentPage().getParameters().get('Zipcode
Predefined');
      }
}
```

10. Navigate to **Setup** > **Build** > **Develop** > **Visualforce pages**.
11. Click **New** to create the VisualForce page.
12. Add the following lines here:

```
 <apex:page controller="VisualForcePOMQueryParamsController">
   <apex:pageBlock title="Outbound Call Information">
       <style type="text/css">
       .zui-table {
       border: solid 1px #DDEEEE;
       border-collapse: collapse;
       border-spacing: 0;
       font: normal 13px Arial, sans-serif;
           }
.zui-table thead th {
    background-color: #DDEFEF;
    border: solid 1px #DDEEEE;
    color: #336B6B;
    padding: 10px;
    text-align: left;
    text-shadow: 1px 1px 1px #fff;
}
.zui-table tbody td {
   border: solid 1px #DDEEEE;
   color: #333;
   padding: 10px;
```

```
   text-shadow: 1px 1px 1px #fff;
}
</style>
      <table class="zui-table">
           <tr style="background-color: #DDEFEF;">
               <td>Parameter Name</td>
               <td>Parameter Value</td>
           </tr>
            <tr>
               <td>Campaign Name</td>
               <td>{!campaign }</td>
            </tr>
            <tr>
                <td>Customer</td>
                <td>{!dialedNumber}</td>
            </tr>
            <tr>
                <td>First Name</td>
                <td>{!firstName }</td>
            </tr>
            <tr>
                <td>Last Name</td>
                <td>{!lastName }</td>
            </tr>
            <tr>
                <td>Phone</td>
                <td>{!phone }</td>
            </tr>
            <tr>
                <td>Time Zone</td>
                <td>{!timeZone }</td>
             </tr>
             <tr>
               <td>E-mail</td>
```

```
                <td>{!email }</td>
            </tr>
            <tr>
                <td>Zip Code</td>
                <td>{!zipCode }</td>
             </tr>
        </table>

    </apex:pageBlock>
</apex:page>
```

Chapter 4: System maintenance and monitoring

WebLM status

To check the WebLM status, navigate to **Licensing > WebLM Server Address** and check that the WebLM IP address is not set to 127.0.0.1 if the local WebLM is disabled as shown in the following figure:

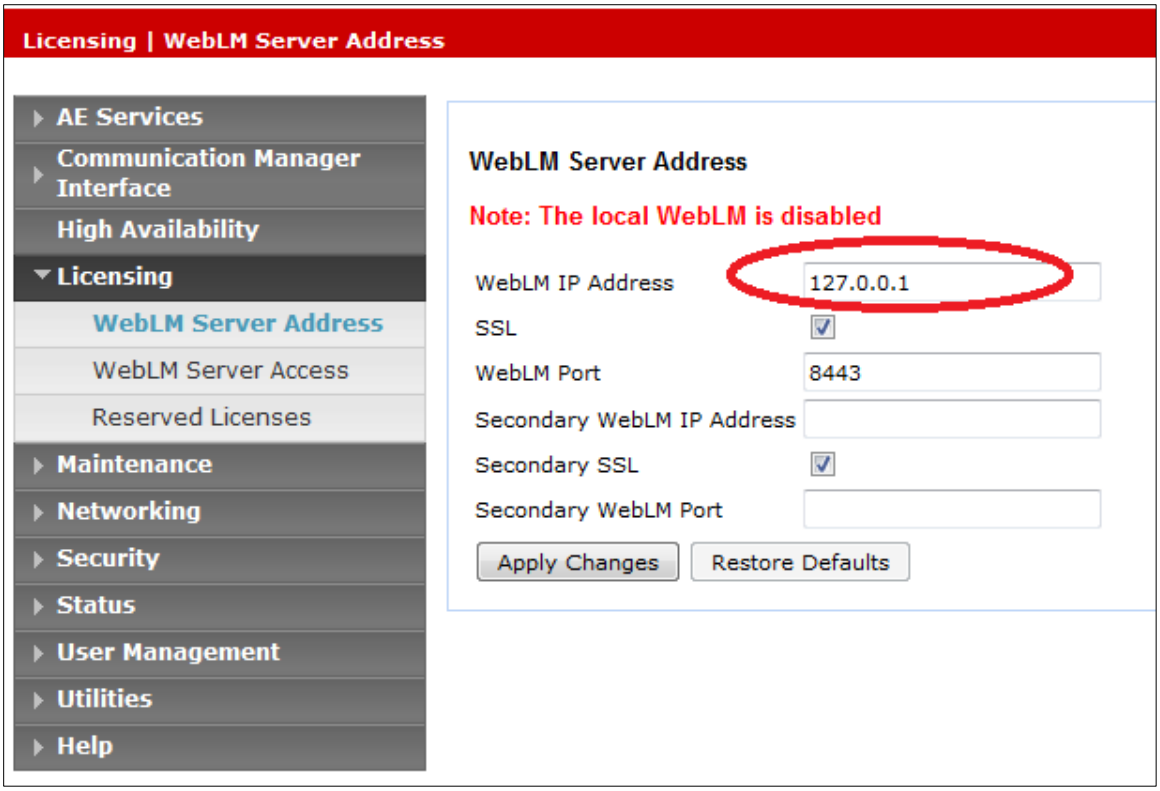

Figure 22: WebLM Server Address

Note

The WebLM IP Address should be administered to a remote location if the local WebLM is disabled.

SIP Endpoints

If a SIP endpoint is registered using an E.164 address, then this will lead to privilege violation exceptions on AES. The SIP endpoints should register using its extension number as configured on Communication Manager.

Appendix A: High Availability and Failover

High Availability and Fault Tolerance

The Avaya CRM Connector 2.2 for AACC application is implemented as a single node. The scalability and high availability are achieved by deploying multiple nodes deployment without session synchronization among the cluster nodes.

A load balancer must be placed in front of the servers and must be used for distributing the load, monitoring down servers, and redirecting requests to online servers. Thus, you must have at least one additional redundant server in production to allow the solution to continue supporting the load in case one of the servers becomes unavailable.

In the situation where a server faces an issue (for example: network outage), the agent may lose control over the current interactions. When the control is lost the agent must complete the current interactions and start over the login process. In this scenario the load balancer forwards the new login request to another online instance.

To provide geo redundancy, the application needs to be deployed into at least two different sites, this prevents the entire contact center operation to stop due a datacenter outage (for example: power outage, network outage, or communication link outage.

The scalability, the fault tolerance, and disaster recovery will depend on:

- The deployment schema chosen by the customer in accordance with Avaya.
- Load balancers and servers to be provided by the customer.

The scalability, the fault tolerance, and disaster recovery will depend on:

- The deployment schema chosen by the customer in accordance with Avaya.
- CM server's high availability and fault tolerance
- Number of Avaya CRM Connector servers
- Load balancers and servers to be provided

Capacity

A single application instance supports up to 1000 simultaneous logged in agents. Adding new instances increase the capacity in:

```
 ((number of servers -1) * number of agents per server) * (number of
data centers)
```

```
Where number of agents per server = 1000
```

A load balancer must be placed in front of the servers and must be used for distributing the load. Thus, you must have at least one additional redundant server in production to allow the solution to continue supporting the load in case one of the servers becomes unavailable.

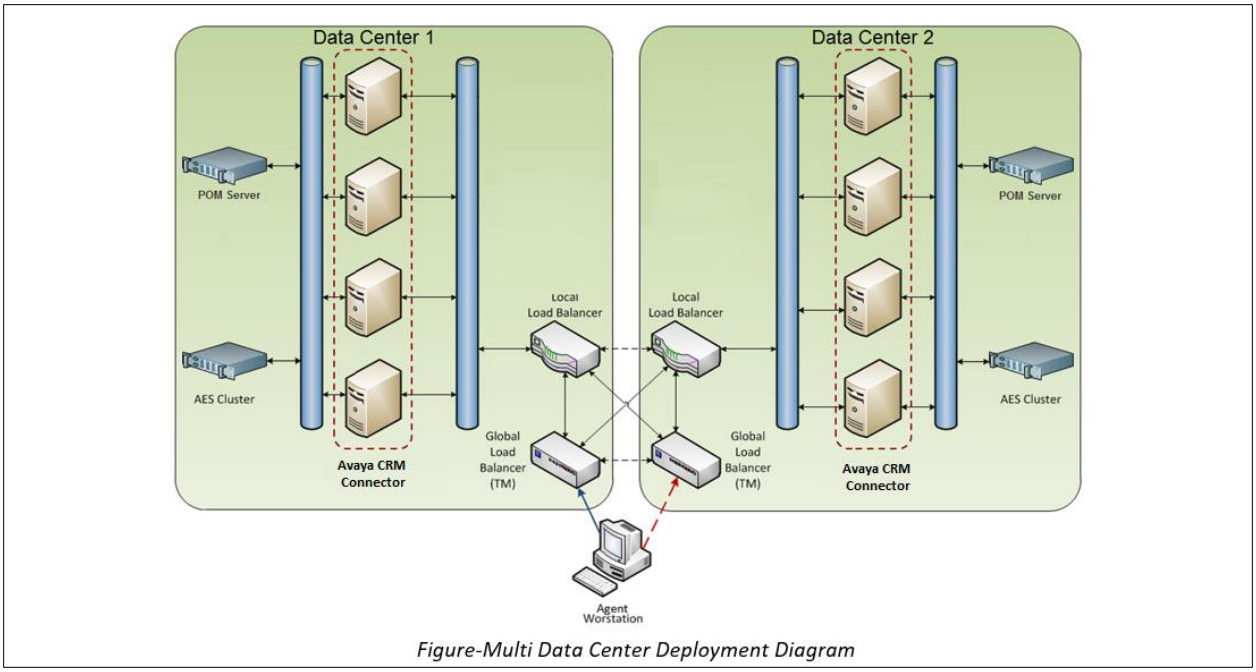

Figure 23: Multi Data Center Deployment Diagram

The Load Balancer depicted in the screenshot is an example of one that has intra-data center and inter-data center balancing features split in to two appliances. Note that, some load balancers perform both features in a single box.

Recommended setting for Load Balancer

- ***proxy_set_header Accept-Encoding ""*** header should be set. The header needs to be passed from the client to backend and the other way around. Load Balancer should not remove the headers and allow all the encodings. Concerning CRM Connector, Load balancer should be transparent.
- ***SameSite=None*** configuration is required and should be done on Load Balancer side
- Session affinity should be set - input for routing decision usually source IP
- Connection persistence timeout should be set (recommended value: 8 hours)
- Session persistence should be enabled.
- Adjust idle timeout – especially for Outbound only agents (recommended value: 1 hour – but it depends on the Agent/CC profile (inbound, outbound, call volume)
- CRM Connector IPs in the whitelist of Reverse Proxy must be added. It is required for the Load Balancer to not falsely mark traffic as DOS attack.

Appendix B: Call Logging

Main scenarios for call logging

The following are the list of main scenarios for call logging in Avaya CRM Connector:

- The call logs are stored on server-side in the session object that refers to a call.
- The Server Start Time is recorded on the Established Event and the End Time on the Release Event.
- The Server Start Time and End Time are based on server's UTC Timestamps.
- The Server Start Time is used to calculate the call log duration.
- The Call Log Duration is calculated from End Date to Server Start Date.
- The Client Start Time is pushed to the server on the Answered Event.
- The Client Start Time is used to reflect the talk time.
- On the client application, the UTC timestamps is displayed in the client's/agent's time zone.

Note:

The timestamps are shared from the server to the client and vice versa in a time format containing values up to milliseconds.

Configuration

The Call Center Definition key value pairs are sent to the server and is stored with the agent's session as soon as the agent logs in to the softphone application.

Alternate scenarios

The following are few alternate scenarios for call logs:

- When a call is auto answered, either through CM or IEC (server-side) based on the call center definition, the call log and the call timer will work following the same rules, which is as if the call was manually answer.
- If the **Save Call Log** is set to **N**, then no Information is pushed to the client. Also, the client will not make requests to update the call logs on the server.
- If **Save Incomplete Call** is set to **Y**, then failed and abandoned calls will generate call logs having the Start Time and End Time set to the time the failed event occurred. This will generate a 0-duration call log.

JournalD

JournalD or systemd-journald is a system service that collects and stores logging data. JournalD creates and maintains structured, indexed journals based on the logging information it receives from the various sources.

Avaya CRM Connector leverages JournalD log driver for Docker as the default configuration which enables log records to get stored into JournalD.

JournalD can be configured using default journald.configuration file in /etc/systemd/.To save log files on disk (/var/log/journal directory) is required to configure `Storage` parameter:

```
Storage=persistent
```

To limit log file size can be used `SystemMaxFileSize` and `MaxRetentionSec` parameters.

Any changes to the configuration file require to restart journal service using command:-

```
systemctl restart systemd-journald
```

Below is an example how can be configured JournalD configuration file:

```
[Journal]
Storage=persistent
Compress=yes
#Seal=yes
#SplitMode=uid
#SyncIntervalSec=5m
#RateLimitInterval=30s
#RateLimitBurst=1000
#SystemMaxUse=
#SystemKeepFree=
SystemMaxFileSize=10M
#RuntimeMaxUse=
#RuntimeKeepFree=
#RuntimeMaxFileSize=
MaxRetentionSec=2m
#MaxFileSec=1month
#ForwardToSyslog=yes
#ForwardToKMsg=no
#ForwardToConsole=no
#ForwardToWall=yes
#TTYPath=/dev/console
#MaxLevelStore=debug
#MaxLevelSyslog=debug
#MaxLevelKMsg=notice
#MaxLevelConsole=info
#MaxLevelWall=emerg
#LineMax=48K
```

For more details how can configure JournalD view the following link:

https://www.freedesktop.org/software/systemd/man/journald.conf.html

Journalctl

Journalctl is tool to perform queries on Journal log records. To query logs, use journalctl features, for example: -

```
journalctl -b CONTAINER_NAME=images_aa4salesforce-app_1
```

To know more about the Journalctl tool, view the following links:

- https://docs.docker.com/config/containers/logging/journald/#retrieve-log-messages-with-journalctl
- https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux_atomic_host/7/html-single/getting_started_with_containers/index

Docker Logs

You can use Docker log commands to see container logs. To query logs, use Docker log features, for example: -

```
docker logs images_aa4salesforce-app_1
```

Appendix C: Troubleshooting

Collecting logs for troubleshooting

- Currently there are two scripts which you can use for troubleshooting:
  - **save-logs.sh:** To save the logs in the integration.log file. Also, saves the logs in different files per each component, such as **aacc3pcc.log** and **iec.log**.
  - **view-logs.sh:** To view the logs on the console.
- You can also run specific Docker commands in case you want some specific information. For instance:-

  `docker logs --since 30m <container ID>`
  This command displays the last 30 minutes of messages for a given container.

Docker services commands

Command	Description
./stop.sh	Use this command to stop a docker service.
./start.sh	Use this command to start a docker service.
docker-compose restart	Use this command to restart a service. You must restart the service in case there are any changes made in the configuration files.
docker –ps	Use this command to view the status of all running services.

Viewing the Docker services individual component logs

To view the individual component logs, you need to navigate to the following locations:

- **docker-compose logs > integration.log**
- **docker-compose logs aacc3pcc-app > aacc3pcc.log**
- **docker-compose logs iec-app > iec.log**

Unable to use the application or open Visualforce page in a Lightning mode

If you have a problem using the application in a lightning mode or if you cannot open the Visualforce pages, then you must modify the following file:

integration/config/git-server/default-config/aa4salesforce-prod.yml

```
cors:
allowed-origins: "force.com, salesforce.com"
```

Also, you must add their domain in the list.

```
allowed-origins: "force.com, salesforce.com, domain"
```

Appendix D: Resources and Glossary

Resources

The following table lists the documents related to Avaya CRM Connectors. Download the documents from the Avaya Support website at http://support.avaya.com.

Title	Use this resource to:	Audience
Avaya Aura® Communication Manager Overview and Specification guide	Understand the key features, the functionalities, and the system requirements of the Avaya Aura® Communication Manager application and the components.	IT Management and support personnel
Using Avaya CRM Connector 2.2 for AACC	Understand and use the Avaya CRM Connector 2.2 softphone application	IT Management and support personnel
Avaya Aura® Contact Center Overview and Specification	Understand the product features, specifications, licensing, and interoperability with other supported products.	IT Management and support personnel
Avaya Aura® Contact Center Server Administration	Understand the information and procedures for day-today maintenance of all servers in the Contact Center suite, including server maintenance tasks, administrative tasks, managing data, configuring data routing, performing archives, and backing up data. It also describes the optional configuration procedures for server configuration.	IT Management and support personnel
Avaya Aura® Contact Center Client Administration	This document contains information and procedures to configure the users and user access, skillsets, server management, and configuration data in the Contact Center database.	IT Management and support personnel

Glossary

AACC

Avaya Aura® Contact Center. Avaya Aura® Contact Center (AACC) is a call-center application that can be used to efficiently serve customers. It is a context-sensitive, collaborative voice customer experience management solution that allows businesses to manage all types of customer interactions through a unified application.

CTI

Computer Telephony Integration. A technology that integrates the telephone with the computer for managing telephone calls.

Softphone

Software used to make calls over the Internet by using a computer. A softphone functions like a traditional telephone, but without the dedicated hardware, such as telephone cables and phone sets.

Salesforce

An application that manages customer relationships, integrates with other systems, and allows user to build own applications.

ANI

Automatic Number Identification. A display of the calling number so that agents can access information about the caller.