A MITEL
PRODUCT
GUIDE

Unify OpenScape Contact Center Enterprise

WebRTC V11 R1

Description
07/2023

A31003-S22B1-T100-04-7618

Mitel

Notices

The information contained in this document is believed to be accurate in all respects but is not warranted by Mitel Europe Limited. The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

Trademarks

The trademarks, service marks, logos, and graphics (collectively “Trademarks”) appearing on Mitel’s Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC) or its subsidiaries (collectively “Mitel), Unify Software and Solutions GmbH & Co. KG or its affiliates (collectively “Unify”) or others. Use of the Trademarks is prohibited without the express consent from Mitel and/or Unify. Please contact our legal department at iplegal@mitel.com for additional information. For a list of the worldwide Mitel and Unify registered trademarks, please refer to the website: http://www.mitel.com/trademarks.

Table of Contents

History of Change

Date	Issue	Summary
07/2022	1	Create V11R1 version
08/2022	2	Update regarding the support for Windows Server 2022
04/2023	3	Add reference to HAProxy White Paper
07/2023	4	Updated chapters: • 4.2 – Firewall, • 5.3 – Application Server, • 5.4 – OpenScape Contact Media Service, • 5.6.1 - Commercial Load Balancer

1. Introduction

1.1 Agent Portal Web – Integrated Phone

The Integrated Phone is an embedded WebRTC client in the OSCC Agent Portal Web application. With the WebRTC client you can handle voice calls.

Specifically:

- Receive voice contacts in Agent Portal Web without needing a physical phone or a soft phone client.
- Configure which devices will be used for audio.
- Start voice contacts via the Integrated Phone in Agent Portal Web.
- Dial DTMF during voice contacts to interact with an answering machine or IVR.

The OpenScape Contact Media Service server has been enhanced to work also as a WebRTC server. The WebRTC server function is provided by an integrated Media Server, which works as a gateway between WebRTC and SIP/RTP to the communication platform. Each WebRTC subscriber has a corresponding SIP subscriber in the communication platform. OSCC monitors the communication platform via the CSTA protocol to control the call routing.

Multiple OpenScape Contact Media Service servers can be deployed for load balancing and scaling. Each OpenScape Contact Media Service server can support up to 300 registered WebRTC clients.

The WebRTC server component is installed as part of the OpenScape Contact Media Service server.

The WebRTC client component is installed as part of the Agent Portal Web on the Application Server.

1.2 Agent Portal Web – Video and Screen Sharing

This feature allows you to interact with customers using Video and Screen Sharing. The feature can be used in a voice contact between two agents or between an agent and an external caller. To use the feature "Video/Screen Share", the feature "Integrated Phone" must be enabled for any agent involved as a prerequisite.

In case of an external caller involved, the "Click-to-Contact" component needs to be embedded in a web page on the corporate web server, through which the caller can initiate a call to the contact center without using a physical phone or a soft phone client.

The "Click-to-Contact" component offers the following functionality:

- Select which devices for audio and video are used by the caller;
- Enter the phone number of the caller (used as source number for caller identification in the OSCC system).
- Select the service the caller would like to use (optional, associated with multiple OSCC queues).
- Initiate the call to the contact center.

The camera and headset must be properly configured and the supported browsers must have permission to access them. Once the call has been routed and established with an agent, the agent can start a video or screen share with the caller through his Agent Portal Web, as well as the caller can start a video or screen share with the agent through the "Click-to-Contact" component on the web page.

The feature "Video/Screen Share", like "Integrated Phone", is based on WebRTC standard, see chapter 1.1. Agent Portal Web - Integrated Phone.

Note: This feature is not available for OpenScape 4000.

2. Description

The Integrated Phone solution uses the following components:

- Agent Portal Web – The Integrated Phone runs as a WebRTC client in a web browser
- Application Server – The Application Server hosts the Agent Portal Web application and is responsible for:
    - Intermediating the registration process of the Integrated Phone with the OpenScape Contact Media Service.
    - Load balancing between the multiple OpenScape Contact Media Service nodes, for scalability or in case of redundancy scenarios.
    - Intermediating the authentication process of the Integrated Phone with the TURN Server (see OpenScape Contact Media Service below).
- OpenScape Contact Media Service – The OpenScape Contact Media Service server is responsible for:
    - Acting as the WebRTC server, interworking directly with the WebRTC client after registration of the Integrated Phone.
    - Hosting the TURN server, which must be used when the Integrated Phone is running behind a NAT.
    - Acting as a SIP User Agent to the communication platform.

  Note: OpenScape Contact Media Service acts as a gateway between WebRTC and SIP.
- Load Balancer – Commercial (like F5 or Netscaler) or HAProxy (suggested) – The Load Balancer can be used for:
    - Load balancing between the multiple Application Servers.
    - Protecting the external access to the OpenScape Contact Media Service configuration web page.
- OSCC Server – Monitors the extensions of the Integrated Phones as well as any other extension used by agents.
- The Integrated Phone is seen by the communication platform as a regular phone, using variations of CSTA protocols depending on the communication platform (OpenScape Voice / OpenScape 4000).

The Video and Screen Share solution uses in addition the following component (not applicable in agent-to-agent scenarios without external caller):

- Corporate Web Server – The "Click-to-Contact" component is embedded in a customer's web page on the corporate web server.

- Application Server – The Application Server hosts the Web Interaction SDK application and is responsible for:
    - Intermediating the registration process of the Click-to-Contact with the OpenScape Contact Media Service.
    - Load balancing between the multiple OpenScape Contact Media Service nodes, for scalability or in case of redundancy scenarios.
    - Intermediating the authentication process of the Click-to-Contact with the TURN Server (see OpenScape Contact Media Service below).

3. Deployment Scenarios

3.1 Integrated Phone

The Integrated Phone feature can be deployed in a standalone or in a redundant system.

In a standalone deployment scenario, there is only one instance of each component. Because of this design, whenever a component has an issue, the functionality will not work as expected.

In a redundant system there are multiple instances of each component in such a way that if one instance has issues, another instance will take over the functionality and keep the service running.

The Integrated Phone can be either located inside the company's local area network resp. in a VPN (Internal Access) or on the Internet (External Access).

3.1.1 Standalone System

The standalone system contains only one instance of the Application Server, OpenScape Contact Media Service, OSCC and eventually the Load Balancer.

3.1.1.1 Internal Access

When the Integrated Phone is located the company's local area network, no TURN server is required.

The media between the Agent Portal Web and the OpenScape Contact Media Service is exchanged via the SRTP protocol. WebRTC requires using DTLS as the SRTP key negotiation mechanism.

Note: For the complete list of ports please verify the Security Checklist or the IFMDB tool.

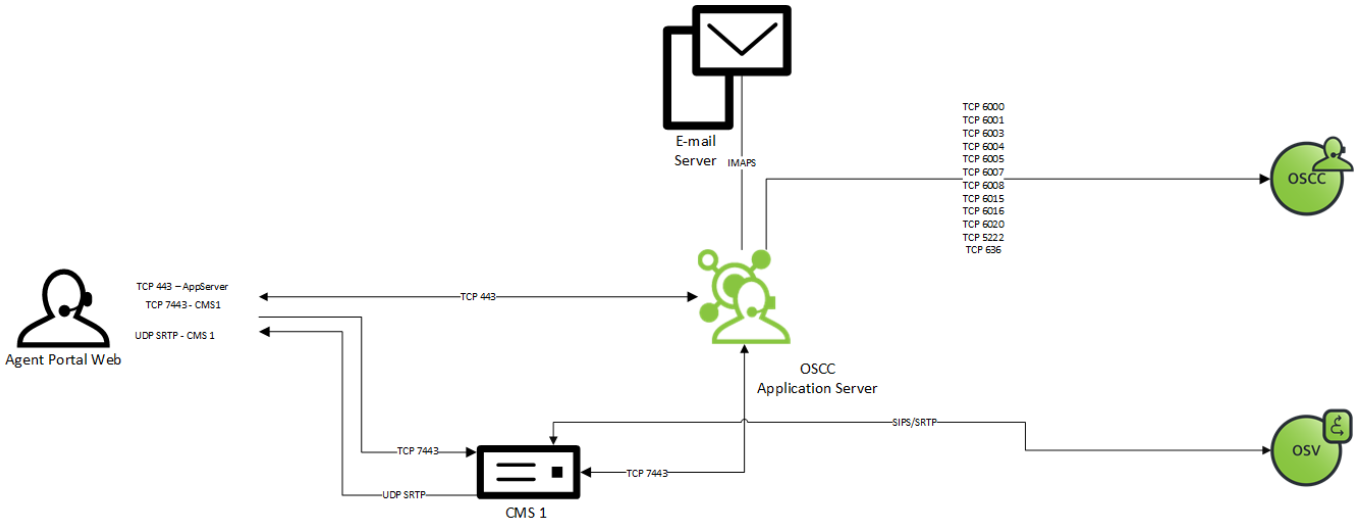


Figure 1 WebRTC Standalone Internal Access

3.1.1.2 External Access

If the Integrated Phone is connected to the OpenScape Contact Media Service server via the Internet, the TURN server needs to be enabled on the OpenScape Contact Media Service server to act as a media relay. In this scenario the Firewall must perform the NAT between the public IP address and the private IP address for the OpenScape Contact Media Service.

The media between the Agent Portal Web in the browser and the TURN server is exchanged by means SRTP messages that are tunneled via the TURN protocol. Since this is a real time media flow, it is highly recommended to run TURN protocol over UDP.

Note: The TURN protocol should only run over TCP if there are security restrictions, which prevent to use UDP. It must be clear that the use of TCP to carry real time media can lead to performance problems, due to the nature of the TCP protocol.

Note: For the complete list of ports please verify the Security Checklist or the IFMDB tool.

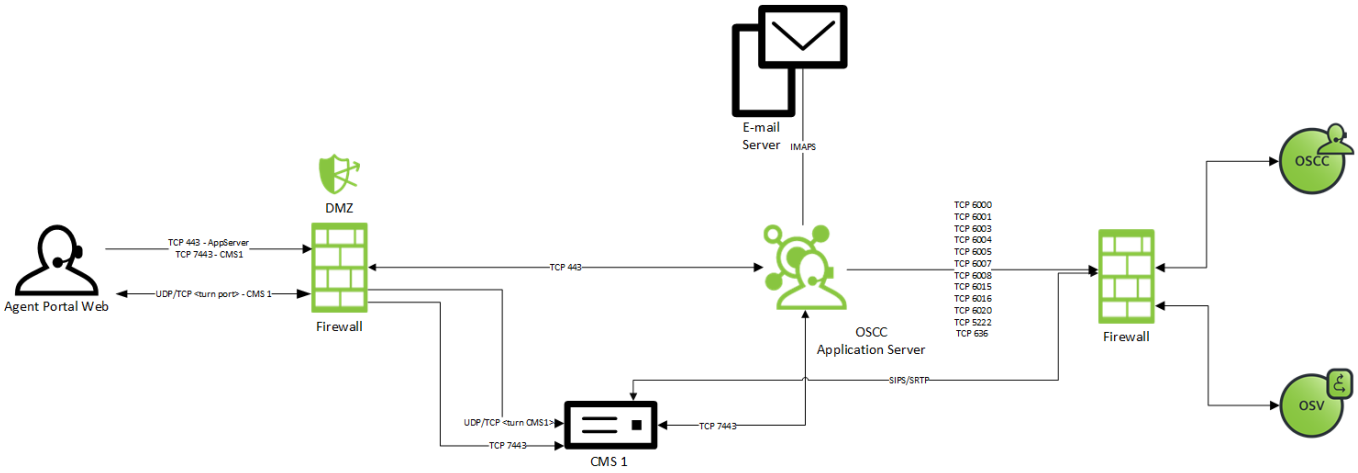


Figure 2 WebRTC Standalone External Access

3.1.2 Redundant System

Any component of the Integrated Phone solution can be deployed redundantly, for example, by multiple Application Servers and/or multiple OpenScape Contact Media Service Servers. In an environment which requires high availability, it is highly recommended to have all components of the solution redundant. Otherwise, when the component is not redundant and it has an issue, the WebRTC functionality will also present issues.

The Load Balancer (for example HAProxy) performs the load balancing between the Application Servers and the Application Server(s) perform the load balancing between the OpenScape Contact Media Service nodes.

3.1.2.1 Internal Access

When the Integrated Phone is located in the company's local area network, then no TURN server is required.

The media between the Agent Portal Web in the browser and the OpenScape Contact Media Service is exchanged via the SRTP protocol. The WebRTC requires DTLS as an SRTP key negotiation mechanism.

Note: For the complete list of ports please verify the Security Checklist or the IFMDB tool.

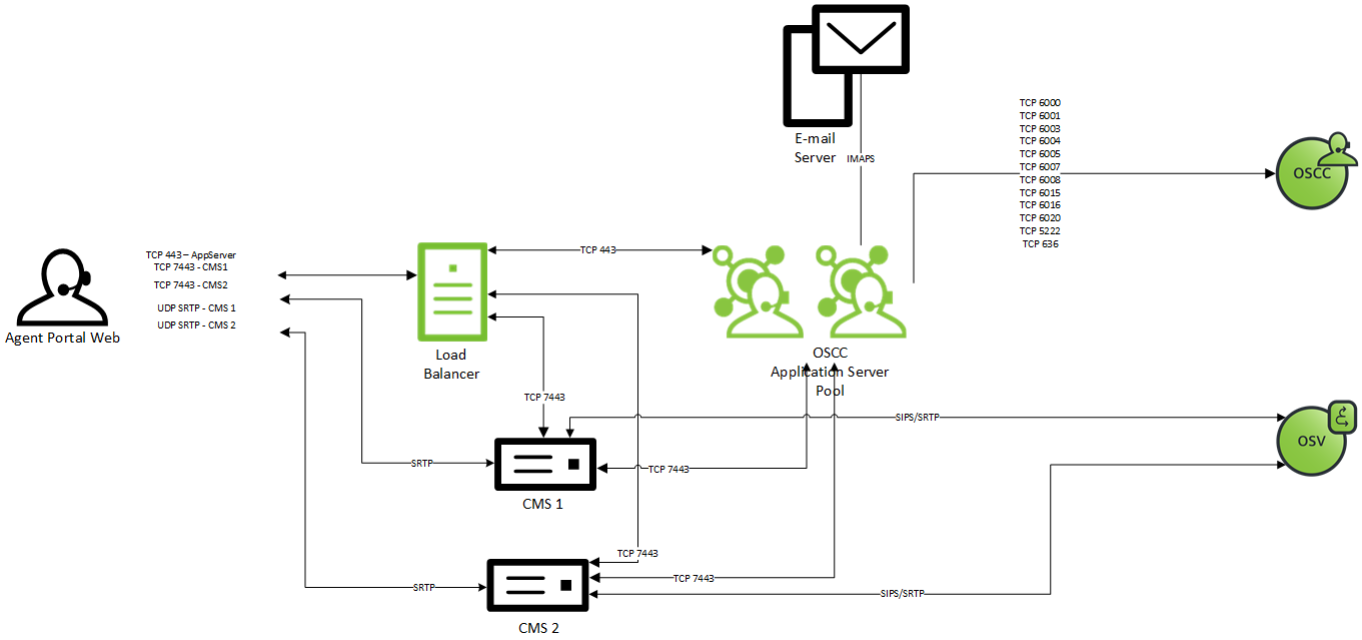


Figure 3 WebRTC Redundant Internal Access

3.1.2.2 External Access

When the Integrated Phone is connected to the OpenScape Contact Media Service server via the Internet, the TURN server needs to be enabled on the OpenScape Contact Media Service server to act as a media relay.
In this scenario the Firewall must perform the NAT between the public IP address and the private IP address for the OpenScape Contact Media Service.

The media between the Agent Portal Web in the browser and the TURN server is exchanged by means tunnelled SRTP messages in the TURN protocol. Since this is a real time media flow, it is highly recommended to run TURN protocol over UDP.

Note: The TURN protocol should only run over TCP if there are security restrictions, which prevent to use UDP. It must be clear that the use of TCP to carry real time media can lead to performance problems, due to the nature of the TCP protocol.

Note: For the complete list of ports please verify the Security Checklist or the IFMDB tool.

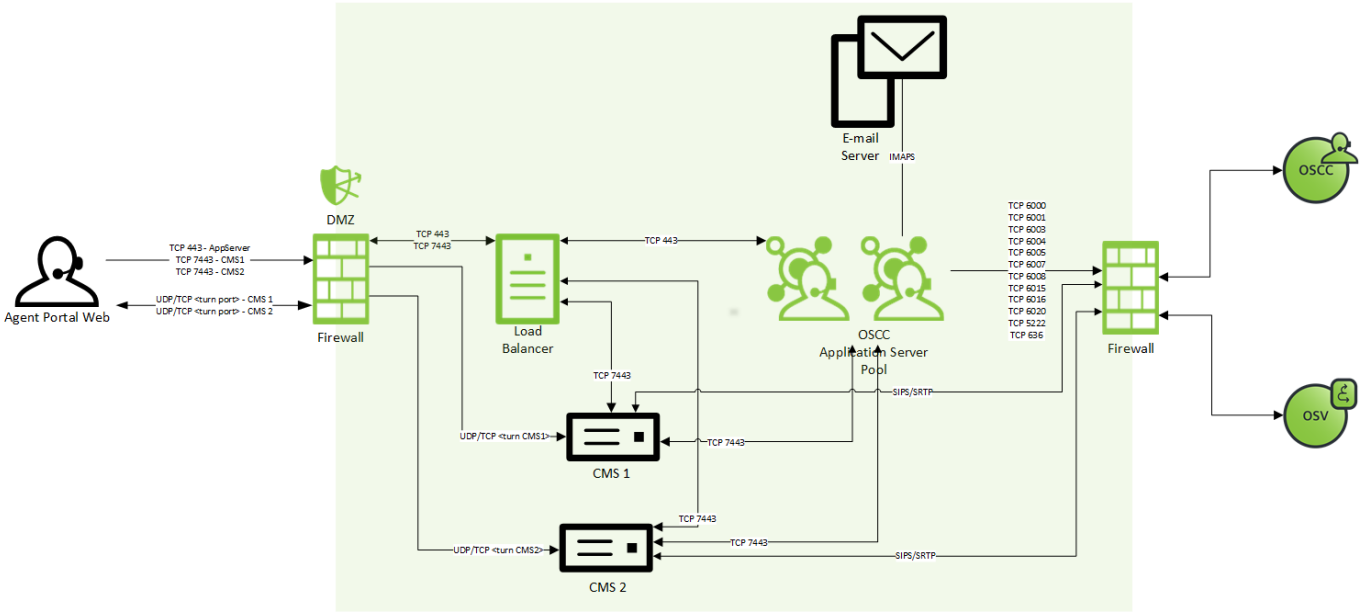


Figure 4 WebRTC Redundant External Access

3.2 Click-to-Contact

The Click-to-Contact feature can be deployed in a standalone or in a redundant system.

In a standalone deployment scenario, there is only one instance of each component. Because of this design, whenever a component has an issue, the functionality will not work as expected.

In a redundant system there are multiple instances of each component in such a way that if one instance has issues, another instance will take over the functionality and keep the service running.

3.2.1 Standalone System

The standalone system contains only one instance of the Application Server with the Web Interaction SDK component, OpenScape Contact Media Service, OSCC and eventually a Load Balancer (e.g. HAProxy).

3.2.1.1 Internal Access

When the Click-to-Contact is located the company's local area network, no TURN server is required.

The media between the Agent Portal Web and the OpenScape Contact Media Service is exchanged via the SRTP protocol. WebRTC requires using DTLS as the SRTP key negotiation mechanism.

Note: For the complete list of ports please verify the Security Checklist or the IFMDB tool.

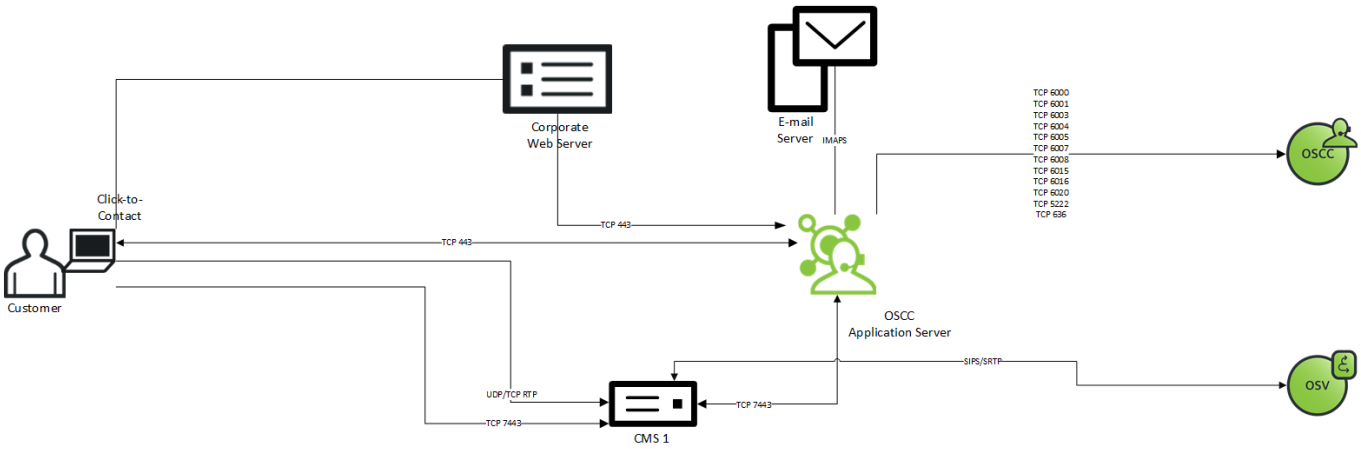


Figure 5 WebRTC Click-to-Contact Standalone System Internal Access

3.2.1.2 External Access

If the Click-to-Contact is connected to the OpenScape Contact Media Service server via the Internet, the TURN server needs to be enabled on the OpenScape Contact Media Service server to act as a media relay. In this scenario the Firewall must perform the NAT between the public IP address and the private IP address for the OpenScape Contact Media Service.

The media between the Agent Portal Web in the browser and the TURN server is exchanged by means of SRTP messages that are tunneled in the TURN protocol. Since this is a real time media flow, it is highly recommended to run TURN protocol over UDP.

Note: The TURN protocol should only run over TCP if there are security restrictions, which prevent to use UDP. It must be clear that the use of TCP to carry real time media can lead to performance problems, due to the nature of the TCP protocol.

Note: For the complete list of ports please verify the Security Checklist or the IFMDB tool.

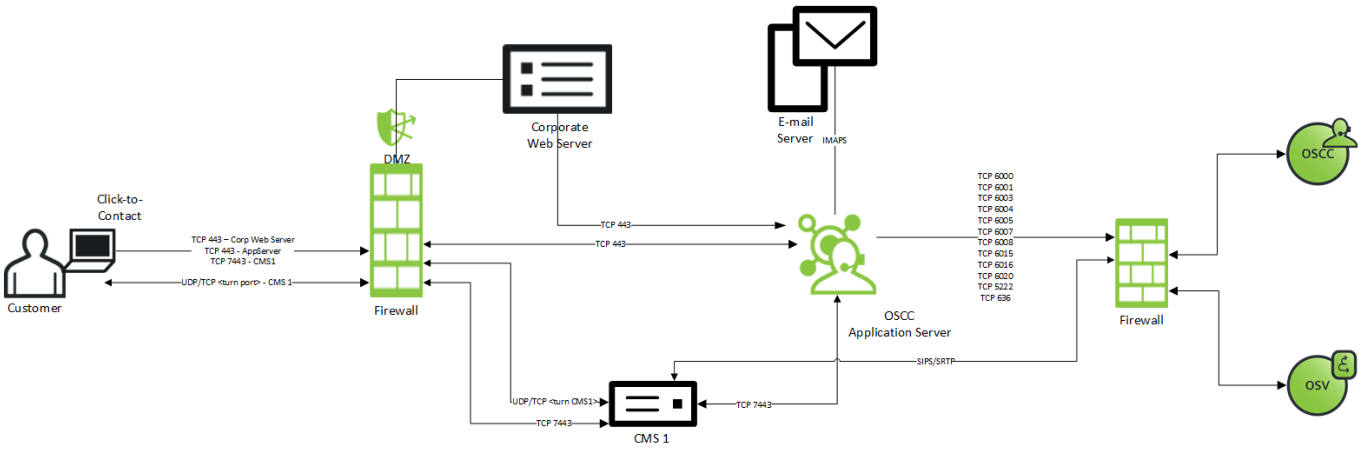


Figure 6 WebRTC Click-to-Contact Standalone System External Access

3.2.2 Redundant System

Any component of the Click-to-Contact solution can be deployed redundantly, for example, by multiple Application Servers with the Web Interaction SDK component and multiple OpenScape Contact Media Service Servers. In an environment which requires high availability, it is highly recommended to have all components of the solution redundant. Otherwise, when the component is not redundant and it has an issue, the WebRTC functionality will also present issues.

The load balancing to the Web Interaction SDK component in the Application Server can be performed by a commercial Load Balancer (e.g. F5 or Netscaler) or an HAProxy component.

The load balancing to the OpenScape Contact Media Service nodes will be performed by the Web Interaction SDK component in the Application Server.

3.2.2.1 Internal Access

When the Click-to-Contact is located in the company's local area network, then no TURN server is required.

The media between the Agent Portal Web in the browser and the OpenScape Contact Media Service is exchanged via the SRTP protocol. The WebRTC requires DTLS as an SRTP key negotiation mechanism.

Note: For the complete list of ports please verify the Security Checklist or the IFMDB tool.

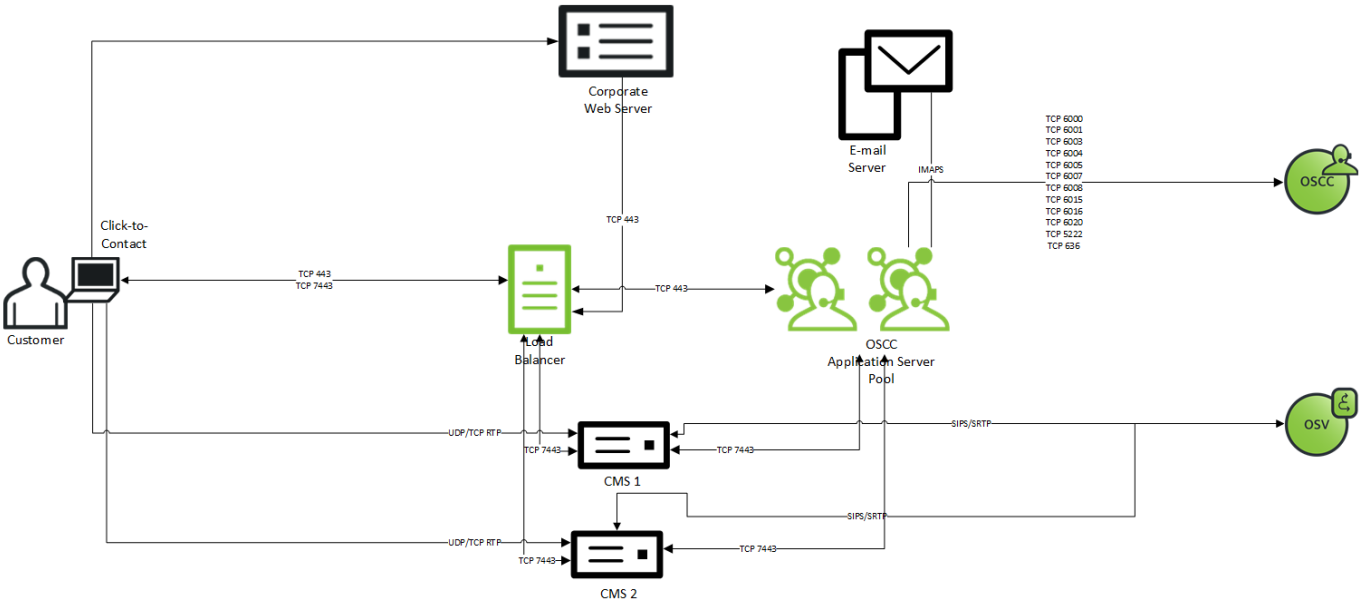


Figure 7 WebRTC Click-to-Contact Redundant System External Access

3.2.2.2 External Access

When Click-to-Contact is connected to the OpenScape Contact Media Service server via the Internet, the TURN server needs to be enabled on the OpenScape Contact Media Service server to act as a media relay. In this scenario the Firewall must perform the NAT between the public IP address and the private IP address for the OpenScape Contact Media Service.

The media between the Agent Portal Web in the browser and the TURN server is exchanged by means of SRTP messages that are tunneled in the TURN protocol. Since this is a real time media flow, it is highly recommended to run TURN protocol over UDP.

Note: The TURN protocol should only run over TCP if there are security restrictions, which prevent to use UDP. It must be clear that the use of TCP to carry real time media can lead to performance problems, due to the nature of the TCP protocol.

Note: For the complete list of ports please verify the Security Checklist or the IFMDB tool.

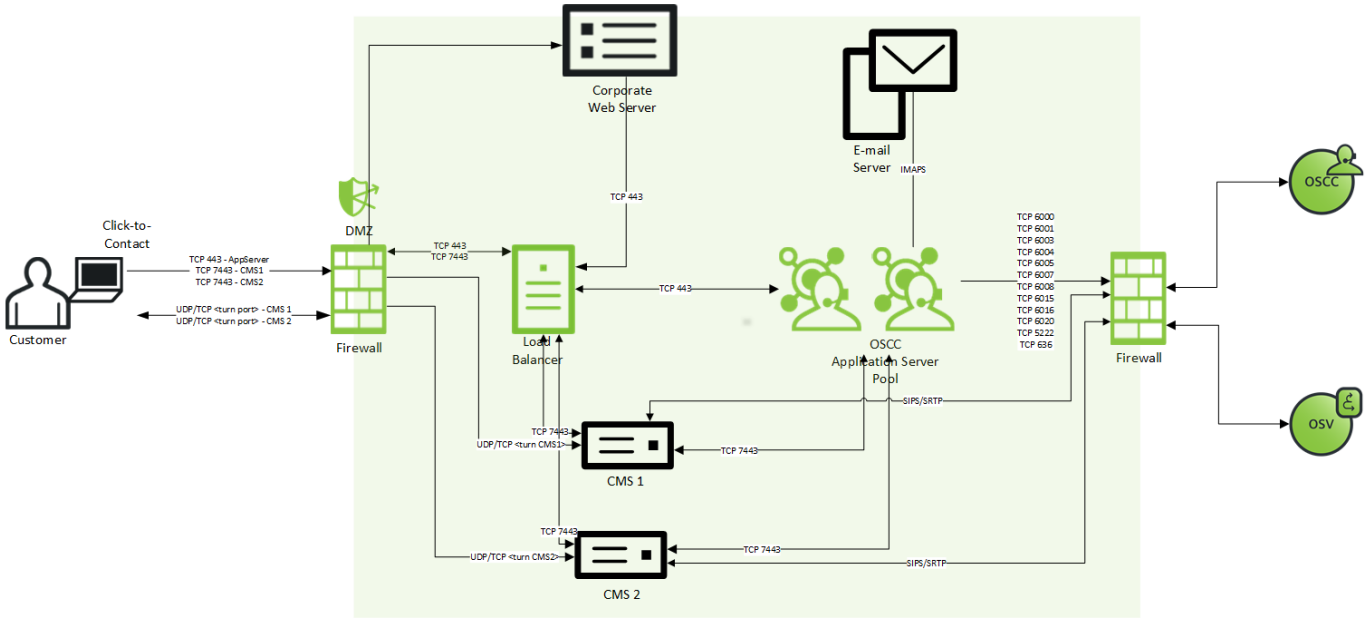


Figure 8 WebRTC Click-to-Contact Redundant System External Access

4. Infrastructure

4.1 System Requirements

4.1.1 OpenScape Contact Media Service

The minimum system requirements for installing the OpenScape Contact Media Service software on a stand-alone server machine, supporting up to 200 WebRTC ports, are described in the following table:

Requirement	Description
Processor	Intel Xeon E-2174G 3.80GHz
Memory	4 GB
Hard Drive	160 GB, 7200 RPM, SATA
Display settings	1024 x 768 pixels with 16-bit color
Other	1 Gbps Ethernet network interface card DVD-ROM drive

Table 1 System requirements for a stand-alone server machine with up to 200 WebRTC ports

The minimum system requirements for installing the OpenScape Contact Media Service software on a stand-alone server machine supporting a range from 200 to 300 WebRTC ports are described in the following table:

Requirement	Description
Processor	Intel Xeon E-2174G 3.80GHz
Memory	8 GB
Hard Drive	160 GB, 7200 RPM, SATA

Display settings	1024 x 768 pixels with 16-bit color
Other	1 Gbps Ethernet network interface card DVD-ROM drive

Table 2 System requirements for a stand-alone server machine from 200 up to 300 WebRTC ports

The minimum system requirements for installing the OpenScape Contact Media Service software on a stand-alone server machine supporting up to 100 WebRTC ports and 100 Recorder ports are described in the following table:

Requirement	Description
Processor	Intel Xeon Silver 4208 2.10GHz
Memory	8 GB
Hard Drive 1	160 GB, 7200 RPM, SATA
Hard Drive 2	>=960GB, SSD, SATA/SAS Enterprise >= 960GB, >=10k RPM, SAS Enterprise
Display settings	1024 x 768 pixels with 16-bit color
Other	1 Gbps Ethernet network interface card DVD-ROM drive

Table 4 System requirements for a stand-alone server machine with up to 100 WebRTC ports and 100 Recorder ports

The OpenScape Contact Media Service software is a hardware independent application and can be installed in a wide range of hardware by following the requirements above. However, it is possible that some specific hardware is not covered by the delivered drivers in the OpenScape Contact Media Service installation.

Note: For details about the Virtualization Dimensioning of the OpenScape Contact Media Service, please refer to the OpenScape Solution Set V10, OpenScape Virtual Machine Resourcing and Configuration Guide, Service Documentation.

Note: The assigned IP to the OpenScape Contact Media Service must be Static. When using DHCP, make sure that the DHCP server will not change the assigned IP during normal operation. When the IP is changed directly via OS, the Media Server service must be restarted (or the machine rebooted) in order for the Media Server to update the IP on the system. Otherwise, the OpenScape Contact Media Service loses connection to the Communication Platform.

4.1.2 Application Server

The application server can be installed on the OSCC server or on a separate server, depending on deployment consideration. A separate server will be considered to support redundancy and scalability and in case of WebRTC external access.

For the Application Server:

Requirement	Up to 750 active users	More than 750 active users
Processor	Intel Xeon E3-1271v3	Two Intel Xeon E5-2609v2
Memory	10 GB (See below)	10 GB (See below)
Hard Drive	1 TB, 7200 RPM, SATA	1 TB, 7200 RPM, SATA
Operating System	• Windows Server 2022 Standard or Datacenter • Windows Server 2016 Standard or Datacenter • Windows Server 2012 R2 Standard or Datacenter • Windows Server 2012 Standard or Datacenter	
Other	• Teamed network interface card c to provide adapter fault tolerance for the customer LAN (optional) • Redundant disks, power supplies, and cooling units (optional, but highly recommended) • JRE 64Bit installed	

Table 7 System requirements for the Application Server

For an Application Server machine with minimum resources of:

vCPUs: 4
Memory: 8 GB for the TomCat / 10 GB for the System.

In addition, **maxThreads** shall be as **650 (maxThreads=”650”)** in the **server.xml** file, at **\\Program Files (x86)\OpenScape\Contact Center\ApplicationServer\ApacheWebServer\conf**.

Note: If “chat between agents” feature is not used, please disable it from all agents´ user permissions.

We illustrate with 2 scenarios:

1) **Voice & Callback / all agents with avatar / small-mediumTeamlist / Openfire-“chat between agents”**

Media:	Voice (12.000 BHCA) and Callback (750 BHCA)
Avatar:	450 agents having Avatar
Team list entries:	75 agents with 150 entries and 375 agents with ca. 20 entries
Users’ permissions:	All agent’s setup with “chat between agents” permission enabled
Chat between agent:	Configured, but not used.
Shifts:	Shifts having login/logoff of all agents being executed.

For this scenario, the maximum number of Agents per Application should be: 450

2) **Voice, e-mail & chat / all agents with avatar / medium Teamlist / Openfire-“chat between agents”**

Medias:	Voice (12.000 BHCA), e-mail (600 BHCA) and chat (800 BHCA)
Avatar:	300 agents having Avatar
Team list entries:	ca. 150 entries in each one of the 300 agents.
Users’ permissions:	All agent’s setup with “chat between agents” permission enabled
Chat between agent:	Configured, but not used.
Shifts:	Shifts having login/logoff of all agents being executed.

For this scenario, the maximum number of Agents per Application should be: 300

Note: For details about the Virtualization Dimensioning of the OpenScape Contact Center – Application Server, please refer to the OpenScape Solution Set V10, OpenScape Virtual Machine Resourcing and Configuration Guide, Service Documentation.

4.2 Firewall

The Firewall must perform the NAT between the public IP address and the private IP address of the OpenScape Contact Media Service.

The following ports must be open in the Firewall:

- Port 443 – To the Application Server
- Port 7443 –To the OpenScape Contact Media Service server. Please note that this is the default HTTPS port and it is configurable.
- Port 3478 – To the TURN server (on OpenScape Contact Media Service server). Notice that this is the default port for STUN/TURN and it is configurable.

4.3 DNS

The FQDN part of the OpenScape Contact Media Service URL must be resolved by external DNS to the public IP address.

The FQDN part of the OpenScape Contact Media Service URL must be resolved by internal DNS to the private IP address.

The FQDN part of the Application Server URL must be resolved by external DNS to the public IP address.

The FQDN part of the Application Server URL must be resolved by internal DNS to the private IP address.

4.4 Security

Some security considerations regarding to the Integrated Phone solution:

- The agent in Agent Portal Web is authenticated by the Application Server (via authentication by OSCC or via SSO with SAML2).
- The users for the OpenScape Contact Media Service server administration can be configured to follow policies which are managed via

a PAM module in OpenScape Contact Media Service Linux. The Security Checklist provides instructions on how to configure the password policies in OpenScape Contact Media Service.

- The Application Server is authenticated by OpenScape Contact Media Service via OAuth (Client Id + Client Secret).
- The Integrated Phone (WebRTC client) is intrinsically authenticated by OpenScape Contact Media Service via a dynamic session token that is negotiated via Application Server.
- The Integrated Phone (WebRTC client) is intrinsically authenticated by the TURN server via TURN credentials which are negotiated via Application Server.
- Each Integrated Phone corresponds to a SIP subscriber in OpenScape Voice/ OpenScape 4000. The SIP Subscribers can be protected by authentication via Digest Authentication.
- The HAProxy can restrict the access to the OpenScape Contact Media Service configuration web page. In the case of a single Application Server and single OpenScape Contact Media Service, the HAProxy may be removed if the OpenScape Contact Media Service user is configured with stronger policies.
- All connections between the solution components are encrypted via HTTPS or pure TLS. Notice that WebRTC only supports SRTP via DTLS. **Note**: The only connection that is not encrypted is the communication between Agent Portal Web (Integrated Phone) and the TURN server (TURN over UDP) because this is used to transport media. However, this is not a security risk, because the transported media is encrypted via SRTP.
- The Application Server is provided by default with Self Signed certificates, but it is strongly recommended to replace them by certificates generated by a CA.
- The Certificate for WebRTC in OpenScape Contact Media Service must be a valid one which is generated by a Certificate Authority.

4.5 Certificates

A Server Certificate is needed for the Application Server and for the OpenScape Contact Media Service.

Note: It is recommended to generate certificates with known Certificate Authorities.

For the Application Server and for the OpenScape Contact Media Service server, a company wildcard certificate can be used. However, there can be problems with the company security policies regarding the handling of the corresponding private key.

4.5.1 OpenScape Contact Media Service

When the Integrated Phone is activated, the Agent Portal Web in the browser will directly contact the OpenScape Contact Media Service. So, the browser will validate the certificate against the FQDN which is used to access OpenScape Contact Media Service.

For the Integrated Phone, the certificate for OpenScape Contact Media Service must be generated by a known Certificate Authority so that it is validated by the browser. The server X.509 certificate, the CA certificate and the private key must be in PEM format (BASE64).

In addition, this certificate needs to have a SAN field (SubjectAlternativeName) of type DNS that contains the FQDN of the OpenScape Contact Media Service server.

Note: if the Server Certificate is generated from a private PKI, the CA Certificate and the Root Certificate which were used to sign the Server Certificate must be installed in the agent computers.

Usually the certificate is provided by the Certificate Authority in a PKCS12 file (.p12 or .pfx files). The certificates must be extracted from the PKCS12 file by using the following procedure:
Note: The openssl application can be used for the conversion.

1. Convert the certificate from PKCS12 to PEM by using the following command:
   `openssl pkcs12 -in <.p12 or .pfx filename> -out cert.pem -nodes`
   A text file is generated with the certificates and the private key.
2. Open the cert.pem file with a text editor.
3. Look for the lines “-----BEGIN PRIVATE KEY-----” and “-----END PRIVATE KEY-----“.
4. Copy the text between these lines to a file called privatekey.pem.
5. Look for the lines “-----BEGIN CERTIFICATE-----” and “-----END CERTIFICATE-----“.
6. Identify the public certificate.
7. Copy the text for the public certificate to a file publiccert.pem.
8. Identify the CA certificate(s).
9. Copy the text for the CA certificate(s) to a file CAcert.pem.

To install the certificate for the OpenScape Contact Media Service configuration tool, the certificate files and private key file shall be selected and uploaded to the corresponding fields Public Certificate, Private Key and CA Certificate for HTTPS Certificates.

4.5.2 Application Server

The OpenScape Contact Center Application Server uses a keystore in format JKS or PKCS12. To replace the default certificate by a certificate generated by a Certificate Authority in the OpenScape Contact Center Application Server, the following process must be followed:

- Obtain the certificate from the Authority.
- Identify the Java instance, which is being used by Tomcat on the Application Server by checking the environment variable JRE_HOME.
- Change to the directory of JRE_HOME, as for example:

```
cd "\Program Files\OpenScape\Contact Center\Java\IBM\jre”
```

Note: IBM Java from the OpenScape Contact Center V10 DVD must be used and by default the JRE_HOME folder shall be "\Program Files\OpenScape\Contact Center\Java\IBM\jre".

- Tomcat can access the Certificate and Private key in two ways:
    - as a pkcs12 keystore or
    - as a Java keystore.

1. When the decision is to use the pkcs12 keystore (mystore.p12), just configure the server.xml file with the proper keystore path plus password, as for example:

```
<Connector port="443"
protocol="org.apache.coyote.http11.Http11NioProtocol"
maxThreads="150" SSLEnabled="true" scheme="https"
secure="true" clientAuth="false" sslProtocol="TLS">
keystoreFile="${catalina.base}\conf\mystore.p12"
keystorePass="<PKCS#12 keystore password>"
keystoreType="PKCS12"/>
```

2. When the decision is to use a Java keystore, then the pkcs12 keystore must be imported into a Java keystore. The following command shall be used:

```
.\bin\keytool.exe -importkeystore -srckeystore
<path>\mystore.p12 -srcstoretype PKCS12 -destkeystore
"${catalina.base}\conf\keystore.jks" -deststorepass <Java
keystore password> -storepass <Java keystore password>
```

Note: By executing this command a password will be requested for the pkcs12 keystore (mystore.p12).
Note: No alias parameter shall be entered.

The keystore name and the corresponding keystore password shall be configured in the file Server.xml which is also in the conf folder as in the example below:

```
<Connector port="443"
protocol="org.apache.coyote.http11.Http11NioProtocol"
maxThreads="150" SSLEnabled="true" scheme="https"
secure="true" clientAuth="false" sslProtocol="TLS">
keystoreFile="${catalina.base}\conf\keystore.jks"
keystorePass="<Java keystore password>" keyPass="<PKCS#12
keystore password>"/>
```

5. Components Configuration

5.1 OpenScape Contact Center

In OpenScape Contact Center, "Integrated Phone" and "Video/Screen Share" are separated licensed features. The Integrated Phone and the Video/Screen Share are licensed on the OpenScape Contact Media Service server only.

The user Permission "Enable Integrated Phone" must be set to "Yes" for those users which will use the "Integrated Phone" and "Video/Screen Share" feature. The “Video/Screen Share” can only be used together with “Integrated Phone”.

The permission must be set in the Administration Center of the Manager application:

User > Permissions > Client Desktop/Agent Portal Permissions > Feature Access.

If additionally, the "Video/Screen Share" feature is used, then for each agent in his Agent Portal Web the option "Enable Video/Screen Share" must be checked:

Settings > Agent > Integrated Phone

Note: Enabling the "Video/Screen Share" option consumes a video license from the OpenScape Contact Media Service server when the agent is logged on. Without enabling it, the agent is still able to handle calls arriving from the "Click-to-Contact" component. However, the controls for video and screen sharing will be disabled and any video or screen sharing started by the external caller will not be shown.

5.2 Communication Platform

The features "Integrated Phone" is currently supported for OpenScape Voice and OpenScape 4000. The "Video/Screen Share" is currently only supported for the OpenScape Voice platform.

5.2.1 OpenScape Voice

5.2.1.1 Integrated Phone

Each Integrated Phone must have a corresponding SIP subscriber in the OpenScape Voice set to “CSTA over SIP”.

5.2.1.2 Video/ Screen Share

Each subscriber using the "Video/Screen Share" feature must be additionally configured with "Video Call" allowed.

If the OpenScape Contact Media Service is connected to OpenScape Voice via an SBC or an SBCThig (in case of OpenScape Enterprise Express), the minimum re-register timer must be set to 300 seconds in the SBC. The following is applicable only if the feature "Video/Screen Share" is used and external callers initiate voice contacts to the contact center via the "Click-to-Contact" component.

- **Adding SIP Trunk Endpoint**

The Click to Contact feature utilizes the Sip Trunk feature from the OpenScape Voice to allow calls from customers.

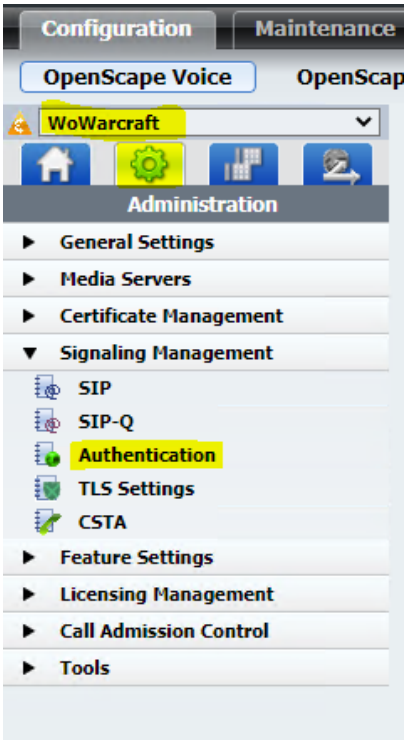
- **Adding OpenScape Contact Media Service as Trusted**

To create an endpoint, the OpenScape Contact Media Service must be set as trusted by the OpenScape Voice.

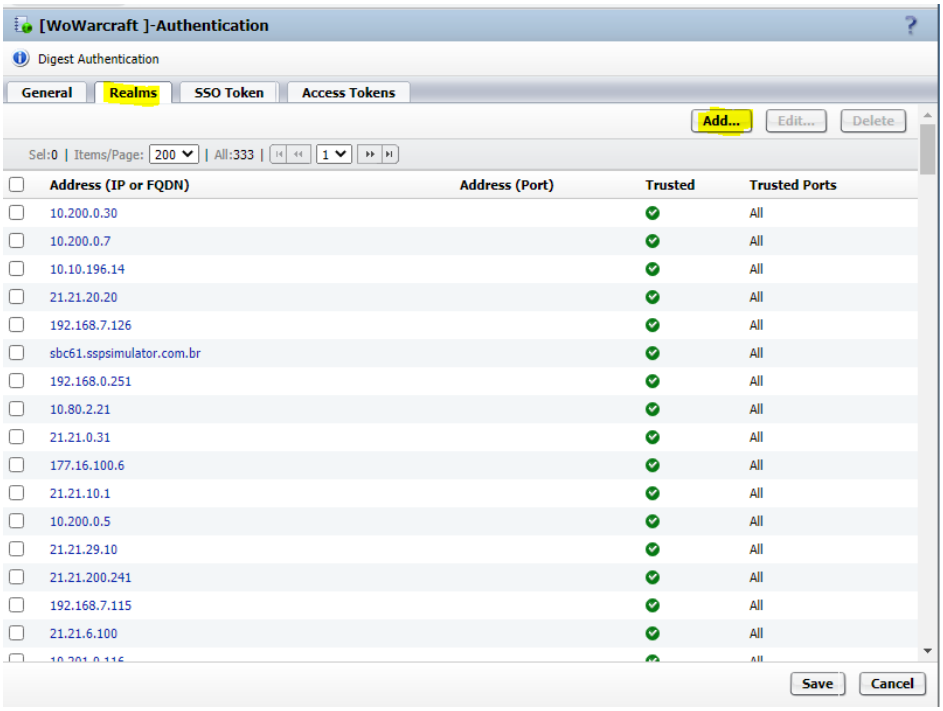
To do so, follow the steps below:

1. Open the switch's **Common Management Platform**;
2. Go to Configuration tab → **OpenScape Voice**

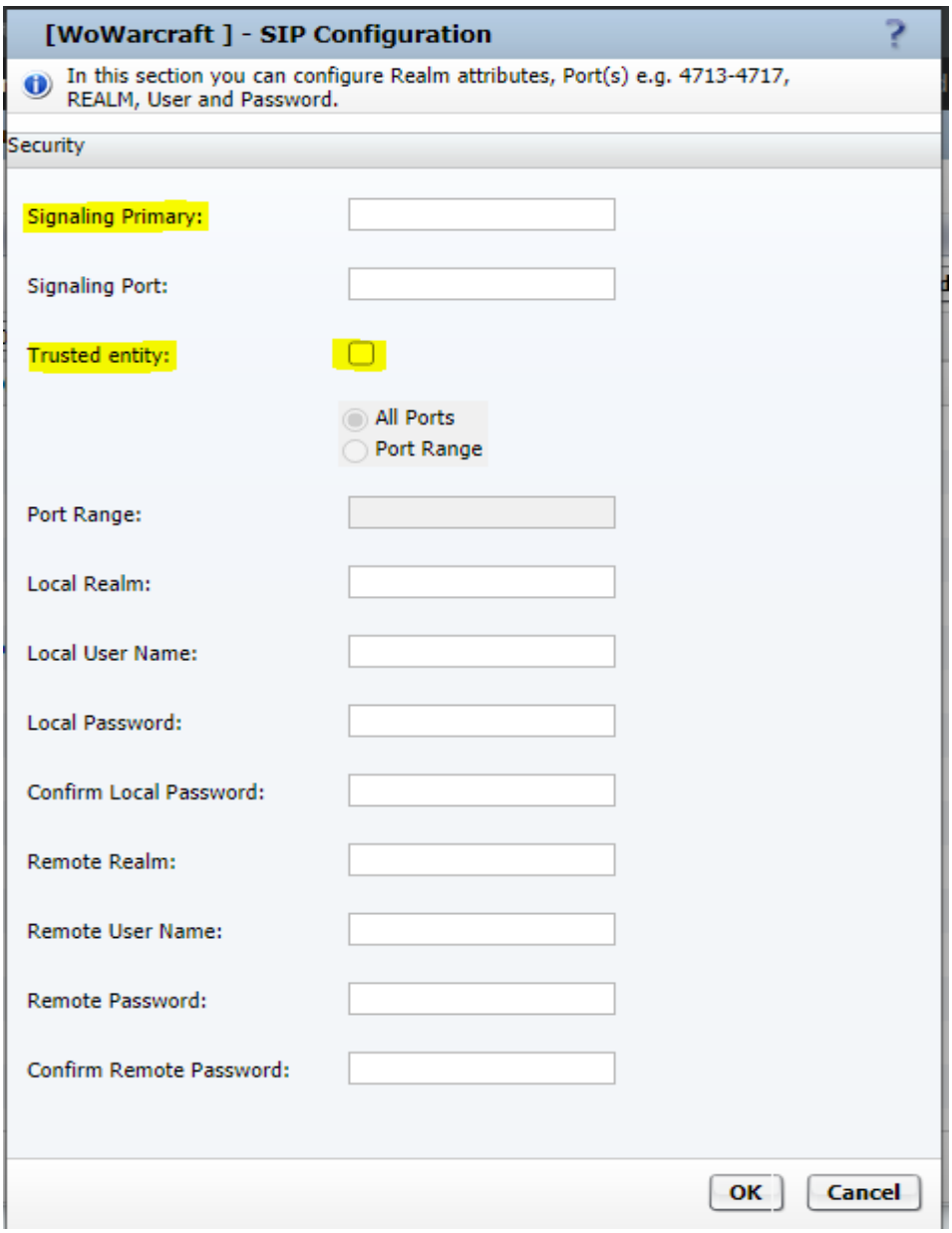
3. Select the appropriate switch and click on the "**Administration**" tab, then go to **Signaling Management → Authentication**:



4. On the popup go to "**Realms**" tab and click "**Add**..."

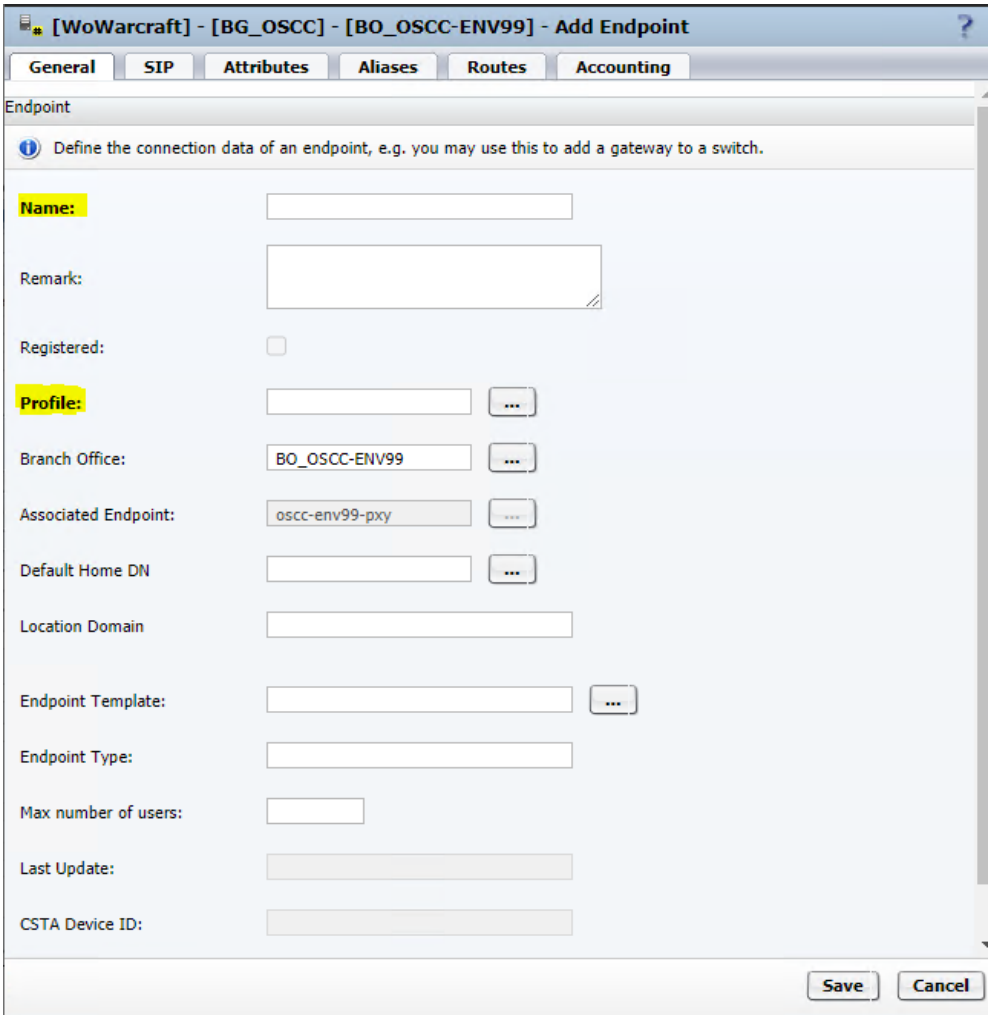


5. On Signaling Primary add the IP of your OpenScape Contact Media Service and mark "**Trusted entity**" and "**All ports**"

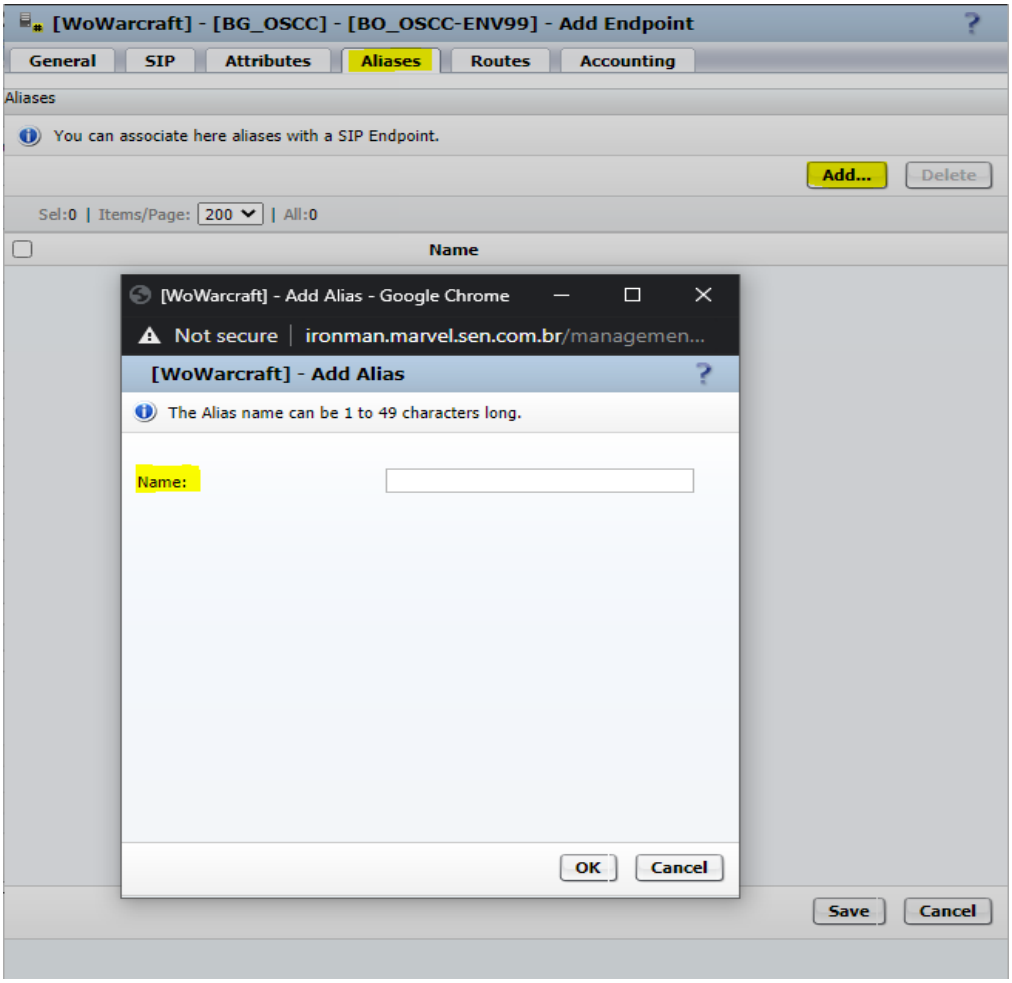


6. Adding an Endpoint
    a. Go to the third tab "Business Group"
    b. Select your business group and select your environment

7. On the endpoints page, click "Add..."
    a. On the general tab, add a Name and select a profile



8. Go to Aliases tab and add a new Alias, which should be the OpenScape Contact Media Service's IP.



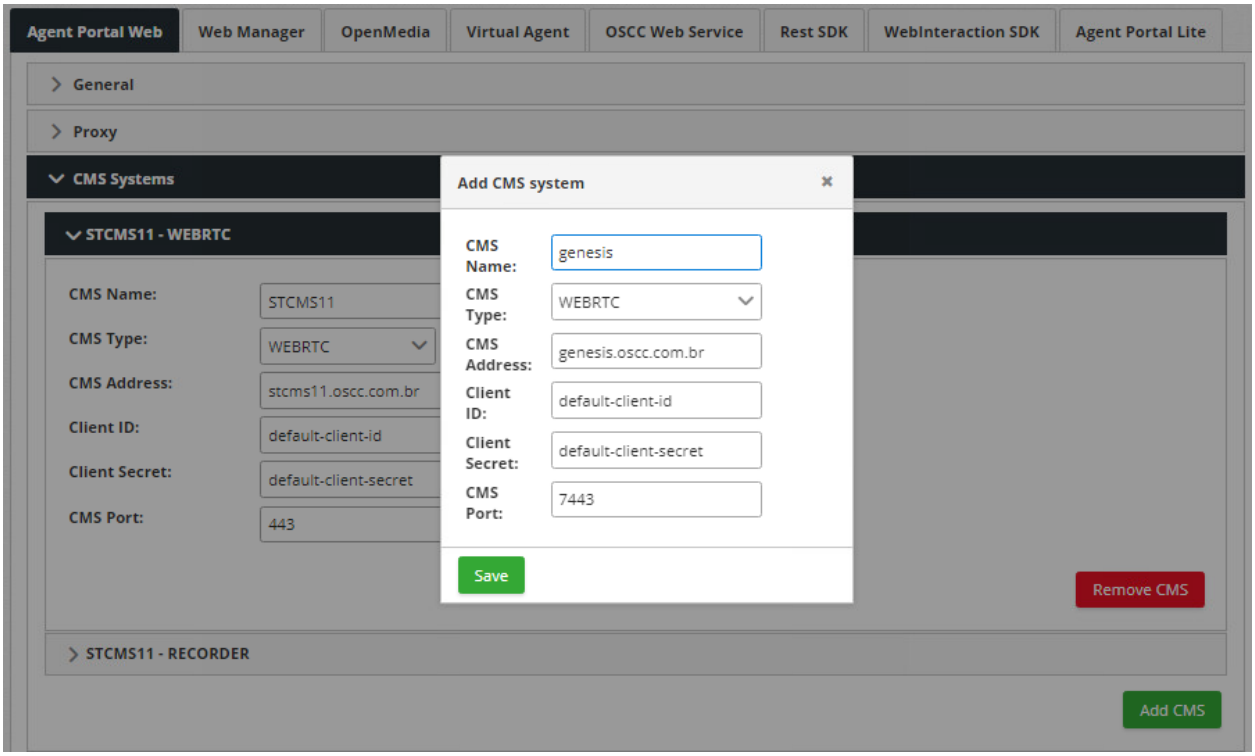
5.2.2 OpenScape 4000

Each Integrated Phone must have a corresponding SIP subscriber in the OpenScape 4000. The SIP subscriber must be configured as UFIP.

5.3 Application Server

The Application Server can communicate with one or multiple OpenScape Contact Media Service servers to register Integrated Phones. The following parameters must be configured to communicate with a OpenScape Contact Media Service:

- **CMS Name**: The name to identify the OpenScape Contact Media Service server on the Application Server.
- **CMS Type**: This parameter is to identify if the CMS is used for WebRTC or Recorder type. If the same CMS will be used for both features, include one register of the same CMS for each type.
- **CMS Address**: This parameter must have the FQDN which will be used to access the OpenScape Contact Media Service and will also be passed to the browser.
- **Client ID**: Identification for the authentication via OAuth. This parameter is also configured on the OpenScape Contact Media Service server.
- **Client Secret**: Secret for the authentication via OAuth. This parameter is configured on the OpenScape Contact Media Service server.
- **CMS Port**: The port number used by connection from the WebRTC Client in the browser to CMS.

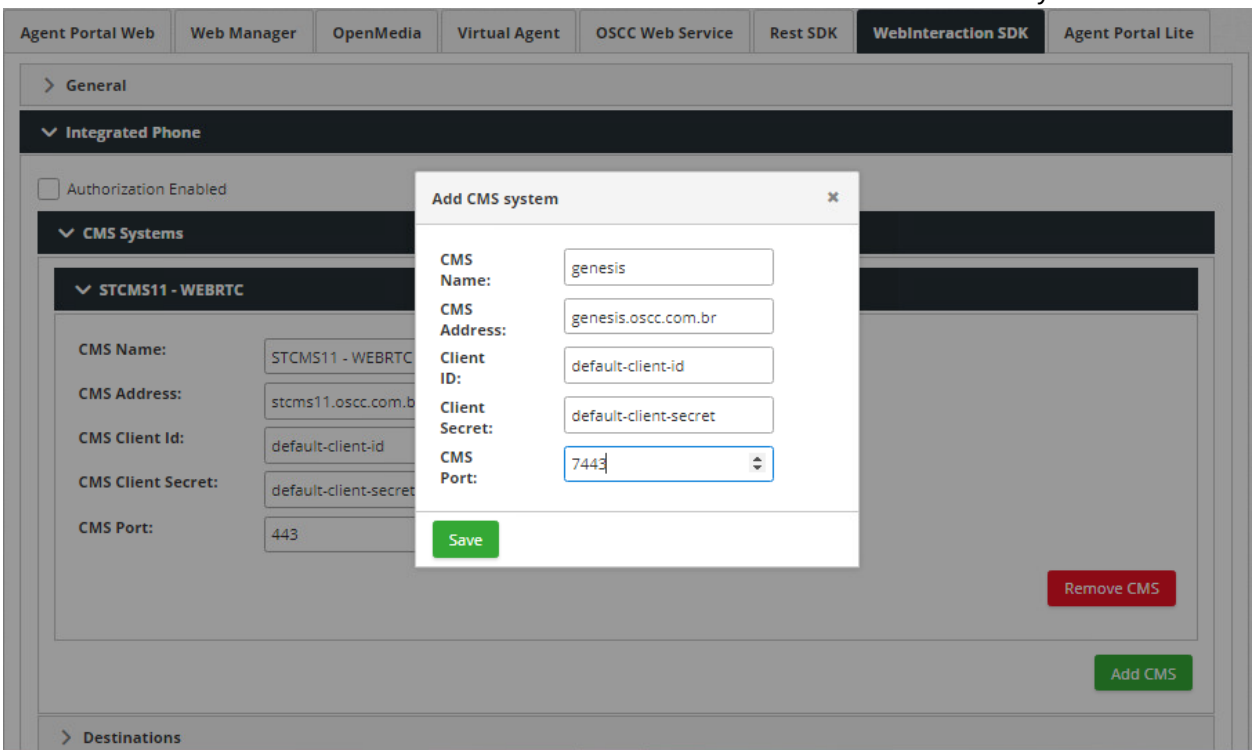


The following configuration is applicable only if the feature "Video/Screen Share" is used and external callers initiate voice contacts to the contact center via the "Click-to-Contact" component.

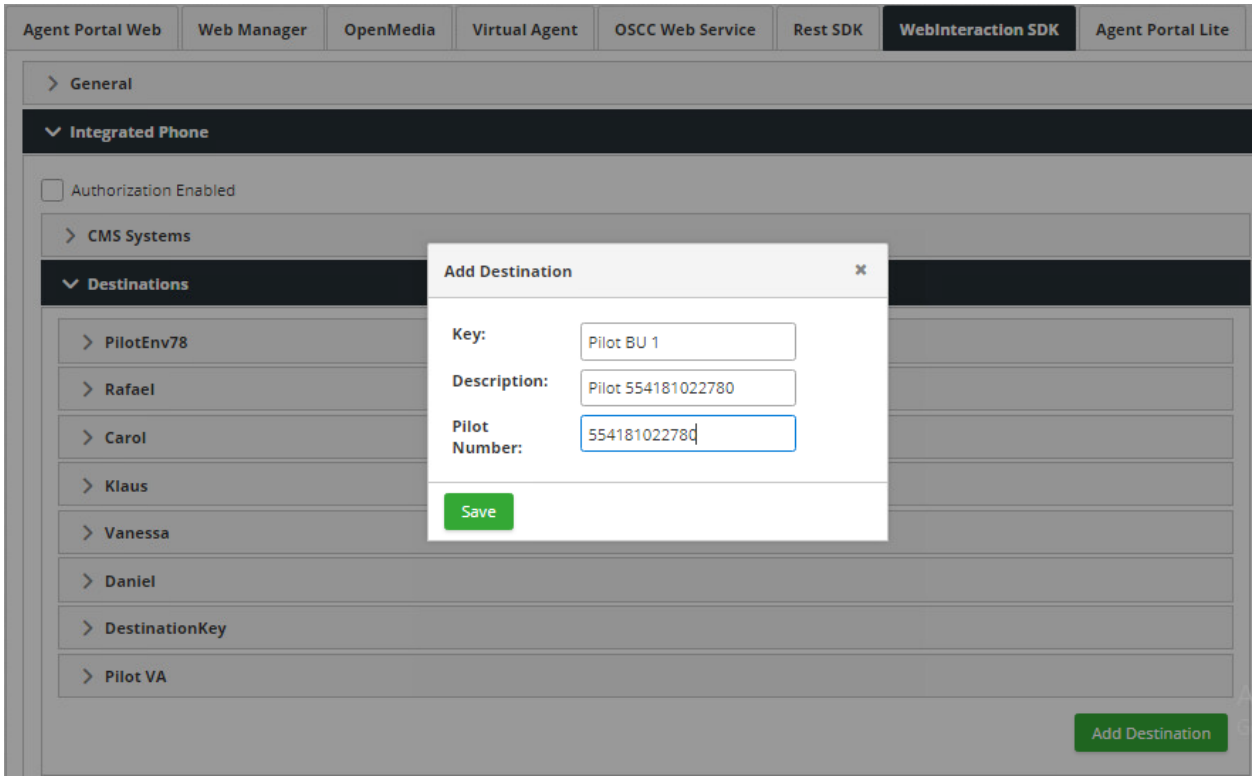
The "Click-to-Contact" component functions via the WebInteraction SDK, which needs to be installed and configured on the OSCC Application Server.

- The SDK modules "webinteractionsdk.war" and "webinteractionsdkexample.war" are provided with the OSCC software in the directory "OpenScape Contact Center Web Components" and subdirectory "Apache Tomcat". For patches the mentioned directories are contained in the file "OSCC<patchversion>WebComponents.zip". Copy both .war files to the directory "ApplicationServer\ApacheWebServer\webapps" of the OSCC Application Server.

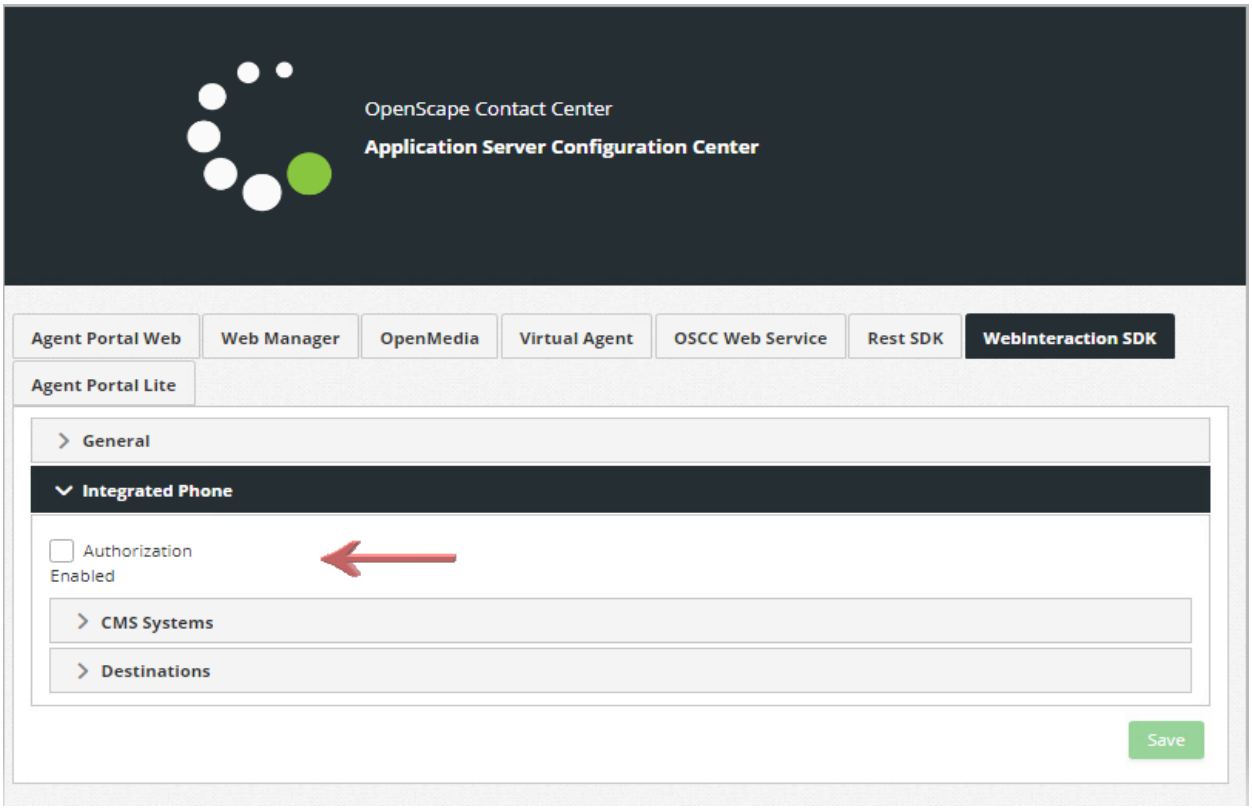
- In the Application Server Configuration Center in the tab "WebInteraction SDK" configure:
    - WebInteraction SDK tab > Integrated Phone > OpenScape Contact Media Service Systems: Add the OpenScape Contact Media Service's used to Click to Contact functionality.



    - WebInteraction SDK tab > Integrated Phone > Destinations: Add the "Destination Key" for one or more pilot numbers. The Destination Keys will be presented on the "Click-to-Contact" component, where the customer will select for which service he wants to call.



- WebInteraction SDK tab > Integrated Phone > Authorization Enabled (optional): Activate in case the "Click-to-Contact" component needs to authenticate with the WebInteraction SDK.



Note: You can perform such a configuration through the Application Server Configuration Center. It is accessible only through the machine where the Application Server is installed (https://localhost/configcenter/index.html).

5.4 OpenScape Contact Media Service

The OpenScape Contact Media Service server has been enhanced to work also as a WebRTC server. The WebRTC server function is provided by the Media Server, which works as a gateway between WebRTC and SIP/RTP to the OpenScape Voice. Each WebRTC subscriber has a corresponding dynamically licensed SIP subscriber in the OpenScape Voice /OpenScape 4000.OpenScape Contact Center monitors the OpenScape Voice /OpenScape 4000 to control the call routing.

For the configuration details about Networking and VoIP, please refer to the OpenScape Contact Media Service V11 R1, Installation Guide.

5.4.1 Licensing

- For the feature "Integrated Phone" a license for the WebRTC ports in OpenScape Contact Media Service is required.
- For the feature "Video/Screen Share" an additional license is required.
- For each WebRTC port used in OpenScape Contact Media Service, there must be a corresponding SIP Subscriber in OpenScape Voice /OpenScape 4000.

5.4.2 Configuration

1. Click the WebRTC icon .
2. Click the Extensions tab. Here all OpenScape Contact Media Service WebRTC extensions (SIP subscribers) are listed. Here you have the option to add a range of extension numbers on which agents using the "Integrated Phone" feature will logon to Voice media. Extensions can be edited and removed as well.

Note: The extensions must be the same as configured in OpenScape Contact Center. Configure just the extensions that will be used as Softphone.

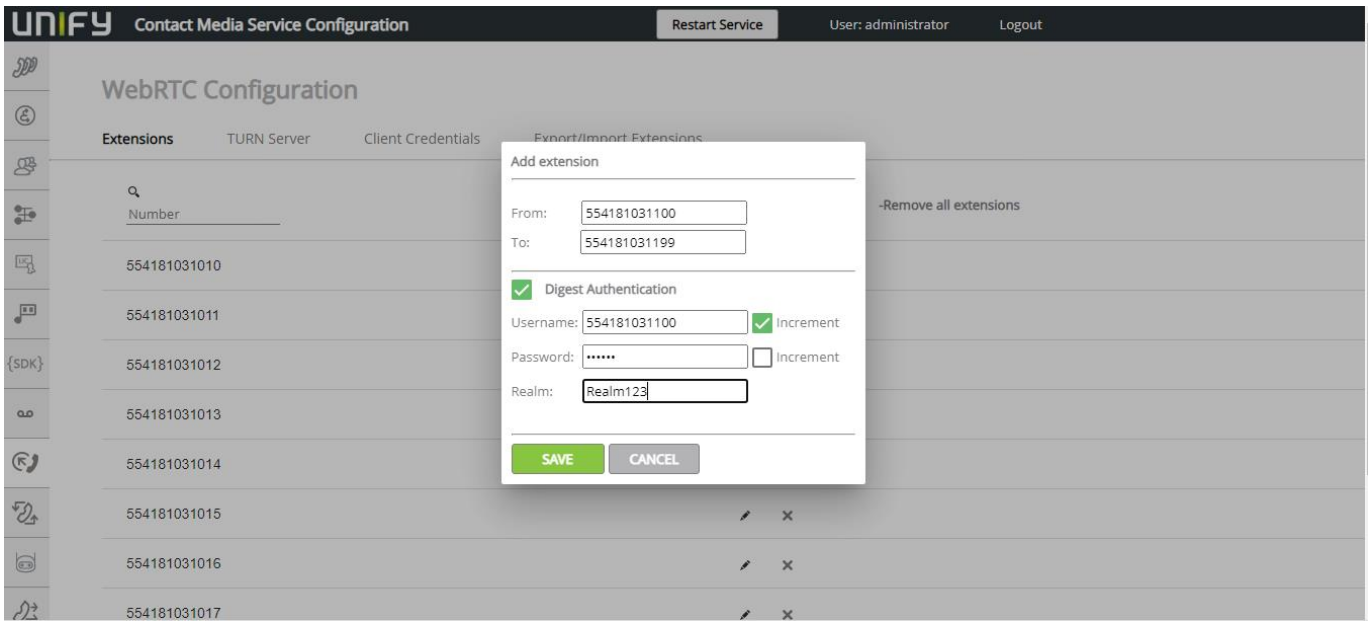
Note: If the switch OS4000 is configured, it is possible to configure the extensions with E.164 format i.e. beginning with the "+" followed by country code and area code.

Note: The customer must not have both Integrated Phone and Deskphone registered on the same subscriber number. Due to limitations on the implementation of Multiple Contacts in the OSV since CSTA Make Call, CSTA Answer and uaCSTA are not recommended to be used with Multiple Contacts.

If Basic IVR (Call Director ports) are being used, the extensions must remain configured with "short" number format in both OpenScape Contact Media Service and OSCC.

- In the **Number search** field, you can search for a specific number to check whether it belongs to a range of extensions.
- Click **+Add extensions**: The pop-up window **Add extension** appears, where you can add a range of extensions:
    - **From**: The lower part of the extensions' range
    - **To**: The upper part of the extensions' range
    - **Digest Authentication**: Check this flag to enable digest authentication. After enabling digest authentication, a new set of parameters appears

        - **Username**: Username used for Digest Authentication. This is a mandatory parameter. Check the Increment flag to enable incrementation
        - **Password**: Password used for Digest Authentication. This is a mandatory parameter. Check the Increment flag to enable incrementation
        - **Realm**: Realm on which the user will be authenticated
    - Click **Save** to add your extension(s) to the list or **Cancel** to discard. The new extension is now part of the list of extensions that will be listed
- Click **-Remove** all extensions to delete all extensions from the list
- Select an individual extension and click the **Edit** icon. The pop-up window **Edit extension** will appear, where you can modify:
    - Enable/disable Digest Authentication
    - When Digest Authentication is enabled:
        - Username
        - Password
        - Realm



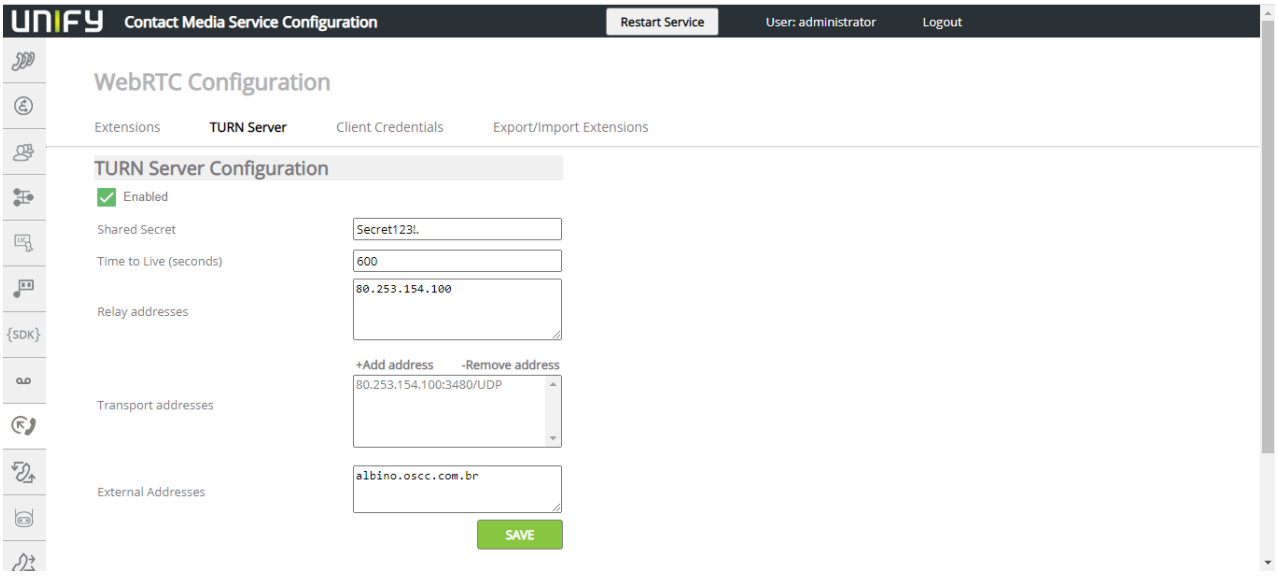
3. Click the TURN Server tab. Here you can configure the TURN server.
   TURN stands for "Traversal Using Relays around NAT".
   It is a standard method of NAT traversal used in WebRTC. The protocol is defined in specification RFC 5766.
   This server must be enabled to use the "Integrated Phone" feature via an

internet (not intranet) connection, e.g when Agent Portal Web users are remotely located and separated from the company network.

Note: Since TURN relays traffic of all media from/to the remote Agent Portal Web applications, this can be expensive in terms of bandwidth and CPU usage at the data center.

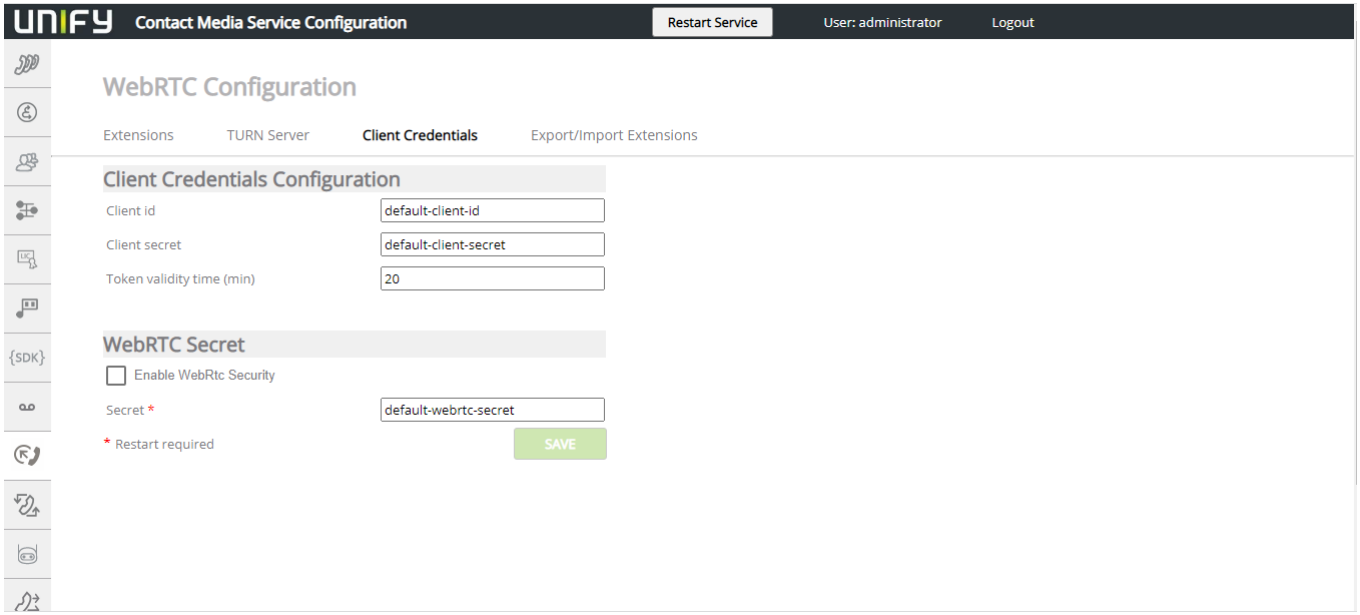
The following parameters can be configured:

- **Enabled**: Check this flag to enable the TURN Server configuration
- **Shared Secret**: A secret key used by OpenScape Contact Media Service to create the TURN account on the Media Server TURN service. The default value is: Secret123!.
- **Time to Live (seconds):** The session time of the TURN service in seconds. Default value: 600.
- **Relay Addresses**: The address used to relay the communication with the peer endpoints. The default value is empty, which means that the system will use the available IP addresses. It is the IP address of the OpenScape Contact Media Service server. It is recommended to use fixed IP configuration instead of DHCP.
- **Transport Addresses**: The IP addresses, ports and transport protocols that will be used by the system.
  The string format is:
  `<IP address>:<port>/<transport protocol>`
  for example:
  `80.253.154.100:3478/UDP`
  `80.253.154.100:5349/TLS`
  Note: UDP transport protocol is recommended because it is used for real-time communication.
- **External Addresses**: The addresses used by the browser application (Agent Portal Web) to reach the TURN server. This value must be the FQDN used over the internet.

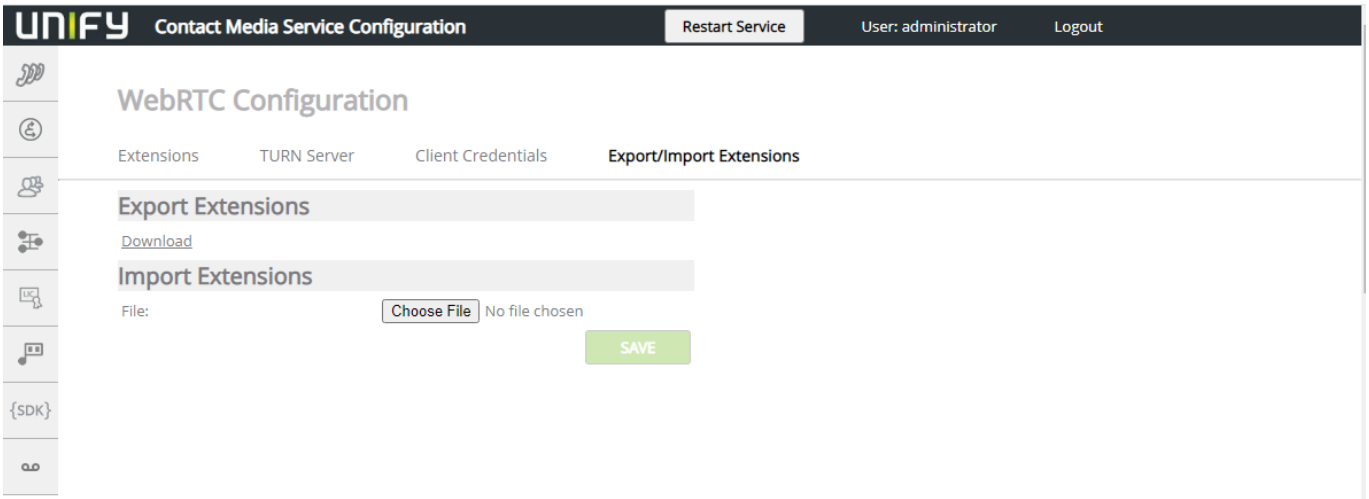


4. Click the **Client Credentials** tab. Here you can define the credentials, which correspond to the same settings of each OSCC Application Server in the **Configuration Center > Agent Portal Web**. The following parameters can be configured:

- **Client id**: The client identification. Default value: default-client-id. This is a default value, and it can be modified.
- **Client secret**: The client secret used for authentication. Default value: default-client-secret. This is a default value, and it can be modified.
- **Token validity time (min)**: Time in minutes, in which the API token will be valid. When the value of the parameter is 0, it means that the time is unlimited. The value must be a positive number and the recommendation is to use a timeout value for security reasons. Use 0 only for special instances. Default value: 20
- Click **SAVE** to save your entries



5. Click the **Export/ Import Extensions** tab. Here you can export/import the Extensions configuration from one OpenScape Contact Media Service server to another OpenScape Contact Media Service server. This is very helpful when multiple OpenScape Contact Media Service servers are used for load balancing, because you can replicate the extension configuration.
   - In the **Export Extensions** area, click **Download** to export the extensions related data to a file
   - In the **Import Extensions** area, click **Choose File** and select the file you want to import
   - Click **Save**

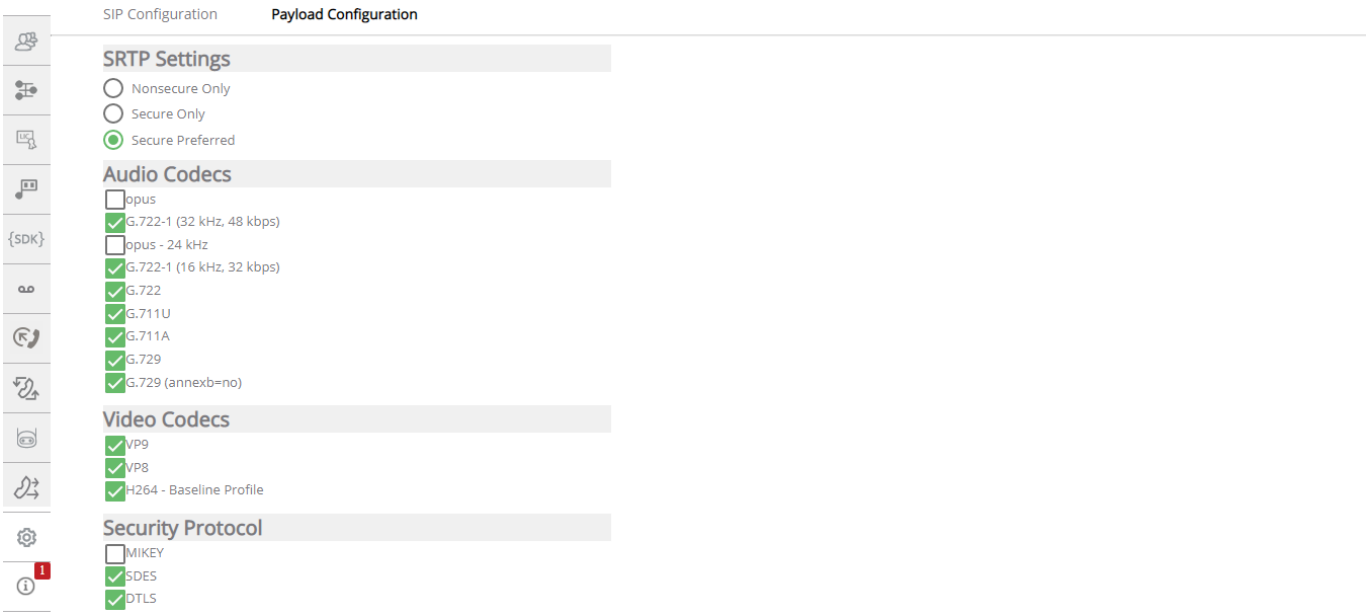


Note: Before using the WebRTC Video and Screen Sharing, you must first configure the WebRTC Integrated Phone and make sure that both features are licensed.

6. Click the VoIP Configuration icon  and then tab Payload Collaboration

For the Integrated Phone the following configuration must be done:

- **SRTP Settings** – Set SRTP to either Secure Only or Secure Preferred. WebRTC must be secure by definition.
- **Audio Codecs** – Enable the codecs “opus”, “opus, 24kHz”, “G.711A” and “G.711U, as they are the most supported codecs by the web browsers.
- **Video Codecs** - Select one or more codecs, “VP9”, “VP8”, “H264 - Baseline Profile”, to enable the WebRTC Video and Screen Sharing feature
- **Security Protocols** – Enable the security protocol DTLS. WebRTC requires SRTP key negotiation via DTLS by definition.



Note: When running the Agent Portal Web on Mozilla Firefox, the following configurations are necessary:

For **WebRTC Integrated Phone:**

1. Open the Mozilla Firefox browser and type about:config on the URL field
2. Search for media.setsinkid.enabled and set it to true
3. Search for security.sandbox.content.level and set it to 2
4. On the OpenScape Contact Media Service web GUI, navigate to **VoIP Configuration > Payload Configuration** and do not select the Opus codec.

For **WebRTC Video and Screen Sharing:**

On the OpenScape Contact Media Service web GUI, navigate to VoIP Configuration > Payload Configuration and do not select only the H264 - Baseline Profile video codec.

Note: Video/Screen Sharing resolutions:

- For Screen Sharing it streams in HD resolution 1280×720.
- The max bandwidth consumed by video in our WebRTC resolution is 500 Kbps (0.5 Mbps) in any CODEC.

5.5 Corporate Web Server

This chapter is applicable only, if the feature "Video/Screen Share" is used and external callers initiate voice contacts to the contact center via the "Click-to-Contact" component.

The Click-to-Contact Component is a Web Component developed with “LitElement” from the “Polymer Project” (https://lit-element.polymer-project.org/guide), which can be embedded in a customer’s web page. The component allows the web page to have the following WebRTC functionalities: Letting users dial to a configured pilot number directly from the web page in the browser. In case the agent is working with the WebRTC Integrated Phone, the component also allows the user to share his screen or his webcam video with the agent, as well as receiving the agent's screen or webcam video.

This chapter describes the implementation and customization of the Click-to-Contact component in a customer’s web page.

5.5.1 Implement the JavaScript module

The JavaScript module “oscc-clicktodial-bundle.js” is provided with the OSCC software in the directory “OpenScape Contact Center Web Components” and subdirectory “Click To Dial”. It needs to be stored on the corporate web server, preferably in a directory where other JavaScript programs for this web application are stored as well. Then, the module needs to be referred in the web page in the section “<head>” by the tag “<script>”:

```
<script type=”module” src=”...path.../oscc-clicktodial-bundle.js”></script>
```

Usually the path is specified as a relative path from the root of the web application.

5.5.2 Implement the Click-to-Contact component

The Click-to-Contact component is represented by the tag “<oscc-click-to-dial>” and the corresponding end tag. It must be entered in section “<body>” of the web page. Further, the attribute “webInteractionUrl” needs to be added to the tag, which needs to refer to the Webinteraction SDK:

```
<oscc-click-to-dial

     webInteractionUrl=”https://...fqdn.../webinteraction sdk”

></oscc-click-to-dial>
```

For more information regarding WebInteraction SDK, see chapter 5.3 Application Server.

5.5.3 Authenticate the Click-to-Contact component (optional)

This step applies, if in the Application Server Configuration Center: Tab “Webinteraction SDK” > Integrated Phone the option “Authorization Enabled” is activated.

The Click-to-Contact component will not authenticate with the WebInteraction SDK by its own, so if the above authorization is active, the component will require a backend service to do the authentication. This authentication expects an OAuth2 response:

```
{

"access_token": "JSON Web Token",

"token_type": "Bearer",

"expires_in": 20,
```

```
"scope": "all"

}
```

In this case the attribute “authorizationUrl” needs to be added to the tag “<oscc-click-to-dial>”, which must be the authentication request returning the answer from the WebInteraction SDK with the proper object above.

```
<oscc-click-to-dial

     webInteractionUrl=”https://...fqdn.../webinteraction
sdk”

     authorizationUrl=”https://...fqdn.../webinteractions
dk/webtrc/authentication”

></oscc-click-to-dial>
```

The "Click-To-Contact" component is highly customizable, allowing a complete change of the component's texts, styles and other relevant properties. For details on customization, see the document "WebInteraction SDK Rest API Framework".

5.6 Load Balancer

In principle, any Load Balancer that supports WebSocket connections can be used in the Integrated Softphone solution.

The Load Balancer can be implemented standalone or in High Availability. In a High Availability environment, the deployment can be Co-located (layer 2) or Geo Separated (layer 3).

The Load Balancer must be provided by the customer as part of the network infrastructure. A commercial Load Balancer (like F5 or Netscaler) or HAProxy (suggested) can be used as part of the solution.

5.6.1 Commercial Load Balancer

If a commercial Load Balancer, as for example F5 or Netscaler, is used to support High Availability for the Application Server and/or for Integrated Softphone, the following requirements must be attended:

- Load Balancing through the Application Servers.
    - Balancing mode suggest to be round robin but this is not mandatory.

    - Health Check of the Application Server nodes can be done via HTTP GET request.
      Note: The response can be 200 or 302.
- Session persistence must be supported - recommended support of Persistence via Cookie.
- Session timeout must be either disabled or long enough to allow that the agent is connected through the shift duration.
- TLS termination - the TLS connection can be terminated in the Load Balancer, but the Application Server requires HTTPS by default.
- Eventually the port can be converted if required - the application server is deployed to receive the HTTPS traffic at port 443.
- NAT can be performed by the Firewall before reaching the Load Balancer.

For the Integrated Softphone, the following additional requirements must be considered:

- For CMS the signaling traffic, namely HTTPS, must only traverse the Load Balancer for URL filtering. The CMS load balancing is performed by the Application Server.
- URL filter for the CMS - to avoid that the configuration web page of the CMS is accessed from the Internet.
  Note: the following URL shall be permitted: <CMS external FQDN>:7443/webkit.
  Note: the URL <CMS external FQDN>:7443/cms shall not be permitted.
  Note: the 7443 is the default port for HTTPS port and it is configurable.
- TLS Termination - the TLS connection to CMS can be terminated in the Load Balancer, but the CMS requires HTTPS by default.
- TURN traffic can be routed via the Load Balancer if it supports UDP. Note: it must be verified if the used Load Balancer supports the required UDP traffic.
- NAT for the CMS can be performed by the Firewall before reaching the Load Balancer.

Note - if the switchover between the data centers is mandatory, the Load Balancer must monitor the status of the OSCC and OSV to switchover the whole data center if either the OSCC or the OSV fails. This scenario was not tested and it is not covered by the HAProxy Whitepaper.

5.6.2 HAProxy

HAProxy is an open source load balancer for HTTP based solutions. HAProxy is part of the OpenSUSE Linux distribution that must be obtained by the customer (not provided with OSCC software). The HAProxy is mainly used to load balance the access to the Application Server for the Agent Portal Web. The HAProxy can also be used to protect the access from the Internet to the configuration portal of the OpenScape Contact Media Service server.

For HAProxy, please refer to the HAProxy Deployment Recommendation Whitepaper.

6. Diagnostic Data

When using the WebRTC functionality in Open Scape Contact Center multiple servers and components are involved. In case of disfunction, diagnostic data of those components is needed to diagnose and solve the problem by Global Vendor Support and Development. This chapter provides an overview of which diagnostic data is required, how to set log levels and where to collect the log files from.

As the WebRTC production environment can differ largely from customer to customer, it is recommended to provide a topology diagram of the resources, which may be involved: Contact Center Server(s), Application Server(s), Contact Media Service Server(s), Load Balancer (e.g. HAProxy), DMZ, Firewalls, DNS (FQDN).

6.1 Agent Portal Web

Diagnostic data can be obtained from the Agent Portal Web application, which runs in the web browser on the agent's client machine.

In the header bar of the Agent Portal Web application (the bar where the agent's avatar and name are shown), press <Ctrl><Shift> with left mouse click. The button "Diagnosis" appears in the bar, click on it and check in the subsequent window the boxes "Enable file logging" and "Enable console output". Enable all "Diagnosis Levels" Verbose, Debug, Info, Warning and Error. Choose the maximum log file size and the maximum number of log files appropriately and click "Apply changes". The configured settings take immediately effect and are saved for a next Agent Portal Web session.

In this same window where the diagnostics are configured, the corresponding log files for the current session can be downloaded by clicking on "Save current state log". The logs files are named "AgentPortal_Client.log" (extended by a date/time stamp) and are written to the selected download directory.

6.2 Contact Media Service

Diagnostic data can be obtained from the Contact Media Service Server from three resources: From the Contact Media Service itself, from the Media Server being part of the Contact Media Service and from the Network on the Linux level.

6.2.1 Contact Media Service

In the Contact Media Service Configuration web application click in the left-side main menu on the button “System” and then on the tab “Diagnostics”, select the diagnostic level “Debug”. Choose the maximum number of log files appropriately. The configured and saved settings take immediately effect. The logs files are named “cms.log” (extended by a sequence number) and are written to the directory “/opt/cms/log” with size of ca. 25 MB each.

In the same tab where the diagnostics are set, the log files can be downloaded via the link “Download Diagnostics”. The complete set of log files contained in the directory mentioned above will be compressed and will be available as “cms_log.tar.gz” file (extended by a date/time stamp).

After a problematic scenario has been traced and log files saved, it is recommended for performance reasons to reconfigure the log level back to the original lower level.

6.2.2 Media Server

Sometimes the root cause of a problem is not located in the Contact Media Service itself, but in the integrated Media Server. Media Server uses the Java-based “Log4j” logging utility. For details about this utility, see: https://www.tutorialspoint.com/log4j/index.htm

Note: Apply the log level changes as described below only, when instructed by Global Vendor Support or Development.

Log levels can be configured for various Media Server components. This is done in the file “/opt/Core/application_host/bin/log4j.xml”. This file can be edited, for example with Linux text editor “vi” or “vim”. The log level “FINE” is required for the components “connectivity.sip”, “connectivity.sip.messages” and “media.streaming”:

```
<logger name="com.cycos.connectivity.sip">
<level value="FINE"
class="com.cycos.media.logging.log4j.ExtendedLevel" />
</logger>

<logger name="com.cycos.connectivity.sip.messages">
<level value="FINE"
class="com.cycos.media.logging.log4j.ExtendedLevel" />
</logger>
```

```
<logger name="com.cycos.media.streaming">
<level value="FINE"
class="com.cycos.media.logging.log4j.ExtendedLevel" />
</logger>
```

Note: Do not change the log level for the other components listed, unless instructed otherwise. When doing the log level configuration in this file, take care to change only the “value” attribute and not any other attribute or the XML syntax.

The configured and saved settings take immediately effect. The logs files are named “mediaserver.log” (extended by a sequence number) and are written to the directory “/opt/cms/log” with size of ca. 25 MB each.

As these log files are in the same directory as the log files of Contact Media Service, these and the other log files can be compressed and downloaded via the Contact Media Service Configuration web application, as previously explained.

After a problematic scenario has been traced and log files saved, it is recommended for performance reasons to reconfigure the log levels back to the original lower levels.

6.2.3 Network

A Wireshark network trace can be captured with the Linux utility “tshark”. For details about this utility, see:

https://www.wireshark.org/docs/man-pages/tshark.html

Run the utility via the Secure Shell “ssh” command with root account. The utility must be applied on the appropriate network interface, usually or by default “eth0”, for example:

```
tshark -i eth0 -b filesize:50000 -w
/opt/cms/log/tshark.pcap
```

To stop the capture of the trace press <Ctrl> and C.

In the example above the trace files are named “tshark.pcap” (extended by a file number and date) and are written to the directory “/opt/cms/log” with size of ca. 50000 kB (50 MB) each. If the output directory is chosen as mentioned above, then these and the other log files in this directory can be compressed and downloaded via the Contact Media Service Configuration web application, as previously explained.

6.3 Application Server

Diagnostic data can be obtained from the Application Server. The log files affect the Agent Portal Web application and the Apache Tomcat webserver, which latter is installed as part of the Application Server.

6.3.1 Agent Portal Web

In the Application Server Configuration Center web application click on the tab “Agent Portal Web” and then menu “General”. For “Log Level” select “DEBUG” and save the setting. The configured and saved setting takes immediately effect.

Note: The configurations done in this tab are stored in the file “<installdir>\ApplicationServer\ApacheWebServer\conf\webagent.xml”.

The logs files are named “webagent.log” (extended by a sequence number) and are written to the directory “<installdir>\ApplicationServer\ApacheWebServer\logs” with size of ca. 5 MB each.

After a problematic scenario has been traced and log files saved, it is recommended for performance reasons to reconfigure the log level back to the original lower level.

6.3.2 Apache Tomcat

For certain problematic scenarios involving the Apache Tomcat webserver the so-called “catalina” logs may be required. Catalina is a core component of Apache Tomcat, which is involved in starting up the webserver and the services of the Application Server. Log levels can be configured in the file “<installdir>\ApplicationServer\ApacheWebServer\conf\logging.properties”. The logs files are named “catalina.log” (extended by a date/time stamp) and are written to the directory “<installdir>\ApplicationServer\ApacheWebServer\logs” with various sizes.

6.4 Contact Center Server

Different diagnostic data can be obtained from the Contact Center Server, of which the Telephony Server log file, covering CSTA traffic, is most important in WebRTC scenarios. The log level is configured with the System Monitor application and should be set to at least “INFO” level, preferably full level. The configured setting takes immediately effect.

The logs files are named “TelephonyServer_<servername>” (extended by a sequence number) and are written to the directory “<installdir>\ ShareData\DIAGS” with size of ca. 5 MB each.

After a problematic scenario has been traced and log files saved, it is recommended for performance reasons to reconfigure the log level back to the original lower level.

In addition to these log files the Contact Center configuration is required. It can be obtained by exporting the configuration from the production database to a design database.

References

1. *OpenScape Contact Media Service V11 R1, Installation Guide*
2. *OpenScape Contact Center Enterprise V11 R1, Overview Guide, Description*
3. *OpenScape Contact Center Agile/Enterprise V11 R1, Security Checklist, Security Checklist*
4. *OpenScape Contact Center Enterprise V11 R1, Agent Portal Web, User Guide*
5. *OpenScape Solution Set V10, Certificate Management and Transport Layer Security (TLS), Administrator Documentation*
6. *OpenScape Contact Center Enterprise V11 R1 Web Interaction REST API Framework, Programming Guide*
7. *OpenScape Contact Center HAProxy Configuration, Whitepaper*