Vocera Messaging Platform API Guide

Version 5.2.3

Notice

Copyright © 2002-2018 Vocera Communications, Inc. All rights reserved.

Vocera® is a registered trademark of Vocera Communications, Inc.

This software is licensed, not sold, by Vocera Communications, Inc. ("Vocera"). The reference text of the license governing this software can be found at **http://www.vocera.com/legal/**. The version legally binding on you (which includes limitations of warranty, limitations of remedy and liability, and other provisions) is as agreed between Vocera and the reseller from whom your system was acquired and is available from that reseller.

Certain portions of Vocera's product are derived from software licensed by the third parties as described at **http://www.vocera.com/legal/**.

Microsoft®, Windows®, Windows Server®, Internet Explorer®, Excel®, and Active Directory® are registered trademarks of Microsoft Corporation in the United States and other countries.

Java® is a registered trademark of Oracle Corporation and/or its affiliates.

All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owner/s. All other brands and/or product names are the trademarks (or registered trademarks) and property of their respective owner/s.

Vocera Communications, Inc.
www.vocera.com
tel :: +1 408 882 5100
fax :: +1 408 882 5101

Last modified: 2018-11-27 08:46

VMP-523-Docs build 188

Contents

The Vocera Messaging Platform API

The Vocera Messaging Platform (VMP) API provides a mechanism to submit and track the status of alerts. This API contains methods, data classes, and schemas.

Note: The VMP API uses the SOAP connector. By default, the VMP license is not enabled to use this connector. See the licensing information in the *Vocera Messaging Platform Administration Guide* for more details.

Concepts

Two Status Change update modes are available: Polling Mode and Callback Mode.

In Polling Mode, the calling application needs to send a paging alert and scan the delivery status by the user's request or according to a time interval.

In Callback Mode, the calling application needs to receive delivery notifications in near real-time mode.

These two modes are compatible and can be used together.

Entry Points

The VMP API provides a URL that you can use to display the API in WSDL format.

If you have configured your VMP Server to use SSL, this API is `https://address/wic.asmx?WSDL`. Replace `address` with the IP address or domain name of your VMP Server (or of your load balancer, if you are using VMP in a high-availability environment).

Note: Refer to the *Vocera Messaging Platform Administration Guide* for more information on SSL configuration and using VMP in a high-availability environment.

If your VMP Server does not use SSL (not recommended), this API is `http://address/wic.asmx?WSDL`.

To send a SOAP request to the VMP Server, use one of the following URLs, depending on whether your server is configured to use SSL:

- `https://address/wic.asmx`
- `http://address/wic.asmx`

As before, replace `address` with the IP address or domain name of your of your VMP Server or load balancer.

Methods

The VMP API defines these methods for processing paging alerts.

- Paging_SendAlert on page 5

Paging_SendAlert

This method sends the alert identified by an external system ID.

If a user is not found in the VMP Server, the return array contains the appropriate `PagingAlertUserStatusInfo` with `ERROR` status and `User unknown` detail text.

Table 1: Paging_SendAlert method parameters

Parameter	Type	Description
guid	String	A reference to the VMP instance served by the VMP Paging gateway. The default is MAIN.
user	String	The user login (configured in VMP Administrator).
password	String	The user password (configured in VMP Administrator).
message	PagingAlert	The alert message with all options.
responses	PagingAlertResponse[]	The array of responses.
users	PagingAlertUserRef[]	The array of recipients. Supported references are Pager ID, Vocera ID, and AD Username.
dls	String[]	The array of distribution list ids.
callbackInfo	PagingAlertStatusCallbackInfo	The status notification callback descriptor. The VMP Server supports a callback failover mechanism. It is implemented as a callback notification queue. Status change notifications will be kept in the queue for 24 hours. If the VMP gateway cannot successfully submit a notification during the 24 hour period, the notification is discarded. Gateway clients can use the `Paging_GetAlert` status call to poll the alert status and update their database.

Return type: `WWI.WIC.WWS. PagingAlertUserStatusInfo[]`

Paging_GetAlertStatus

This method gets the status of the Alert identified by an external system ID.

If a user is not found in the VMP Server, the return array contains the appropriate `PagingAlertUserStatusInfo` with `ERROR` status and `User unknown` detail text.

Table 2: Paging_GetAlertStatus method parameters

Parameter	Type	Description
guid	String	A reference to the VMP instance served by the VMP Paging gateway. The default is MAIN.

Parameter	Type	Description
user	String	The user login (configured in VMP Administrator).
password	String	The user password (configured in VMP Administrator).
alertExternalID	String	The shared alert ID submitted in `Paging_SendAlert`.

Return type: `WWI.WIC.WWS. PagingAlertUserStatusInfo[]`

Paging_ResendAlert

This method resends the alert referenced by `AlertExternalID`. The initial alert is deleted from the VMP database.

Table 3: Paging_ResendAlert method parameters

Parameter	Type	Description
guid	String	A reference to the VMP instance served by the VMP Paging gateway. The default is MAIN.
user	String	The user login (configured in VMP Administrator).
password	String	The user password (configured in VMP Administrator).
alertExternalID	String	The shared alert ID submitted in `Paging_SendAlert`.
message	PagingAlert	The alert message with all options.
responses	PagingAlertResponse[]	The array of responses.
callbackInfo	PagingAlertStatusCallbackInfo	The status notification callback descriptor. The VMP Server supports a callback failover mechanism. It is implemented as a callback notification queue. Status change notifications will be kept in the queue for 24 hours. If the VMP gateway cannot successfully submit a notification during the 24 hour period, the notification is discarded. Gateway clients can use the `Paging_GetAlert` status call to poll the alert status and update their internal database.
resendToCurrentRecipientsOnly	Boolean	This flag indicates whether VMP should repeat the process of VMP user identification using the shared ids in the original paging alert. If `ResendToCurrent RecipientsOnly` is false, messages are sent to the same users as the original message.

Return type: `WWI.WIC.WWS. PagingAlertUserStatusInfo[]`

Paging_DeleteAlert

This method deletes a paging alert from the recipients referenced by the Users parameter.

The method returns the alert status for all users to whom this alert was initially sent. For users for which the alert is being deleted, the status flag is `CANCELED`.

Table 4: Paging_DeleteAlert method parameters

Parameter	Type	Description
guid	String	A reference to the VMP instance served by the VMP Paging gateway. The default is MAIN.
user	String	The user login (configured in VMP Administrator).
password	String	The user password (configured in VMP Administrator).
alertExternalID	String	The shared alert ID submitted in `Paging_SendAlert`.
users	PagingAlertUserRef[]	The array of recipients for which the alert should be deleted.

Return type: `WWI.WIC.WWS. PagingAlertUserStatusInfo[]`

Paging_DeleteAlertAll

This method deletes the paging alert from all users to which it was initially sent.

The method returns the alert status for all users to which this alert was initially sent. The status flag is `CANCELED` for all users.

Table 5: Paging_DeleteAlertAll method parameters

Parameter	Type	Description
guid	String	A reference to the VMP instance served by the VMP Paging gateway. The default is MAIN.
user	String	The user login (configured in VMP Administrator).
password	String	The user password (configured in VMP Administrator).
alertExternalID	String	The shared alert ID submitted in `Paging_SendAlert`.

Return type: `WWI.WIC.WWS. PagingAlertUserStatusInfo[]`

Paging_CancelAlertByEventID

This method cancels the group of paging alerts from all users to which it was initially sent.

The method returns the total number of users the alert was canceled from and a list of those users. The status flag is `CANCELED` for all users.

Table 6: Paging_CancelAlertByEventID method parameters

Parameter	Type	Description
guid	String	A reference to the VMP instance served by the VMP Paging gateway. The default is MAIN.
user	String	The user login (configured in VMP Administrator).
password	String	The user password (configured in VMP Administrator).
eventID	String	The original event ID submitted in `Paging_SendAlert`.

Parameter	Type	Description
endPoint	Integer	The end point type submitted in Paging_SendAlert`Paging_SendAlert`. VMP cancels only alerts which match both `EventID` and `EndPoint`.

Return type: `WWI.WIC.WWS. PagingAlertUserStatusInfo[]`

WebComposeAlert_UserExists

This method checks if a user with a specified email, pager ID or device ID exists on the VMP Server and can be used as a recipient in the compose alert URL.

Table 7: WebComposeAlert_UserExists method parameters

Parameter	Type	Description
id	String	The user email address/pager ID/ device ID.
sid	String	A reference to the VMP instance served by the VMP Paging gateway. The default is MAIN.
user	String	The user login (configured in VMP Administrator).
password	String	The user password (configured in VMP Administrator).

Return type: `Boolean`

Data Classes

The VMP API methods use these data classes.

- PagingAlert
- PagingAlertUserRef
- PagingAlertResponse
- PagingAlertStatusCallbackInfo
- PagingAlertUserStatusInfo

PagingAlert

The PagingAlert data class contains these fields.

Field	Type	Description
Subject	String[256]	The alert subject.
Body	String [3000]	The alert message.

Field	Type	Description
ExternalID	String[32]	The ID of the submitted alert. VMP does not require an external ID to accept incoming SOAP messages. If SOAP messages include an external ID, it must be a unique string that is used for all incoming SOAP messages. This string can be any combination of non-Unicode symbols, including special characters. All third-party SOAP integrations share the same SQL database table for external IDs. This means that each external ID must be unique across all SOAP integrations.
OverridePersonalAlarmSettings	Boolean	• True: sound and led notification settings defined on the server are used. • False: sound and led notification settings defined in user's device profile are used.
Severity	Integer	• 0: urgent if OverridePersonal Settings is True, high otherwise • 1: normal • 2: normal
Callback	String[32]	Callback phone number.
EventID	String[40]	Event ID that the alarm is associated with.
BriefBody	String[256]	Short version of a message body. If it is not empty, it is sent to a Vocera badge instead of the full-length **Body** field.
PatientMRN	String[70]	MRN of the patient that the alarm is associated with.
Wavefilename	String[40]	The wavefile name that the alarm is associated with.
ParticipationConditions	PagingAlertParticipationCondition	By default, this is ALL. If it is set to ONLINE_ONLY, the message will be queued and delivered to only Voice logged in users.

PagingAlertUserRef

The PagingAlertUserRef data class contains these fields.

Field	Type	Description
PagerID	String[64]	The user's pager ID.
VoceraID	String[64]	The user's Vocera ID.
ADUsername	String[64]	The user's AD username. If the AD username contains a subdomain, VMP matches subdomain and username. If only the username is provided, VMP searches for the user with same username in all domains.

PagingAlertResponse

The PagingAlertResponse data class contains these fields.

Table 8: PagingAlertResponse fields

Field	Type	Description
ExternalID	String[32]	The ID of the response. VMP does not require an external ID to accept incoming SOAP messages. If SOAP messages include an external ID, it must be a unique string that is used for all incoming SOAP messages. This string can be any combination of non-Unicode symbols, including special characters. All third-party SOAP integrations share the same SQL database table for external IDs. This means that each external ID must be unique across all SOAP integrations.
Body	String[256]	The text of the response.

PagingAlertStatusCallbackInfo

The PagingAlertStatusCallbackInfo data class contains these fields.

Field	Type	Description
Protocol	PagingAlertStatusCallbackProtocol	• EMAIL • HTTP • WCTP
Format	PagingAlertStatusCallbackFormat	If the WIC_GENERIC format is used, the VMP gateway returns the `PagingAlertUserStatusInfo` object. The XSD schema may be found in this document. A WIC_GENERIC compatible callback should respond with the `QueryResult` class. The XSD is also provided in document below.
Address	String[256]	The callback address (according to protocol), with one of the following possible values: • EMAIL – an email address • HTTP – a URL • WCTP – a WCTP address
Options	PagingAlertStatusCallbackOptions	Custom callback options. Reserved for future use.

PagingAlertUserStatusInfo

The PagingAlertUserStatusInfo data class contains these fields.

Field	Type	Description
AlertExternalID	String	Alert ID submitted in `Paging_SendAlert`.
User	PagingAlertUserRef	User reference.
ResponseExternalID	String	Response ID.

Field	Type	Description
Status	PagingAlertStatus	• ERROR_USER_DEVICE_DISABLED • ERROR_UNKNOWN_USER • ERROR • QUEUED • SENT • FAILED • DELIVERED_TO_DEVICE • OPENED • RESPONDED • EXPIRED • CANCELED • CALLBACK_INITIATED
StatusDetails	String	The text description of the status.
StatusChangeDatetime	DateTime	The time when the status was changed to its current status. The time is provided in the GMT time zone.

PagingAlertUserRef XSD

This is XML for the PagingAlertUserRef XSD.

```
<s:complexType name="PagingAlertUserRef">
        <s:sequence>
           <s:element minOccurs="0" maxOccurs="1" name="PagerID"
type="s:string"/>
           <s:element minOccurs="0" maxOccurs="1" name="VoceraID"
type="s:string"/>
           <s:element minOccurs="0" maxOccurs="1" name="ADUsername"
type="s:string"/>
        </s:sequence>
     </s:complexType>
```

PagingAlertUserStatusInfo XSD

This XSD may be used to automatically generate a class that can be used for VMP gateway request de-serialization.

```
<s:complexType name="PagingAlertUserStatusInfo">
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="AlertExternalID"
type="s:string"/>
        <s:element minOccurs="0" maxOccurs="1" name="User"
type="tns:PagingAlertUserRef"/>
        <s:element minOccurs="0" maxOccurs="1" name="ResponseExternalID"
type="s:string"/>
        <s:element minOccurs="1" maxOccurs="1" name="Status"
type="tns:PagingAlertStatus"/>
        <s:element minOccurs="0" maxOccurs="1" name="StatusDetails"
type="s:string"/>
        <s:element minOccurs="1" maxOccurs="1" name="StatusChangeTime"
type="s:dateTime"/>
        <s:element minOccurs="0" maxOccurs="1" name="ResponseComment"
type="s:string"/>
        <s:element minOccurs="0" maxOccurs="1" name="TextResponse"
type="s:string"/>
      </s:sequence>
   </s:complexType>
```

PagingAlertUserStatus XSD

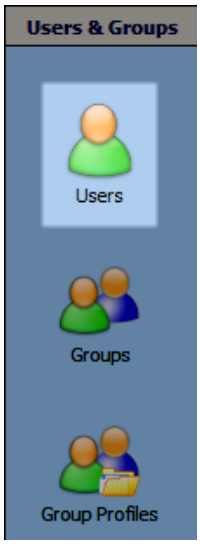
```
<s:simpleType name="PagingAlertStatus">
      <s:restriction base="s:string">
        <s:enumeration value="ERROR_USER_DEVICE_DISABLED"/>
```

```
        <s:enumeration value="ERROR_UNKNOWN_USER"/>
        <s:enumeration value="ERROR"/>
        <s:enumeration value="QUEUED"/>
        <s:enumeration value="SENT"/>
        <s:enumeration value="FAILED"/>
        <s:enumeration value="DELIVERED_TO_DEVICE"/>
        <s:enumeration value="OPENED"/>
        <s:enumeration value="RESPONDED"/>
        <s:enumeration value="EXPIRED"/>
        <s:enumeration value="CANCELED"/>
        <s:enumeration value="CALLBACK_INITIATED"/>
      </s:restriction>
    </s:simpleType>
```

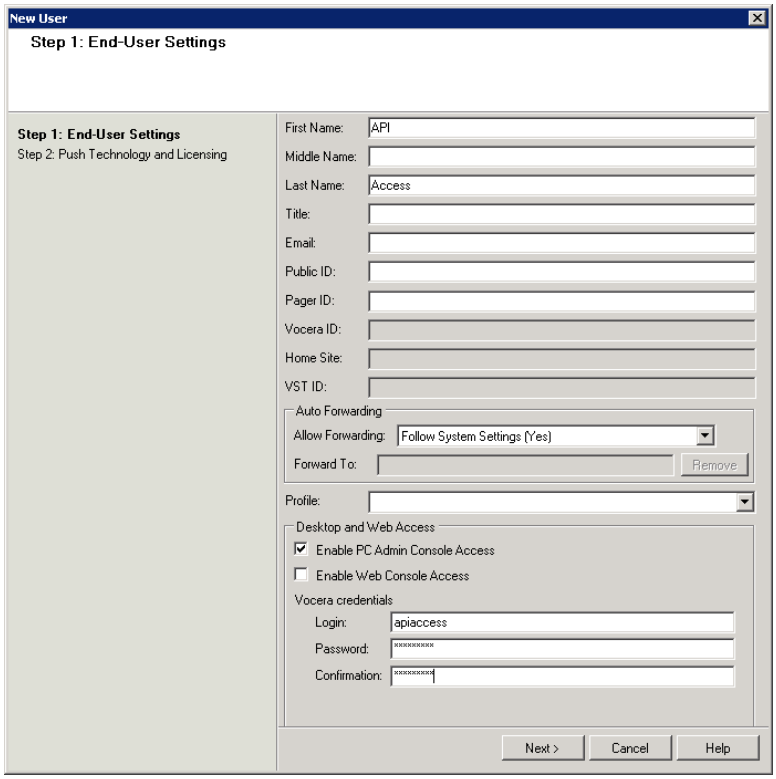
Configuring API Access for a Third-Party System

If you want a third-party system to be able to access VMP, you must create a VMP user and configure it for API access.

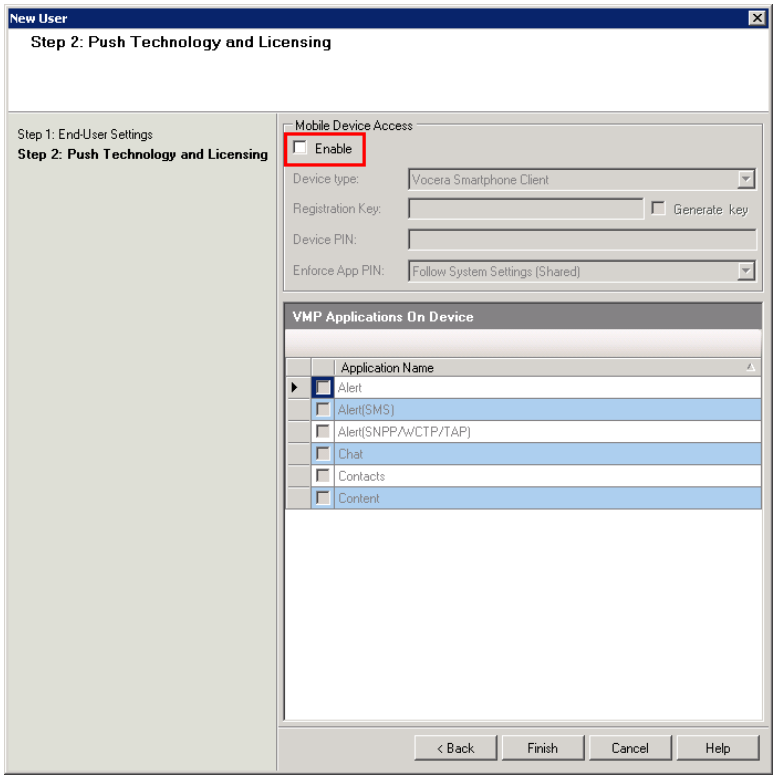
1. Start the VMP Administrator application:
   All Programs > VMP > VMP Administrator
2. Type **admin** (or your administrative credentials) in the VMP Login dialog, and click OK.
3. From the VMP Administrator, select Users & Groups > Users .



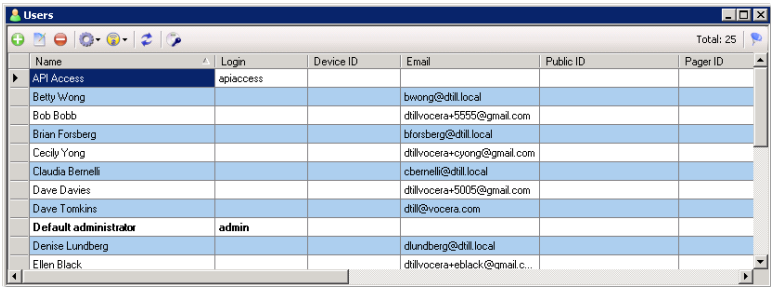
4. In the toolbar in the Users pane, click Add . The End-User Settings window appears.



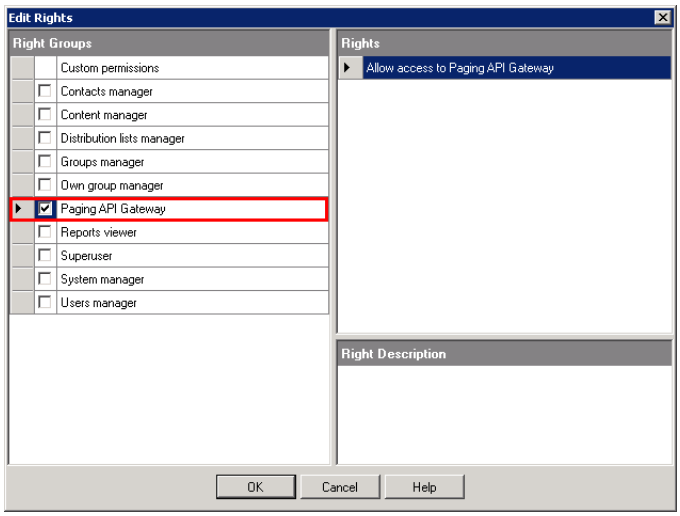
5. In the First Name and Last Name fields, type a first name and last name. The actual names aren't important.
6. Select the Enable PC Admin Console Access checkbox.
7. In the Vocera credentials section, in the Login and Password fields, specify a login name and password. This is the information that the third-party system needs to know to be able to connect to the VMP Server.
8. In the Confirmation field, retype the password that you specified in the Password field.
9. Click Next. The Push Technology and Licensing screen appears.
10. In the Mobile Device Access section, clear the Enable checkbox.



11. Click Finish.
12. In the Users pane, ensure that the user that you have just created is highlighted.



13. In the toolbar, from the User preferences dropdown list, select User rights. The Edit Rights dialog box appears.
14. In the Right Groups pane, select Paging API Gateway.

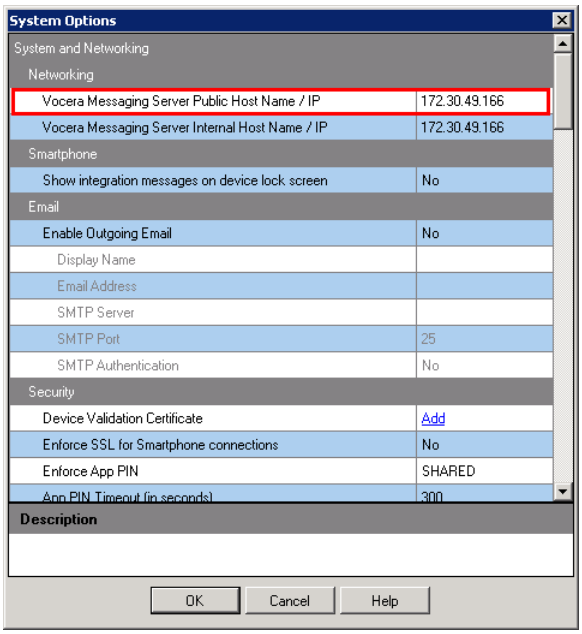


Click OK. The paging API user is now set up and is ready for the third-party system to use.

Note: To access the VMP Server, the third-party system needs the following information:

- The login name and password of the VMP user that you have just created.
- The public hostname or IP address of the VMP Server.

To locate the public hostname or IP address of the VMP Server, select Configuration > System Options. In the System Options window that appears, the public hostname or IP address is displayed in the Vocera Messaging Server Public Host Name / IP field.



The VCS and VMP Web Console API

This URL interface enables third-party device applications running on the same device as the Vocera Collaboration Suite to perform specified tasks within it. This interface also enables applications running on web browsers to perform tasks within the VMP Web Console.

The URL API Interface

The URL format to access the API interface depends on the application that you are accessing.

- Vocera Collaboration Suite: *voceracs://invoke?parameter1=value1&parameter2=value2...*
- VMP Web Console: *http://server/Invoke.aspx?parameter1=value1&parameter2=value2...* (replace *server* with the IP address or domain name of the VMP Web Console)

The parameters that can be provided with this URL are listed in the table below. Some parameters, as noted, can be supplied to the Vocera Collaboration Suite only.

Table 9: The URL API Interface

Parameter	Description	Supported Values
commMode	The Vocera Collaboration Suite application command to be performed.	• Call - Places a normal Vocera call. • UrgentCall - Places an urgent Vocera call. • Alert - Displays the New Alert screen. The recipient specified in recipientKey is specified as the recipient of the Alert. • Chat - Displays a Chat session. If this parameter is not specified, the recipient's Contacts screen is displayed.
recipientKey	The ID of the recipient. This could be an email address, a phone number, or a Vocera ID, depending on the value of the searchScope parameter.	No values provided
searchScope	The contact or user field that should be searched when trying to find the recipient. The search criterion is specified by the recipientKey parameter.	• email - Search for an email address. • voceraID - Search for a Vocera ID as defined on the VMP Server. • adID - Search for an Active Directory user account name. • userPublicID - Search for a Public ID as defined for a user on the VMP Server. • dlPublicID - Search for a Public ID as defined for a Distribution List on the VMP Server.

Parameter	Description	Supported Values
phoneNumber	The phone number to use to place a SIP call through Vocera. If recipientKey and phoneNumber are both supplied, and commMode is Call or UrgentCall, the application uses recipientKey to try to find the recipient. If the recipient is not found using recipientKey, phoneNumber is used to place the call.	No values provided
callBackURL	The URL to bring the user back to the calling application. If callBackURL is specified, a Return icon is added to the Menu screen in place of the Help icon. Tap this icon to invoke the URL specified in callBackURL. The default name associated with this icon is Return. If the callBackLabel parameter is specified, its value becomes the name associated with the icon. This parameter is available for the Vocera Collaboration Suite only.	No values provided
callBackLabel	If callBackURL has been specified, the callBackLabel parameter specifies the name associated with the Return icon. This parameter is available for the Vocera Collaboration Suite only.	No values provided

Here is an example of a URL that can be sent using this interface:

```
voceracs://invoke?
commMode=Call&recipientKey=ggoodman@vocera.com&searchScope=email
```

This searches for the email address ggoodman@vocera.com on the Vocera Collaboration Suite application. If this email address is found, a Call is placed to the user with this email address.