



Facebook Messenger Integration Example

June 2022



Table of Contents

| | |
|--|-----------|
| 1 Integration Overview | 2 |
| 2 Facebook Configuration | 3 |
| 3 Adding Facebook Channel | 8 |
| 4 Securing Your Facebook Connection | 11 |
| 5 Troubleshooting and Tips | 13 |
| 5.1 Facebook Certificate..... | 13 |
| 5.2 Testing the Interface..... | 13 |
| 5.3 Facebook Shortcut..... | 13 |
| 5.4 Facebook ID vs. Facebook Name..... | 14 |
| 5.4.1 Display Name (Facebook name)..... | 14 |
| 5.5 Issues and Log Reading..... | 15 |



1 Integration Overview

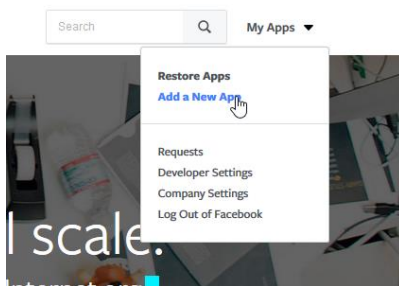
The integration utilizes three different components: SMS Server, Facebook Messenger, and Facebook Webhook service. The purpose of the Webhook service is to monitor Messenger account. If a new chat is initialized, the Webhook delivers the chat to a preconfigured destination. SMS Server configuration is carried out by:

- Operations team in Sinch Contact Pro cloud
- Customer/partner in Sinch Contact Pro on-premise

NOTE NOTE: This configuration is an example so the look and feel of the Facebook pages may change from pictures below. For the latest information, check Facebook documentation.

2 Facebook Configuration

1. Open <https://developers.facebook.com> if you have a developer account already - log in, otherwise create or promote an existing FB account by clicking the 'Log In' button
2. Once the Developer account is created you can move on to create a new application.



3. Insert details for your new application

Create a New App ID

Get started integrating Facebook into your app or website

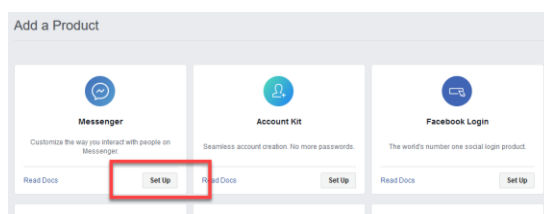
Display Name

Contact Email

By proceeding, you agree to the [Facebook Platform Policies](#)

4. Assign new products to your application.

Navigate to the 'Add a Product' section and look for Messenger then press Set Up.



5. New application with Messenger product is created.

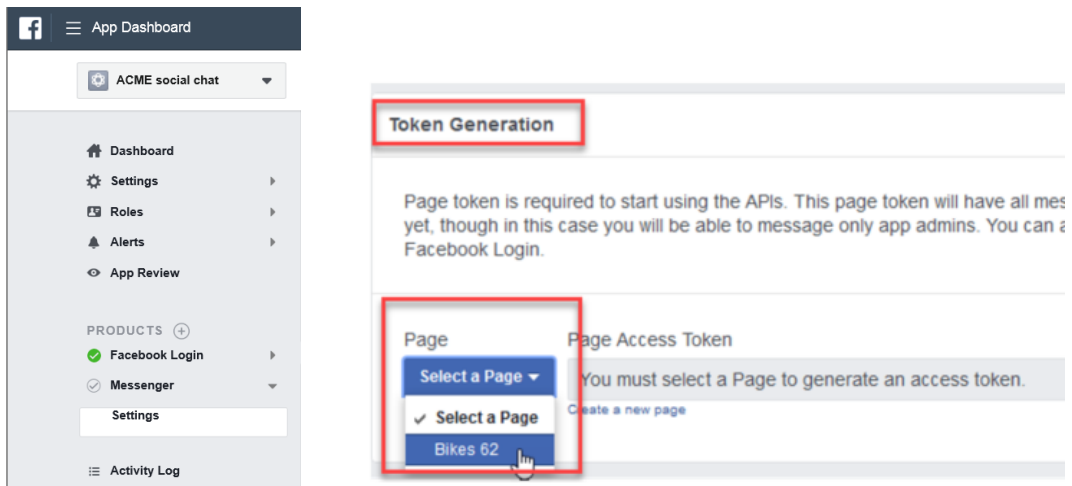
Now the administrator needs to connect this application to a Facebook page. If you don't have one already, create new one at

<https://www.facebook.com/pages/create>.

In this example, we use page called *Bikes 62*.

6. Now assign the Facebook page with *Messenger*.

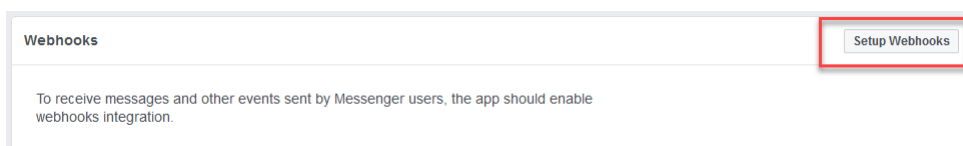
From the App Dashboard page > Select from your products (eg Messenger) > Settings > Token Generation > Your FB page (eg Bikes 62)



Popup will be opened with a request to give permission. Accept these.

7. Enable the Webhooks service

Scroll down the page to the Webhooks section and select 'Setup Webhooks'





8. New Page Subscription configuration

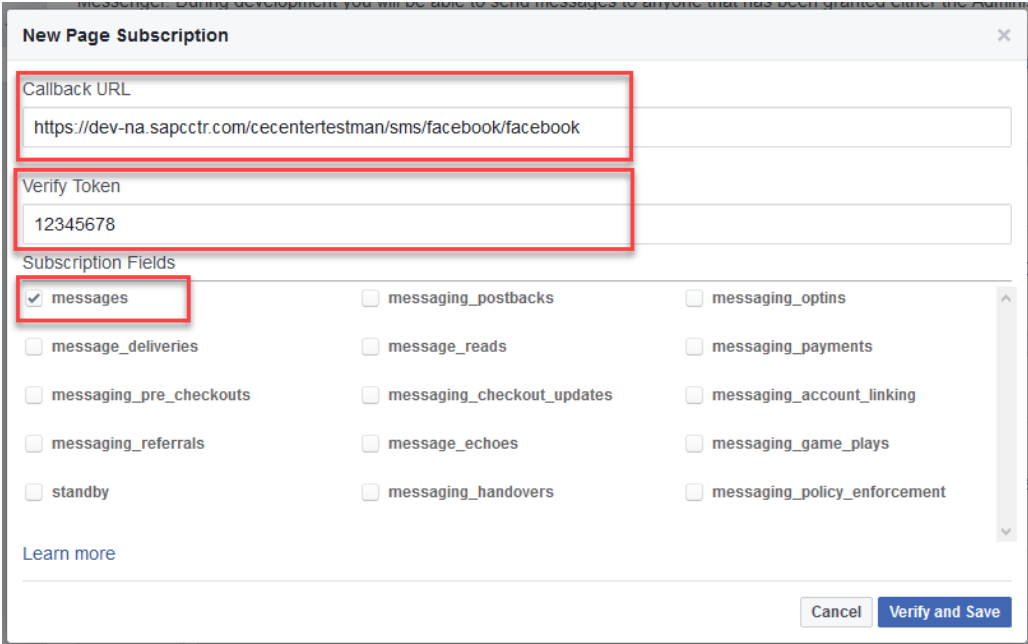
1. Insert a Callback URL that points to your CCenter SMS server.

Typically `http[s]://smsserverlocalhttpserverip:port/sms/facebook/facebook`

Contact Center 365 customer gets this address from DevOps

2. A 'Verify Token' is required, although not used by CCenter a random string can be added, for example 12345678.
3. From the 'Subscription Field' section check: messages.
4. Once done, click Verify and Save. If Facebook fails connecting to SMS server, the page will let you know. –See troubleshooting section in case an error message is returned– If everything goes ok you should be able to see Webhooks icon turning to green on the left pane.

 Webhooks



New Page Subscription

Callback URL

Verify Token

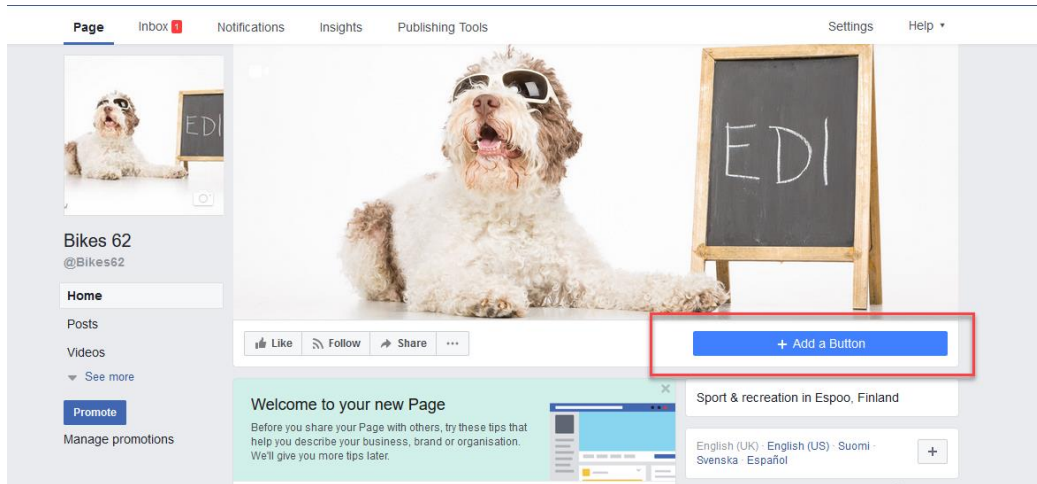
Subscription Fields

| | | |
|--|---|---|
| <input checked="" type="checkbox"/> messages | <input type="checkbox"/> messaging_postbacks | <input type="checkbox"/> messaging_optins |
| <input type="checkbox"/> message_deliveries | <input type="checkbox"/> message_reads | <input type="checkbox"/> messaging_payments |
| <input type="checkbox"/> messaging_pre_checkouts | <input type="checkbox"/> messaging_checkout_updates | <input type="checkbox"/> messaging_account_linking |
| <input type="checkbox"/> messaging_referrals | <input type="checkbox"/> message_echoes | <input type="checkbox"/> messaging_game_plays |
| <input type="checkbox"/> standby | <input type="checkbox"/> messaging_handovers | <input type="checkbox"/> messaging_policy_enforcement |

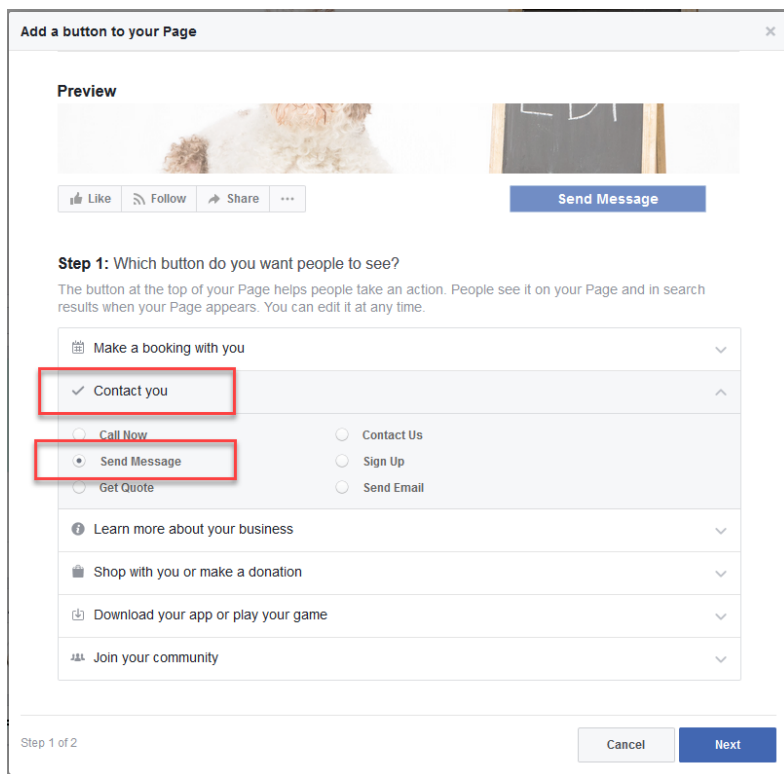
[Learn more](#)



- Next, a send message button should be created to your Facebook site. Open the page and click Add Button.

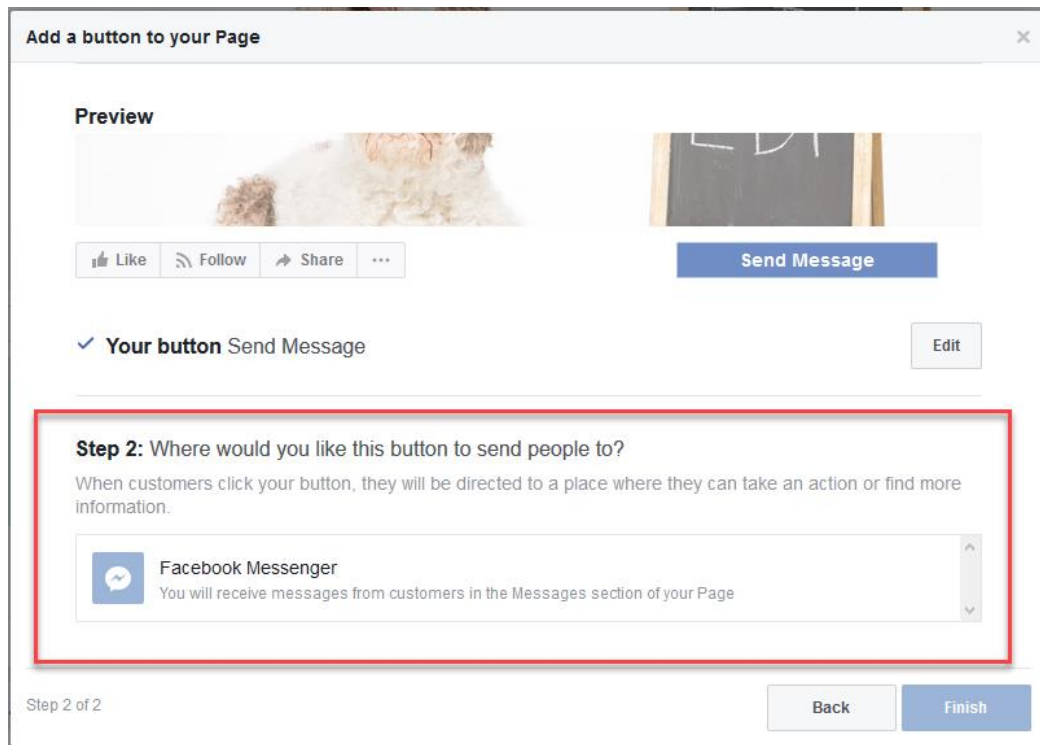


- Select Contact You tab and Send Message, then Next.'





11. Select Facebook Messenger and Finish



You have now finished the Facebook configuration.



3 Adding Facebook Channel

From the System Configurator, the administrator should also configure the Service Connection to Facebook and define the User credentials for associated Facebook account.

1. Open SC – System Management – Messaging Services
2. Add a new service connection by clicking Add New then select Facebook.
3. Provide a name and URL. *https://graph.facebook.com/me/messages*

NOTE The version in the URL above may change. Always use the latest provided by Facebook.

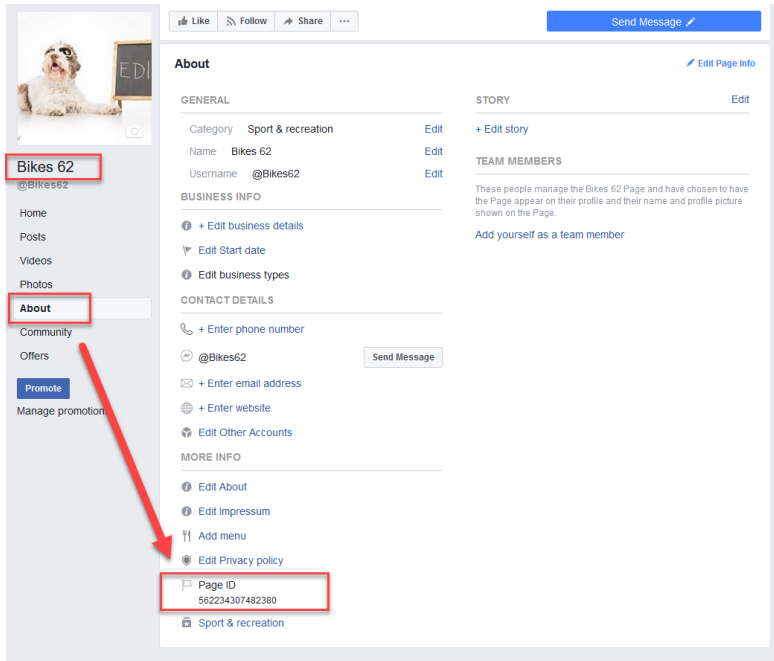
The screenshot shows a web-based configuration interface for 'System Management : Messaging Services : MyPage21_pst'. The interface includes a toolbar with options like 'Save', 'Save and Close', 'Delete', 'Add New', 'Reset', and 'Close'. Below the toolbar, there is a 'Basics' section with the following fields:

| | |
|----------------------|--|
| Name | MyPage21_pst |
| Type | facebook |
| Service Connection | https://graph.facebook.com/me/messages |
| Authentication URL | |
| Authentication Scope | |

At the bottom, there is an 'Authentications' section with a plus sign icon.

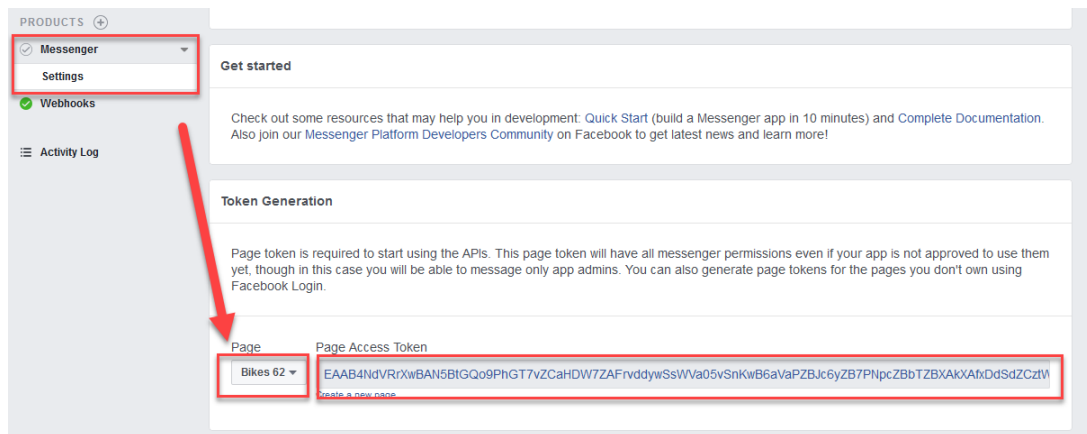
4. In the Authentications section insert the Page ID from your Facebook page and provide password. (see next step for password)

PAGE ID, can be found from Facebook.com side:



| Authentication ID | Password |
|-------------------|----------|
| 2035607336666570 | **** |
| 289639251515062 | **** |
| 562234307482380 | **** |

PASSWORD, can be found from developers.facebook.com side

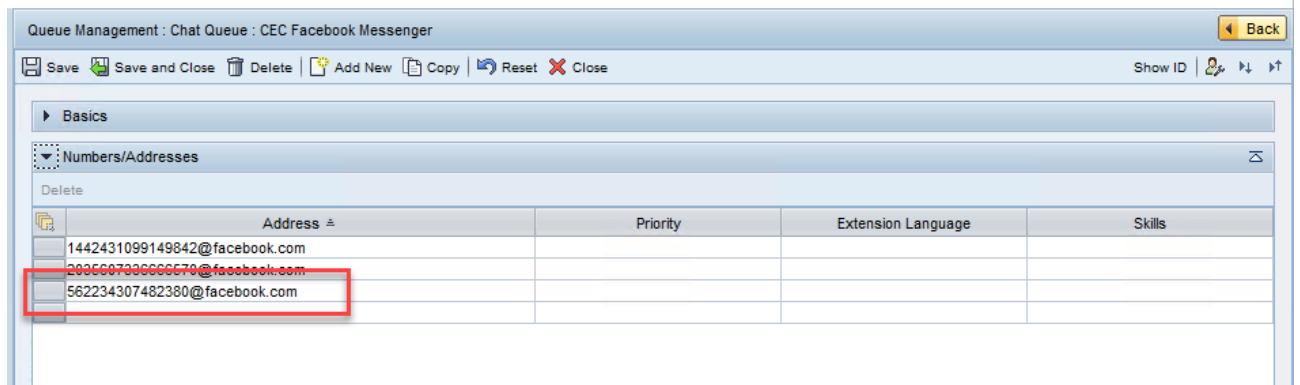


| Authentication ID | Password |
|-------------------|--|
| 2035607336666570 | **** |
| 289639251515062 | **** |
| 562234307482380 | EAAB4NdVrR... (highlighted with a red box) |

NOTE: Entered password will be hidden from view.



5. Create a CHAT queue with following Address configuration.



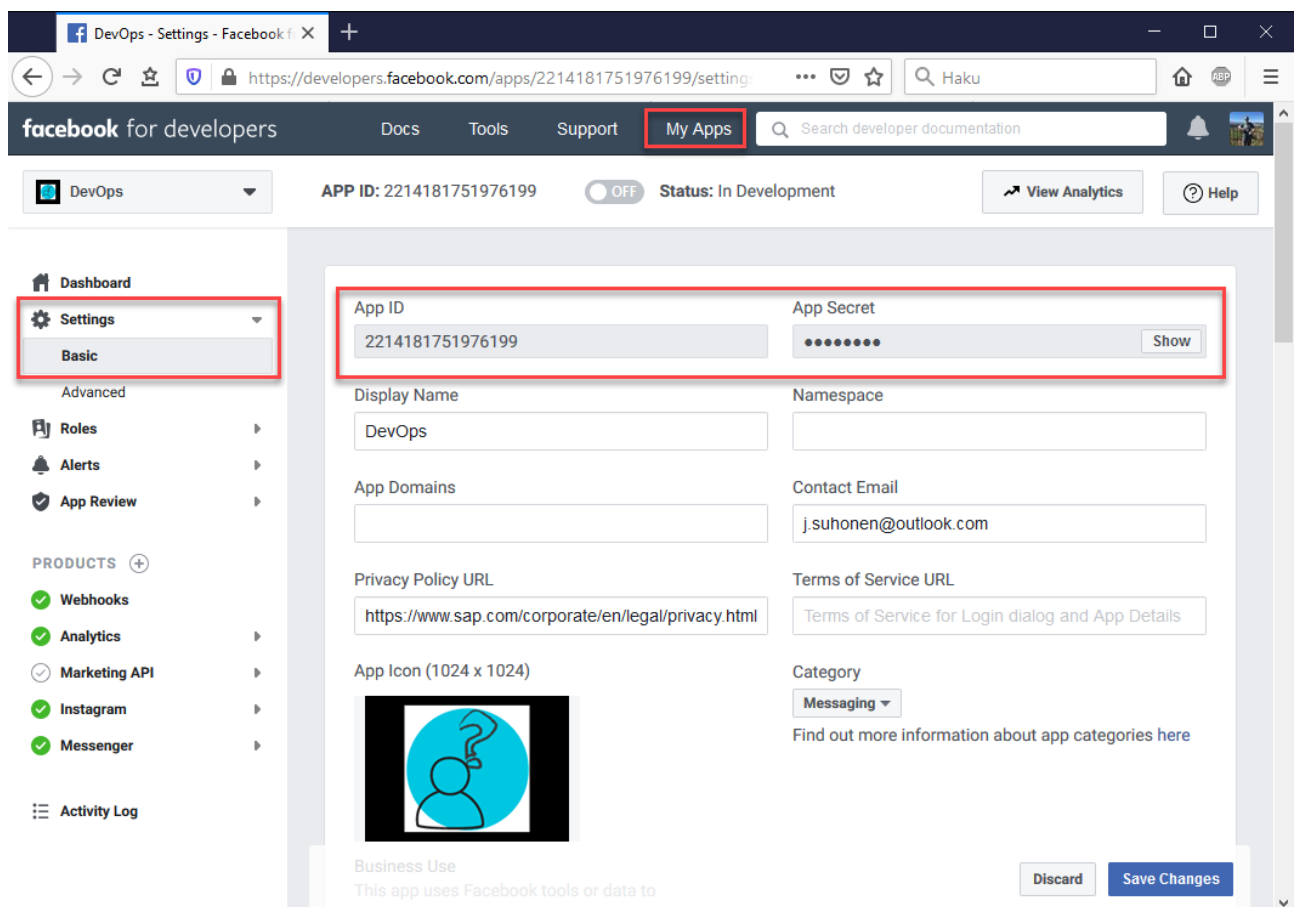
System Configurator configuration is now ready. You can now test the integration.

4 Securing Your Facebook Connection

The example given above creates an unsecure connection and messaging is transferred in plain text format over the internet. This allows anyone to listen the messaging in internet. We do not recommend exchanging confidential information over unsecure connection. This chapter describes how to enable secure integration with Facebook.

NOTE The system allows only one account to be configured like this. Therefore, recommendation is to group all Facebook pages under one App ID.

1. Open <https://developer.facebook.com> and login.
2. Open your app, navigate to Settings – Basics
3. Copy the App ID and App Secret to Notepad for example



The screenshot shows the Facebook Developer console interface. The browser address bar displays the URL: <https://developers.facebook.com/apps/2214181751976199/settings>. The page title is "DevOps - Settings - Facebook". The navigation bar includes "Docs", "Tools", "Support", and "My Apps". The main content area shows the "Basic" settings for the app "DevOps" (APP ID: 2214181751976199). The "App ID" field contains "2214181751976199" and the "App Secret" field is masked with dots. The "Basic" menu item in the left sidebar is highlighted with a red box. The "App ID" and "App Secret" fields are also highlighted with a red box. The "App Secret" field has a "Show" button next to it. The "Display Name" is "DevOps", "App Domains" is empty, "Privacy Policy URL" is "https://www.sap.com/corporate/en/legal/privacy.html", and "App Icon" is a blue circle with a white question mark. The "Category" is "Messaging". The "Save Changes" button is visible at the bottom right.



In System Configurator navigate to System Management – Messaging Services and open your messaging service.

4. Insert

- App ID to Authentication ID field
- App Secret to Password field

5. Save the changes.

6. Use the same App ID at queue level config as well.



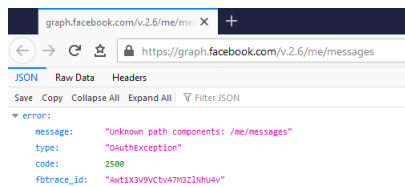
5 Troubleshooting and Tips

5.1 Facebook Certificate

For http connections Facebook uses a certificate. It may happen that you don't have this certificate at Contact Center application server. Easiest way to find out if this the issue is to:

Open following webpage at server <https://graph.facebook.com/me/messages> - if browser indicates that certificate is not trusted then you don't have the certificate -- you can use the browser to obtain the cert. Look for certificate details and use copy function to obtain it. Store to your server certificate store. After certificate is installed try to refresh the browser.

When all are ok , you should see following:



5.2 Testing the Interface

The connection is all about HTTP GET and POST commands. Therefore, opening the URL in browser doesn't return anything. If you wish to check if interface is returning anything, you can try with SOAP UI tool.

5.3 Facebook Shortcut

Facebook allows you to make short URL to your messenger account. The format is https://m.me/Facebook_ID With the link you can easily create Facebook us buttons to your webpage. Please anyhow remember to check Facebook branding guidelines before implementing this feature.



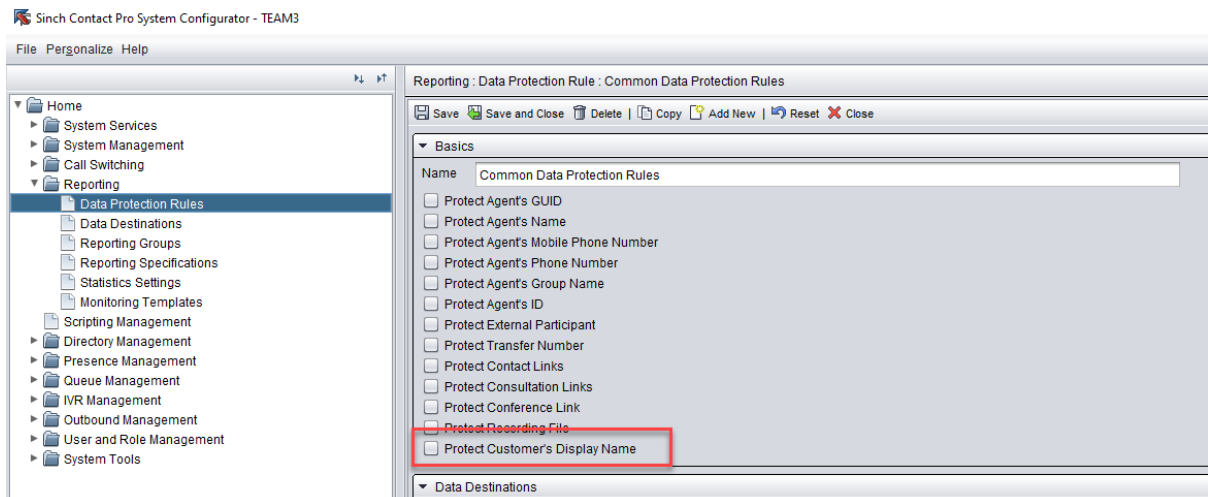
5.4 Facebook ID vs. Facebook Name

Facebook communicates with IDs, like Contact Center has GUIDs for everything. In FP14 release we enhanced the system to query customer name as well. However, in some cases user might have tight security setting in Facebook and it doesn't allow displaying Facebook name. In such a case ID is displayed instead. This applies to Communication Panel user interface.

5.4.1 Display Name (Facebook name)

Contact Pro version 22Q1 supports displaying customer's social name in Communication Panel in the picklist and history views. Additionally, there is new setting to hide the display name. By default, the the setting is enabled. If you want to see the display name in Communication Panel, you should disable this setting.

For the on-premise release, the same will be available in FP19.





5.5 Issues and Log Reading

NOTE The following applies only for on-premise customers.

When reporting issues to support, the most valuable information is SMS Server at trace level. SMS Server supports also debug secure parameter, which reveals more details of incoming message. We do not recommend enabling the parameter for production systems.

You may also try to troubleshoot it by yourself by sorting errors and warning (ERR|WRN in BLV).

- Incoming message from Facebook Messenger can be seen with the following filter `HttpSmsServer::OnRequest /sms/facebook/facebook`
- Incoming message from any channel: ***http***