Index

1. Hermes.NET TheLink adding Facebook Messenger capabilities 2

2. Configuration of Direct Messaging with Messenger 2

2.1. Setup 2

3. Testing 5

3.1. When nothing works 5

1. Hermes.NET TheLink adding Facebook Messenger capabilities

Since latest changes done on July and August in Facebook platform to make public accessible the Messenger API for developers we can now provide Hermes with new functionality on thelink campaigns related to direct message management.

Currently we support the sent, not sent, delivered and readed events for messages that agents sent to users. The events are received in practically real time using the Facebook callbacks facility this means a lower use of calls to the GRAPH API and minimizing the network load used in previous versions.

As a drawback, Facebook changes has brought some limitations on current features that can impact in the way we did direct messaging on Hermes. Noticeable Facebook is limiting now the number of calls to GRAPH API an application can make on a page (the maximum number of calls is allowed to less than 4800 times in 24 hours). As a result, current customers could find their platforms not working along intervals of 24 hours for campaigns on pages that exceeds the limits imposed by GRAPH if they use the direct messaging interface.

2. Configuration of Direct Messaging with Messenger

As pre requisites we need a IIS web server to deploy thelink web application. This web server needs to be public accessible and needs to be configured to work with https protocol and be certified by a public sign entity.

2.1. Setup

We'll deploy on this server TheLink web application and after that the configuration can be done on the web.config file. Thelink web application contains the Facebook webhook in charge to receive the events from Messenger platform and to store it to be used by MediaService. There is too a web service called MessengerService that is the link between the Messenger webhook and MediaService. As the application is public available this webservice can be protected using a white list of allowed ip.

As an example we can do this kind of setup:

```
<appSettings>
        <add key="OctopusRedirectURL" value="https://thelink.vocalcom.com/hermes_net_v5/thelink/" />
        <add key="AllowedIps" value="127.0.0.1,::1"/>
</appSettings>
```

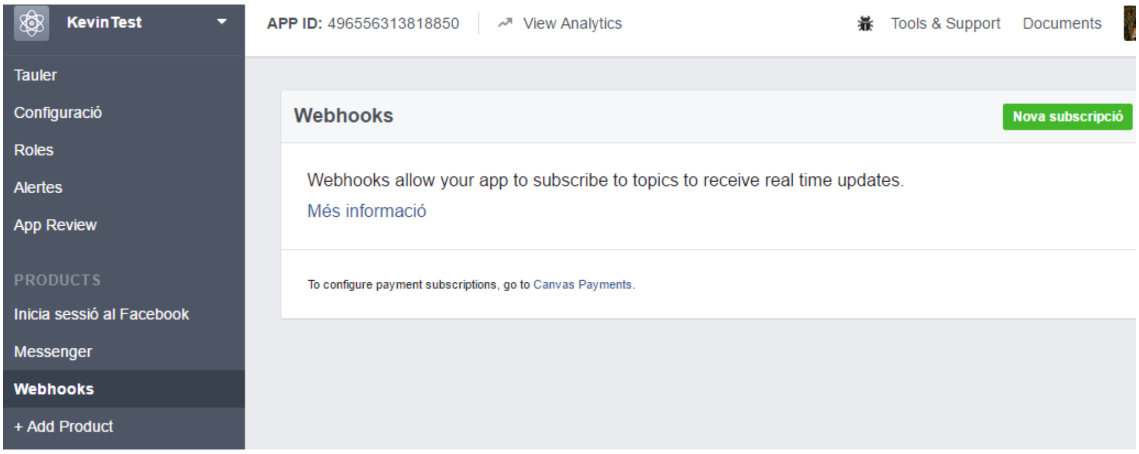
On this example the octopus is replaced by the local thelink application. This can be configured according on the Admin\Web_Service\web.config file too with the parameters:

```
<add key="TheLinkWS"
value="http://localhost/hermes_net_v5/TheLinkCreationCampaignWS/TheLinkCreationCampaignWS.asmx"/>

<add key="TheLinkRedirectUrl" value="https://vpn.infinity.es/hermes_net_v5/thelink/oid/"/>
```

After the setup of thelink web application next step is to connect to our Facebook developers account and create a new o select an existing Facebook Application.

We can find on the left navigation bar a new section called PRODUCTS we need to Add if not present Webhooks and will configure our webhook public endpoint.

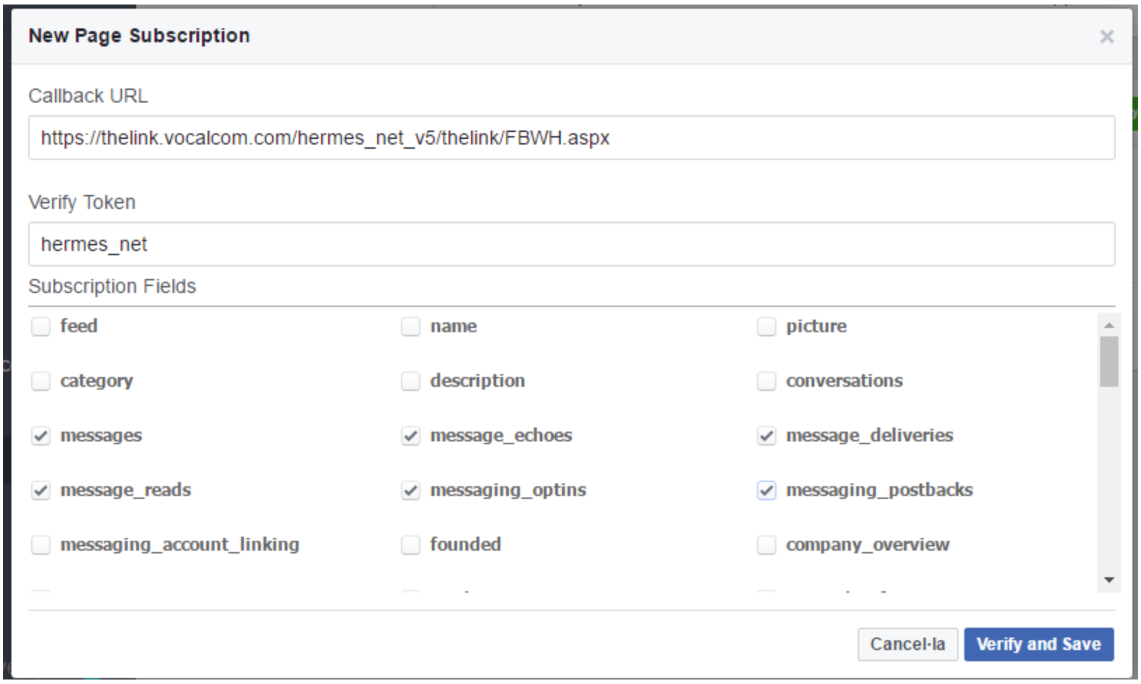


Webhooks allow your app to subscribe to topics to receive real time updates.

To configure payment subscriptions, go to Canvas Payments.

The green button depending of the language used is “add subscription”. We’ll do a new one for “Page”

The interface will show a new dialog to setup our webhooks and permissions required.



Callback URL: https://thelink.vocalcom.com/hermes_net_v5/thelink/FBWH.aspx

Verify Token: hermes_net

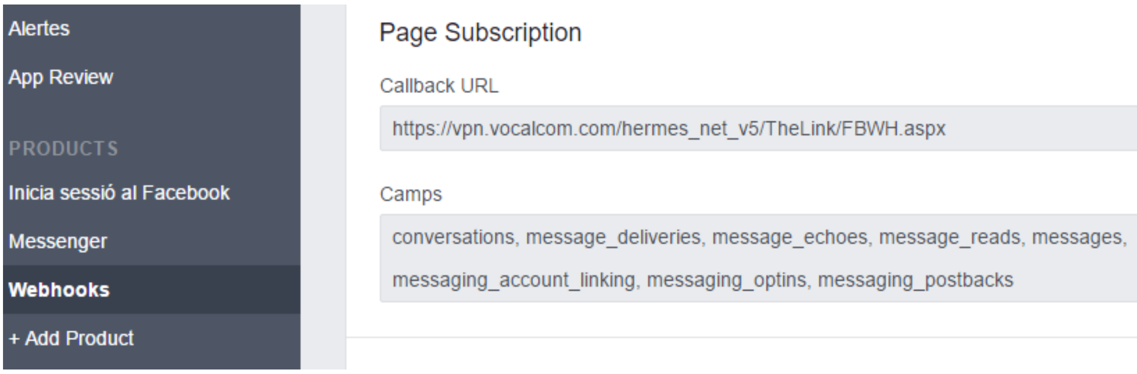
Subscription Fields

☐ feed ☐ name ☐ picture
☐ category ☐ description ☐ conversations
☑ messages ☑ message_echoes ☑ message_deliveries
☑ message_reads ☑ messaging_optins ☑ messaging_postbacks
☐ messaging_account_linking ☐ founded ☐ company_overview

As the webhook “Callback URL” will use the https url to access thelinkweb application adding the webhook endpoint. In our previous exemple: https://thelink.vocalcom.com/hermes_net_v5/thelink/FBWH.aspx

For the “verify token” we’ll write “hermes_net” and then will select the next requiered permissions: `messages, message_echoes, message_deliveries`, `message_reads, ,messaging_optins`, and `messaging_postbacks` under Subscription Fields.

After the verification we’ll have this new subscription added to the Webhooks list.



Page Subscription

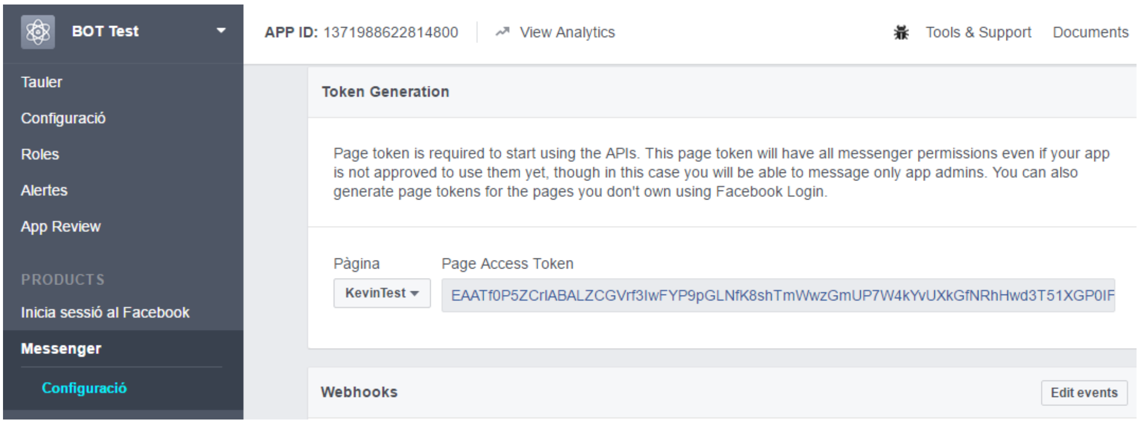
Callback URL

https://vpn.vocalcom.com/hermes_net_v5/TheLink/FBWH.aspx

Camps

conversations, message_deliveries, message_echoes, message_reads, messages, messaging_account_linking, messaging_optins, messaging_postbacks

Now, we need to generate a Page Access Token



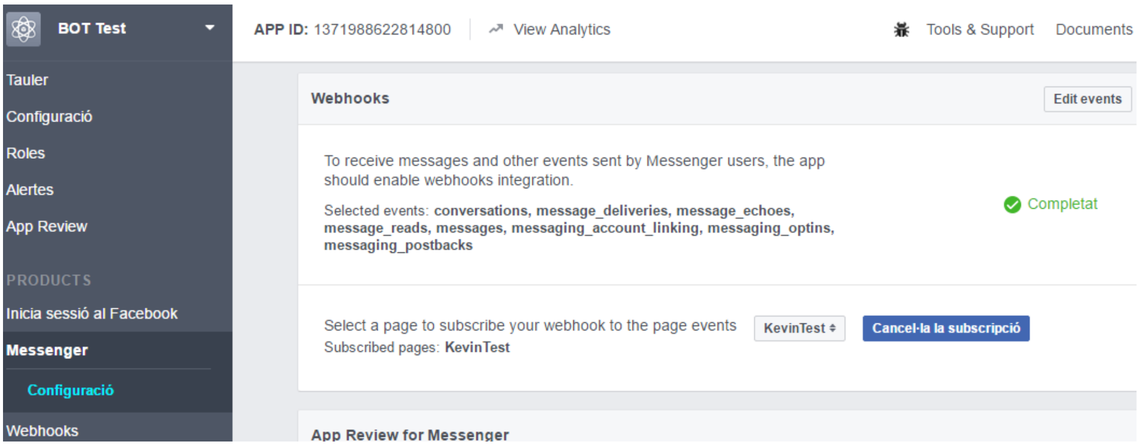
Token Generation

Page token is required to start using the APIs. This page token will have all messenger permissions even if your app is not approved to use them yet, though in this case you will be able to message only app admins. You can also generate page tokens for the pages you don't own using Facebook Login.

Pàgina: KevinTest

Page Access Token: EAATf0P5ZCrIABALZCGVrf3IwFYP9pGLNfK8shTmWwzGmUP7W4kYvUXkGfNRhHwd3T51XGP0IF

After that we can subscribe this page to our Application



Webhooks

To receive messages and other events sent by Messenger users, the app should enable webhooks integration.

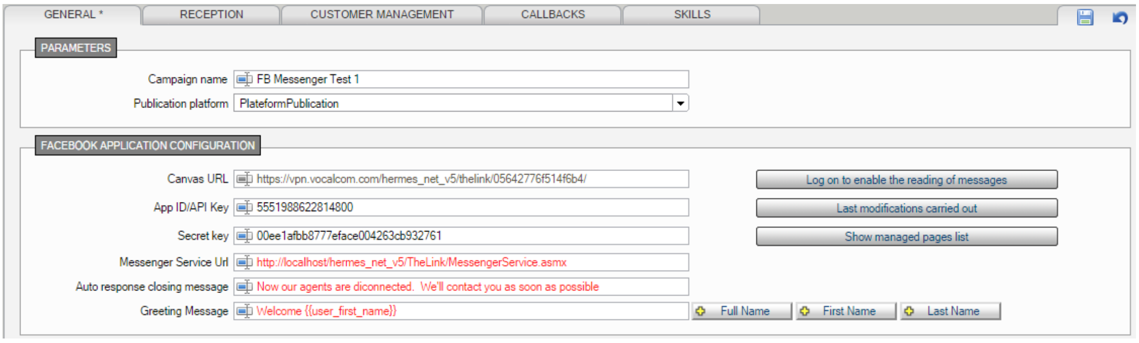
Selected events: **conversations, message_deliveries, message_echoes, message_reads, messages, messaging_account_linking, messaging_optins, messaging_postbacks**

✓ Completat

Select a page to subscribe your webhook to the page events: KevinTest

Subscribed pages: **KevinTest**

The Facebook Messenger configuration is done. We can create now thelink campaign for facebook and login with our user as usual. We ‘ll need to provide the Facebook AppID and Secret Key for the Facebook application that we’ve created. If this application was previously created and is linked to an existing campaign, we can use it only adding the Messenger specific fields as in the image below marked in red



PARAMETERS

Campaign name: FB Messenger Test 1
Publication platform: PlateformPublication

FACEBOOK APPLICATION CONFIGURATION

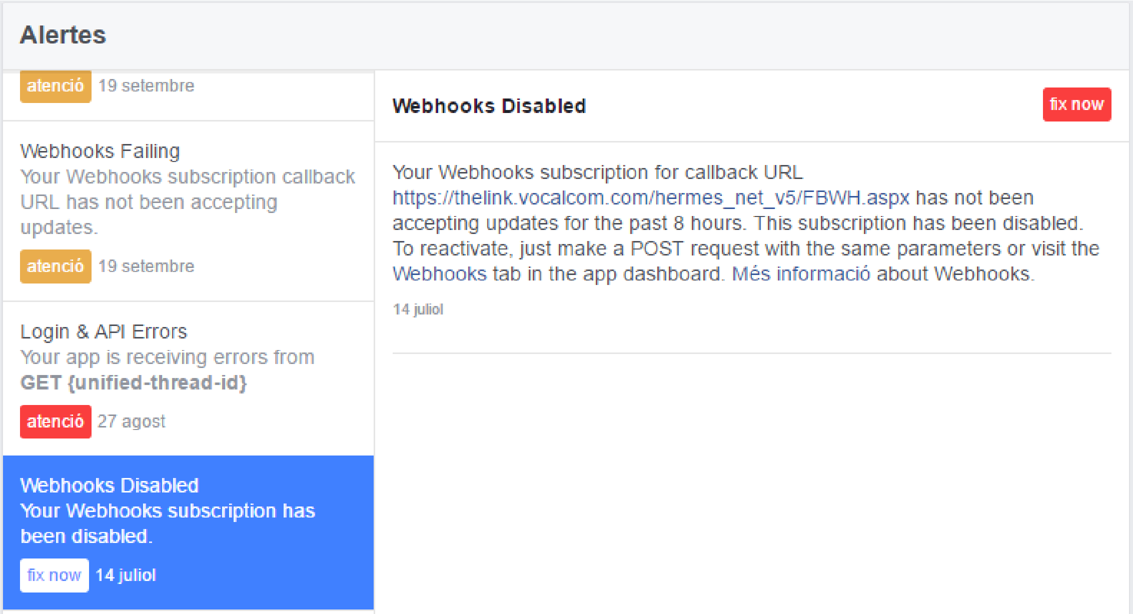
Canvas URL: https://vpn.vocalcom.com/hermes_net_v5/thelink/05642776f514f6b4/
App ID/API Key: 5551988622814800
Secret key: 00ee1afbb8777eface004263cb932761
Messenger Service Url: http://localhost/hermes_net_v5/TheLink/MessengerService.asmx
Auto response closing message: Now our agents are diconnected. We'll contact you as soon as possible
Greeting Message: Welcome {{user_first_name}}

3. Testing

After setup is done and the campaign has been activated assigning to it a queue, agents should receive Direct messages sent from Facebook Messenger (web or mobile) and chat with users in real time.

3.1. When nothing works

First thing to check is Facebook developer’s portal so see on Alerts if we’re getting some problem from webhooks like the one on the next image



Alertes

atenció 19 setembre

Webhooks Failing
Your Webhooks subscription callback URL has not been accepting updates.
atenció 19 setembre

Login & API Errors
Your app is receiving errors from **GET {unified-thread-id}**
atenció 27 agost

Webhooks Disabled
Your Webhooks subscription has been disabled.
fix now 14 juliol

Webhooks Disabled fix now

Your Webhooks subscription for callback URL https://thelink.vocalcom.com/hermes_net_v5/FBWH.aspx has not been accepting updates for the past 8 hours. This subscription has been disabled. To reactivate, just make a POST request with the same parameters or visit the Webhooks tab in the app dashboard. Més informació about Webhooks.

14 juliol

If there are alerts it should be fixed following the facebook instructions.

After setup for Facebook Application has been checked we can set DEBUG level to logs on the webhooks side to see if the events are arriving to our web application and check on the MediaService logs if the connection against the MessengerService.asmx is working too. For example, a list of allowed ips not including the MediaService host machine could avoid MessengerService to allow calls to its methods.