





PLATFORM DEEP DIVE / Plugins /  Custom Plugins /  Notifications plugin / Using the plugin / OTP flow / generate-otp

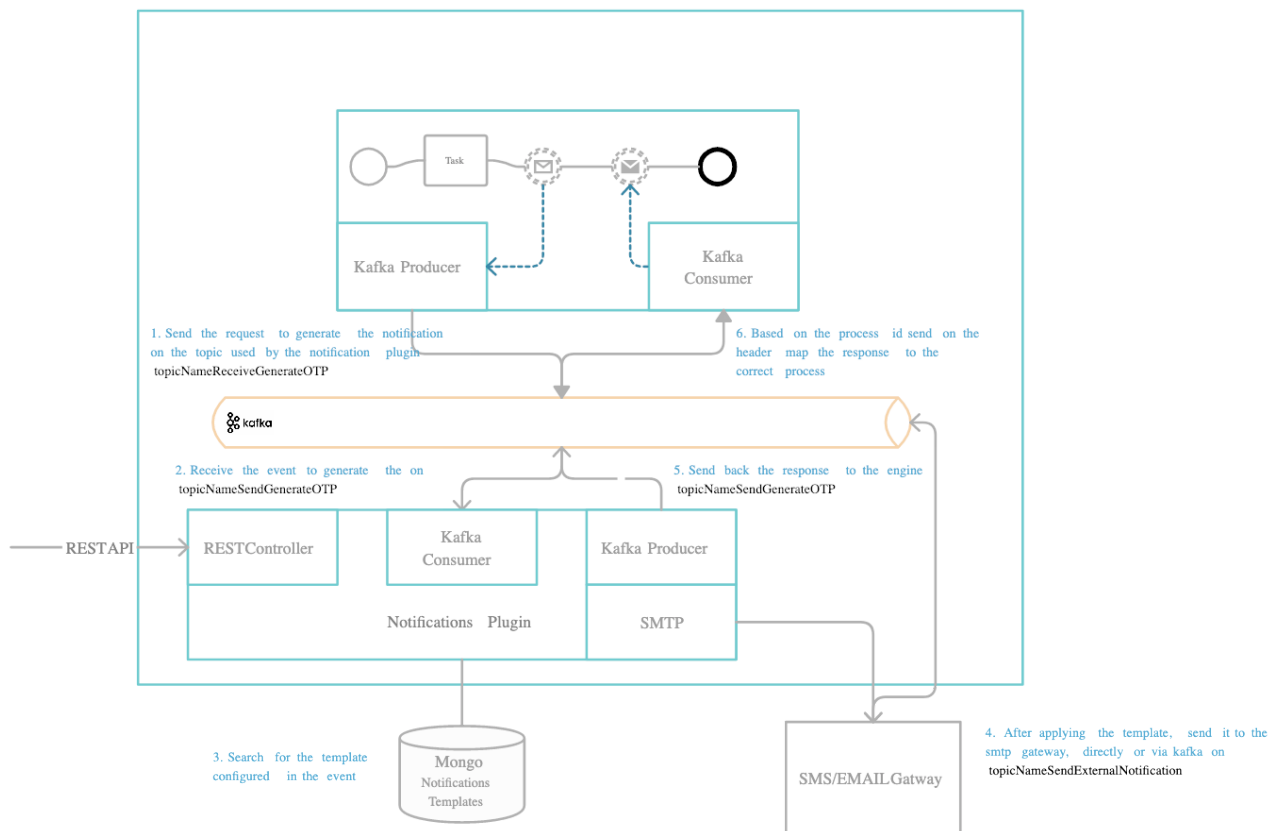
Contents

- PLATFORM DEEP DIVE / Plugins /  Custom Plugins /  Notifications plugin / Using the plugin / OTP flow / Generate OTP
 - Define needed Kafka topics
 - Request to generate an OTP
 - Response from generate OTP
 - Example: generate an OTP from a business flow

PLATFORM DEEP DIVE / Plugins / Custom Plugins / Notifications plugin / Using the plugin / OTP flow / Generate OTP

There are some cases when you will need to generate an OTP (One Time Password) from a business flow, for example when validating an email account.

The notifications plugin handles both the actual OTP code generation and sending the code to the user using a defined **notification template**.



Define needed Kafka topics

Kafka topic names can be set by using environment variables:

- `KAFKA_TOPIC_OTP_GENERATE_IN`
- `KAFKA_TOPIC_OTP_GENERATE_OUT` - after the OTP is generated and sent to the user, this is the topic used to send the response back to the Engine.

⚠ CAUTION

The Engine is listening for messages on topics with names of a certain pattern, make sure to use an outgoing topic name that matches the pattern configured in the Engine.

Request to generate an OTP

Values expected in the request body:

- **templateName**: the name of the notification template that is used (created using the **WYSIWYG** editor)
- **channel**: notification channel: SMS / MAIL
- **recipient**: notification receiver: email / phone number
- **notification template content parameters** (for example, **clientId**): parameters that should be replaced in the **notification template**

Parameters

Custom

From integration

Topics

```
ro.flowx.qa.notification.otp.in.generate.v1
```

Message

```
1 {
2   "templateName": "otpMail",
3   "channel" : "MAIL",
4   "language": "en-US",
5   "clientId" : "${application.client.identificationData.personalIdentificationNumber}",
6   "recipient" : "${application.client.contactData.email.emailAddress}",
7   "contentParams":
8     {
9       "clientId": "${application.client.identificationData.personalIdentificationNumber}"
10    }
11 }
12
13
```

Advanced configuration

Show Headers ☒

```
1 {"processInstanceId": "${processInstanceId}"}
```

Response from generate OTP

Values expected in the reply body:

- processInstanceId = process instance ID
- clientId = the client id (in this case the SSN number of the client)
- channel = notification channel used
- otpSent = confirmation if the notification was sent: true or false

- error = error description, if any

Example:

The screenshot shows a modal window titled "Audit log details" with a close button (X) in the top right corner. The modal contains the following information:

- Event:** process instance, message receive, 18 Oct 2022 at 4:12 PM
- Url:** ai.flowx.updates.qa.notification.otp.generate.v1
- Body:**

```
1 {"processInstanceId":739452,"clientId":"1871201460101",
  "channel":"MAIL","otpSent":true,"error":null}
```

The background shows a blurred view of the FlowX.AI interface, including a sidebar with "Active" and "Audit" sections, and a main area with a table containing process instance details.

Example: generate an OTP from a business flow

It is important to identify what is the business identifier that you are going to use to validate that OTP, it can be, for example, a user identification number.

1. Configure the templates that you want to use (for example, an SMS template).
2. Check that the needed topics are configured correctly: the topic used to generate OTP (`KAFKA_TOPIC_OTP_GENERATE_IN`) and the topic used to receive the response (`KAFKA_TOPIC_OTP_GENERATE_OUT`).
3. Use the FLOWX.AI Designer to add a new Kafka send event to the correct node in the process definition.
4. Add the proper configuration to the action, the Kafka topic, and configure the body message.

Kafka Send Action ☐ Automatic ☒ Manual☐ Mandatory ☒ Optional☒ RepeatableAutorun Children? ☐Allow BACK on this action? ☐

Parameters

☒ Custom ☐ From integration

Topics

ai.flowx.in.qa.notification.otp.generate.v1

Message

```
1 {  
2   "templateName": "otpSMS",  
3   "language": "en-US",  
4   "channel": "SMS",  
5   "clientId": "${application.client.identificationData.personalIdentificationNumber}",  
6   "recipient": "${application.client.contactData.mobilePhone.phoneNumber}"  
7 }
```


Advanced configuration

Show Headers ☒

```
1 { "processInstanceId" : ${processInstanceId}, "destinationId": "register_contact_data", "callbacksForAction": "action21" }
```

Save 

5. Add a node to the process definition (for the Kafka receive event).
6. Configure on what key you want to receive the response on the process instance params.

Topic Name	Key Name
ro.flowx.updates.qa.notification.otp.generate.v1	otpCheck 

Was this page helpful?