

A cluster of colorful gears in various sizes and colors (red, blue, green, yellow, purple) is located in the top-left corner of the page.

betaWorks

IBM Integration Bus

Integration with IBM API Connect On Bluemix

Featuring:

- Creating IBM Secure Gateway connection
- Deploying REST API within IIB
- Pushing REST API to IBM API Connect (APIC)
- Publishing an APIC product containing the REST API
- Testing the published REST API

November 2016

Hands-on lab built at product
Version 10.0.0.6

Contents

1. INTRODUCTION	3
2. TASKS	4
3. PREPARE IBM SECURE GATEWAY ENVIRONMENT	5
3.1 CREATE IBM SECURE GATEWAY ON BLUEMIX	5
3.2 INSTALL AND CONFIGURE IBM SECURE GATEWAY CLIENT ON THE VM	11
3.3 START IBM SECURE GATEWAY CLIENT	15
4. REGISTER FOR API CONNECT ON BLUEMIX	17
5. PREPARE IIB ENVIRONMENT	19
5.1 CONFIGURE TESTNODE_IIBUSER FOR REST APPLICATIONS.....	19
5.2 OPEN IIB LOG MONITOR.....	20
5.3 CONFIGURE TESTNODE_IIBUSER TO WORK WITH DB2	20
6. PREPARE IIB REST APPLICATION	21
6.1 IMPORT AND DEPLOY PRE-REQUISITE IIB PROJECTS	21
6.2 PUSH REST API DEFINITION TO IBM API CONNECT – WEB UI	22
6.3 PUSH REST API DEFINITION TO IBM API CONNECT – CLI.....	26
7. VIEW REST API IN IBM API CONNECT	29
8. TEST REST API	34
8.1 SET UP AND RUN THE TEST	34
8.2 DETERMINE URL OF PUBLISHED REST API	37
END OF LAB GUIDE	38

1. Introduction

This lab guide will show you how to publish a REST API that is deployed on **IBM Integration Bus (IIB)**, using **IBM API Connect (APIC)** on **IBM Bluemix**, ready to be controlled from anywhere with an internet connection, through **IBM Datapower Gateway**, which is provided as part of the API Connect environment on Bluemix.

You will also be shown how to set up a secure connection between your IIB host and Bluemix, using a Bluemix **IBM Secure Gateway** service. This connection will be opened from the IIB host and will enable inbound REST API requests, originating from API users and routed by the IBM Datapower Gateway, to be received by the host.

You will see 3 different URLs used for the REST API:

1. A URL that is entered in a browser on the IIB host with the deployed REST API. It will have this format:

```
http://<IIB host address>:7800/HR_Services/resources/employees/  
{employeeNumber}
```

An example request will look like this:

```
http://192.168.126.205:7800/HR_Services/resources/employees/  
000010
```

This URL requests the details of employee 000010 from the HRDB database and if successful, those details will be returned in the browser.

2. A URL referencing a cloud host and port, replacing the original host IP address and port. This is configured on the IBM Secure Gateway service on Bluemix, which maps the URL to the original one.

The URL will be used by the IBM DataPower Gateway and will have this format:

```
http://<cloud host>:<cloud port>/HR_Services/resources/employees/  
{employeeNumber}
```

An example request will look like this:

```
http://caplonsgprd-3.integration.ibmcloud.com:15186/HR_Services/  
resources/employees/000010
```

3. Finally, the published URL. This is made available to users of the REST API through applications which subscribe to a product created in IBM API Connect, containing the API.

The host IP address and port will again be replaced:

```
https://<API Endpoint Base URL>/HR_Services/resources/employees/  
{employeeNumber}
```

An example request will look like this:

```
https://api.eu.apiconnect.ibmcloud.com/iibttester99workshop-dev/sb/  
HR_Services/resources/employees/000010
```

This is used by API Connect to define access controls that can be implemented on the DataPower Gateway, which maps the URL to the previous one.

2. Tasks

In this lab, you will perform the following tasks:

Within Bluemix

- 1) Add an IBM Secure Gateway service
- 2) Add the IIB host system as a destination (the other end of a connection with the Secure Gateway) and determine the cloud host and port that will be used in the REST API

Within the IIB host system, as user iibadmin

- 1) Install the Secure Gateway client on the IIB host system
- 2) Create a connection to the Secure Gateway to allow inbound traffic to the REST API

Within IIB, as user iibuser

- 1) Import and deploy the REST API Service solution from the *Developing a REST API using a Swagger JSON document* lab to TESTNODE_iibuser.
- 2) Test the service using a browser
- 3) Create a new URL for the service using the cloud host and port from the IBM Secure Gateway environment
- 4) Test the service with the new URL
- 5) Push the deployed REST API definition to IBM API Connect on Bluemix, using the new host and port details, and defining a new product for use in APIC

Within API Connect, as the API Manager:

- 1) Configure a test for the API, using the new product
- 2) Test the REST API
- 3) Derive the URL of the published REST API

As an API user (from a web browser)

- 1) Submit requests to the published REST API

Please note: this lab guide makes use of cloud-based IBM Bluemix services which are subject to change at any time. Whilst this document is updated from time to time to reflect these and other changes, it may not be fully up to date and therefore you may need to adapt the instructions it contains accordingly.

3. Prepare IBM Secure Gateway environment

In order to enable inbound connections to systems on the classroom network from outside, we shall be using IBM Secure Gateway.

A Secure Gateway client will be installed in the VM running IIB on your classroom PC. This client will be connected to a Secure Gateway service on Bluemix. The client will be configured to allow inbound connections to a specified port from outside the network.

Login to Windows as the user "iibadmin", password = "passw0rd".

You will need to be an administrative user in order to install the client in this section. Although you will not be installing the client until later in the section, the session will flow more smoothly if you do this now.

3.1 Create IBM Secure Gateway on Bluemix

1. Determine the IP address of the VMware system running IIB, by running command `ipconfig` in a Windows command console.

It will be something like: **192.168.126.xxx** . Save this in a notepad for use later.

```
C:\Users\iibuser>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : localdomain
    Link-local IPv6 Address . . . . . : fe80::c0c6:fid0:8297:ab8%10
    IPv4 Address. . . . . : 192.168.126.216
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.126.2

Tunnel adapter Local Area Connection* 11:
```

2. In a browser, log on to your Bluemix account at

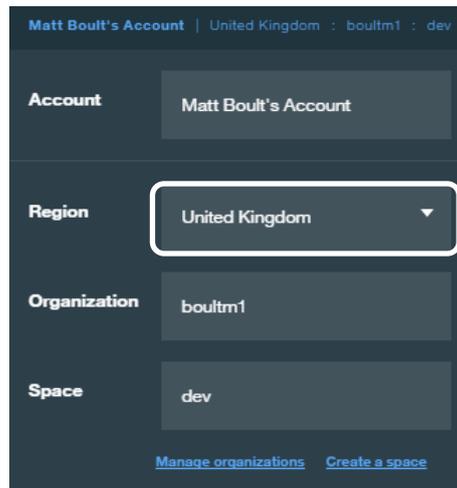
<https://console.eu-gb.ibm.com>.

3. Click on the account details next to profile icon towards the upper right of the screen:



4. Change the region to *United Kingdom* if it is not already set to that. You may choose another region to work in, but latency will be increased during this lab if you do.

Then close the profile details by clicking outside this panel.

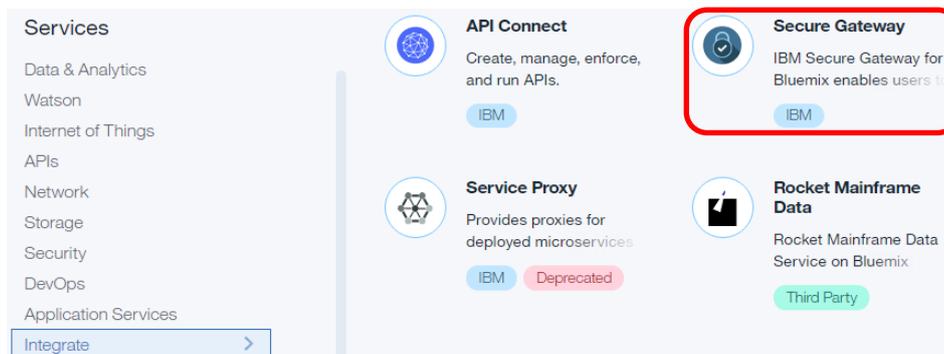


Make a note of your Bluemix **Organization** (which may not be the same as your IBM ID) and **Space**. You will need these details later.

5. Click on the **Catalog** option, again towards the top right of the screen:



6. Scroll down to **Services, Integrate** and select **Secure Gateway**:



At the prompt, click **CREATE** to create the service.

- 7. Select the option to add a gateway:

Secure Gateway 

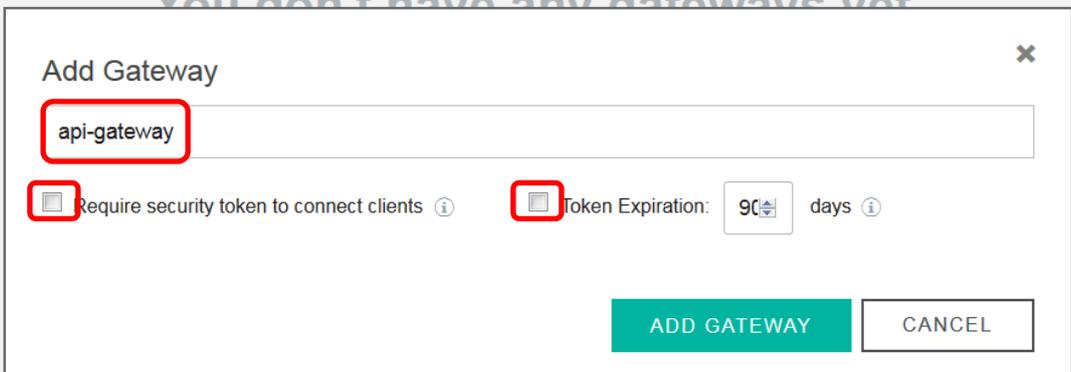
You don't have any gateways yet.

Already know what to do and have a client below installed, then get right to it and add a gateway. Otherwise, go to the learn section to read what you can do with the service and how to do it.

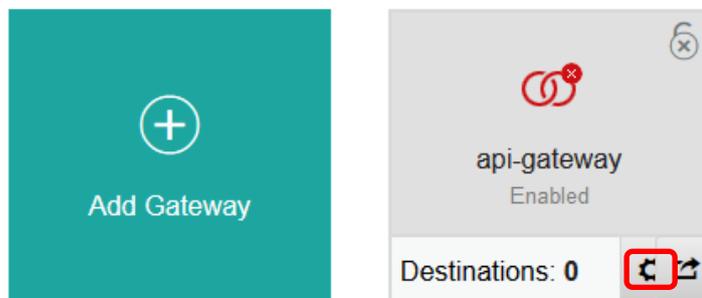
 ADD GATEWAY

 LEARN

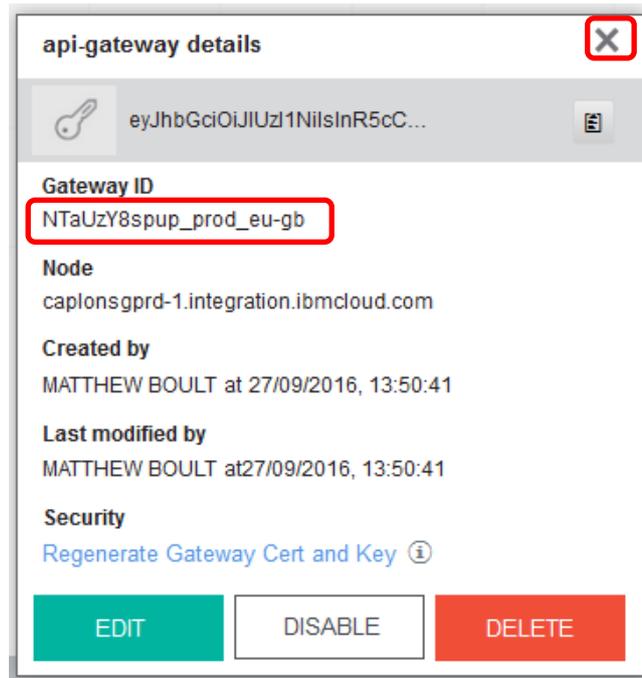
- 8. Name the gateway and de-select the security options (for the sake of simplicity), then click **ADD GATEWAY**.



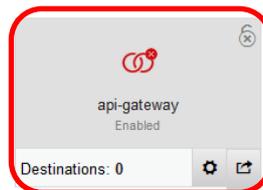
- 9. At the next screen, you will see that an icon has been added for the new gateway. Click, on the cog symbol.



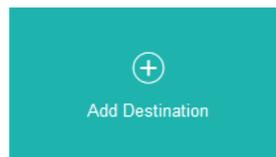
10. Make a note of the Gateway ID, e.g. by copying it to Notepad then close the window by clicking at the top right.



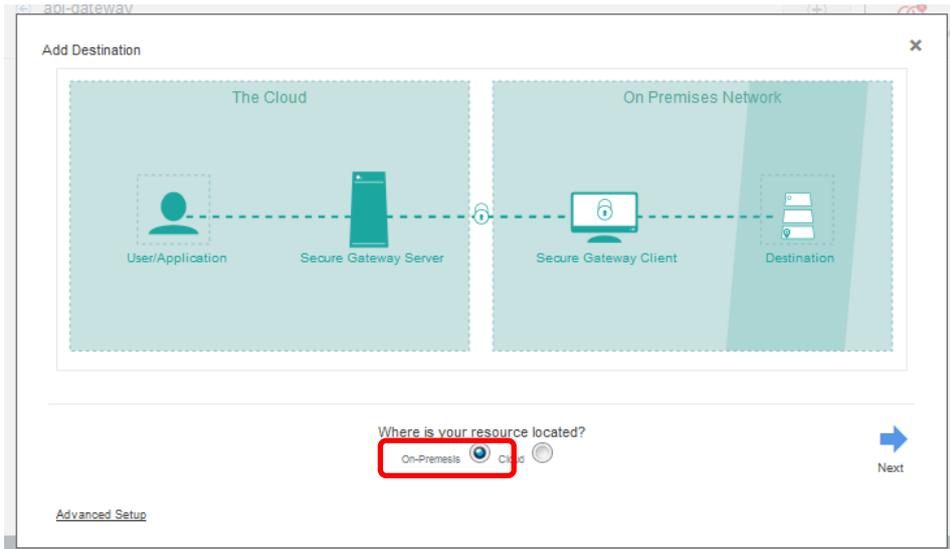
11. Click on the Gateway icon.



12. Then click the **Add Destination** symbol.



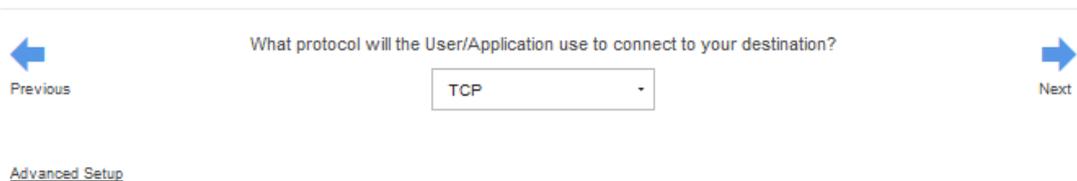
- 13. At the next screen ensure that the resource location is set to 'On-Premise' and click **Next**.



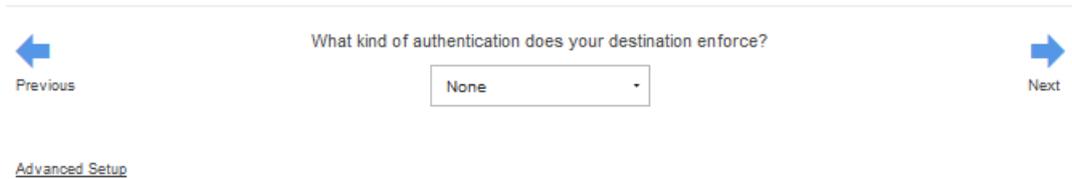
- 14. Add the IP address of the VM system running IIB that you noted in step 1 above. This is the system that will run the Secure Gateway client later on. Specify the port that REST API requests will be sent to (7800) and click **Next**.



- 15. Accept the default protocol and click **Next**.



16. Accept the default authentication (None) and click **Next**.

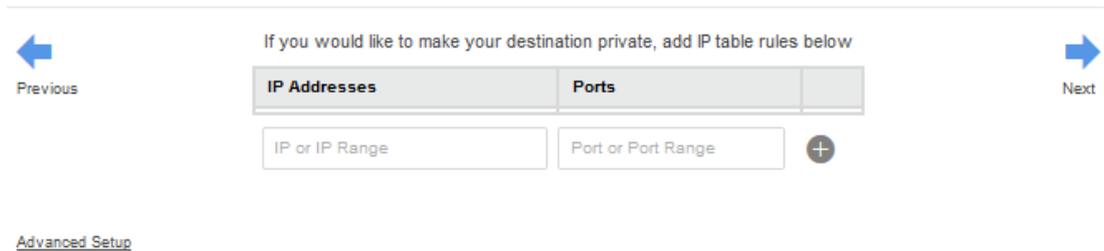


What kind of authentication does your destination enforce?

Previous Next

[Advanced Setup](#)

17. At the next screen click **Next**.



If you would like to make your destination private, add IP table rules below

IP Addresses	Ports	
<input type="text" value="IP or IP Range"/>	<input type="text" value="Port or Port Range"/>	<input type="button" value="+"/>

Previous

[Advanced Setup](#)

18. At the next screen, provide a name to identify the destination and click **Finish**.

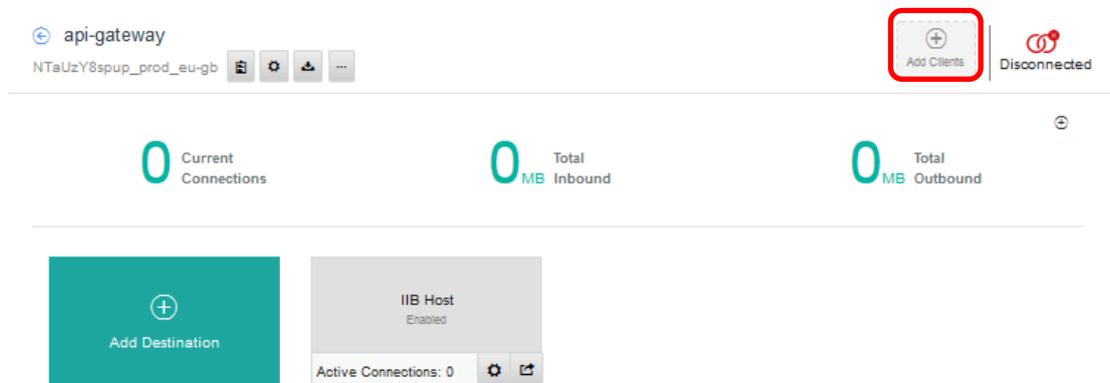


What would you like to name this destination?

Previous Finish

[Advanced Setup](#)

19. At the next screen, you should now see a symbol for the destination. Note the status at top right is currently 'Disconnected'. Click **Add Clients**.

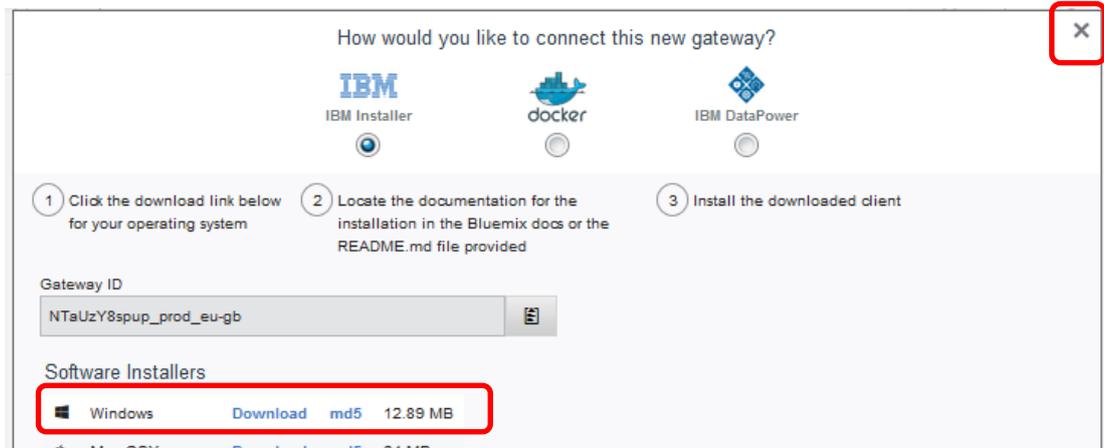


api-gateway
NTaUzY8spup_prod_eu-gb

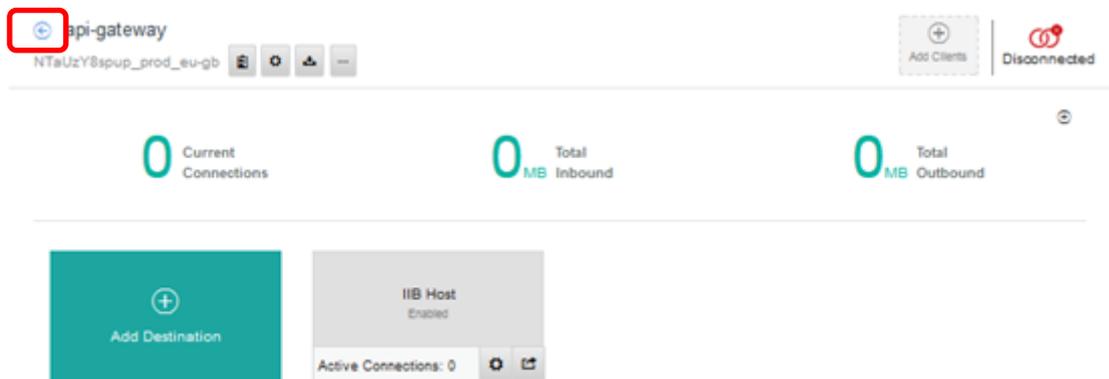
0 Current Connections 0 MB Total Inbound 0 MB Total Outbound

Active Connections: 0

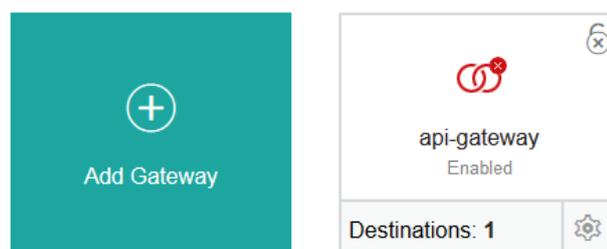
20. At the next screen you are prompted to download the appropriate client for the destination. In the case of the workshop VM, this will be Windows. Download this and save it to a temporary folder, say C:\user\temp, but do not install the software just yet. Close the window when you have finished.



21. At the next screen, click on left arrow to return to the Gateway view.



22. You should now see this. Note the destination counter is now '1'. Also note the interlocking red circles indicating that there are no connected destinations yet.

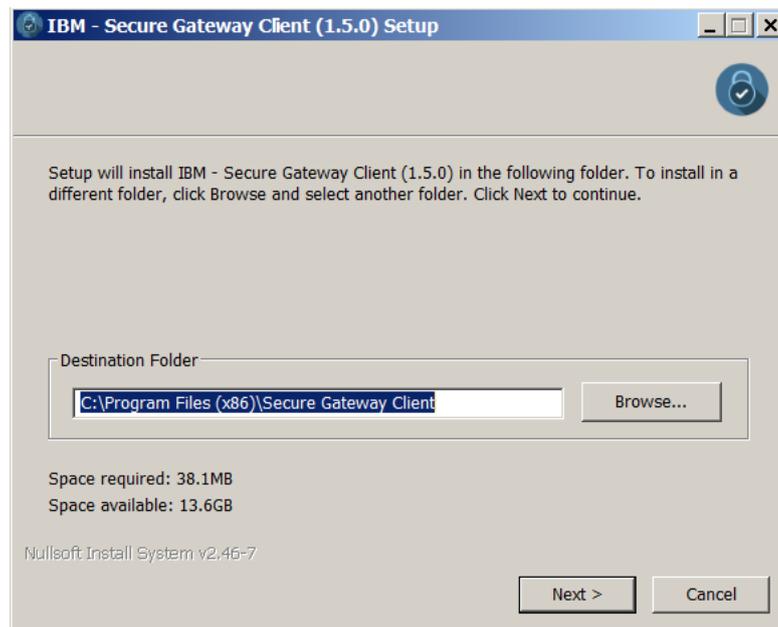


3.2 Install and Configure IBM Secure Gateway client on the VM

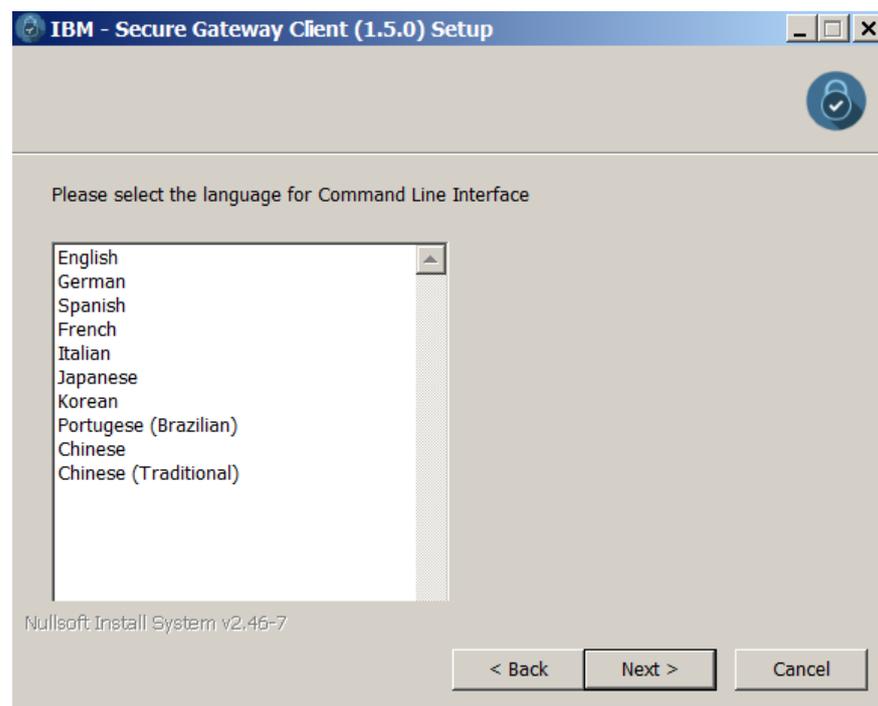
1. If you have not already done so, log on to the IIB Virtual Machine as user *iibadmin*, with password *passw0rd*.

2. Now install the IBM Secure Gateway client that you saved to a folder earlier on (e.g. `C:\user\temp`).

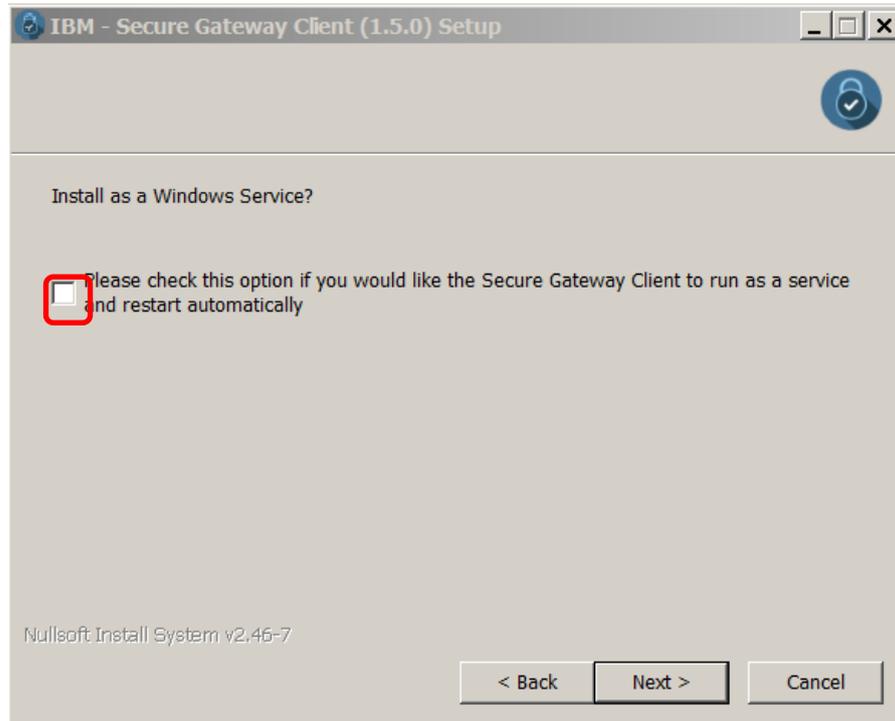
Accept the default destination folder and click **Next**.



3. Choose the appropriate language for the CLI and click Next.



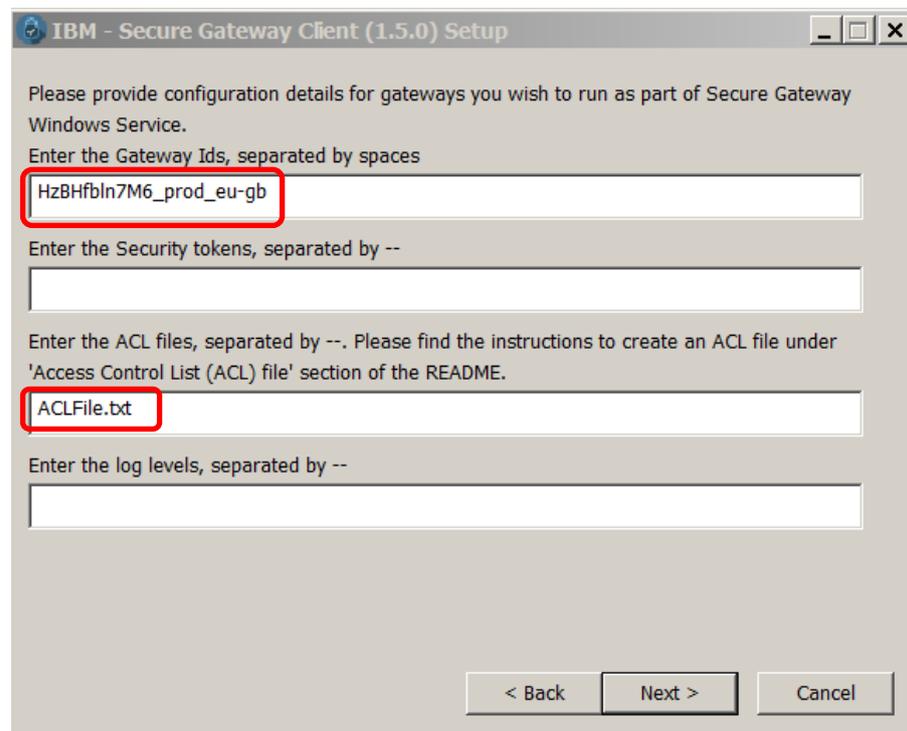
4. Do **not** start the client as a service. This way, you retain control over it and only connect to the Secure Gateway when you need to. Click **Next**.



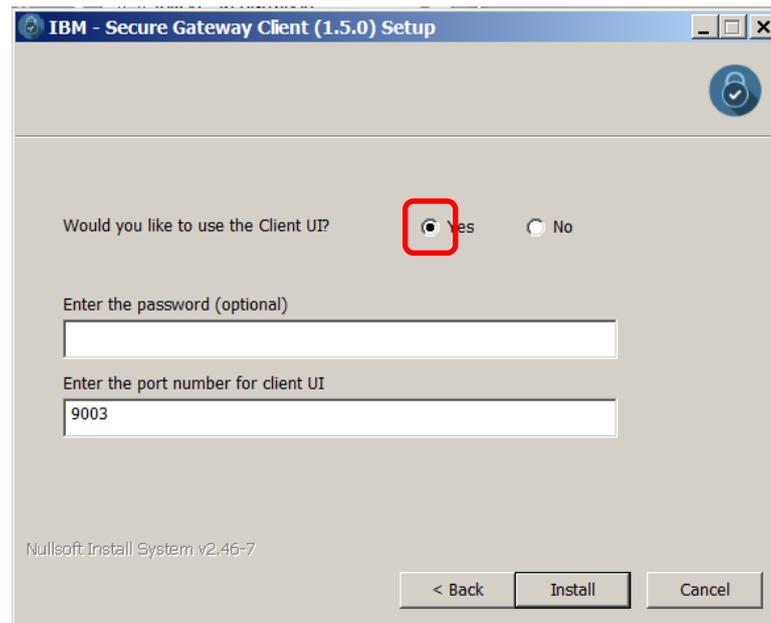
5. Provide the Gateway ID that you copied in step 9 in the last section.

After you have installed the Secure Gateway client, you will need to create an ACL file for access control. You will create this file in the next step.

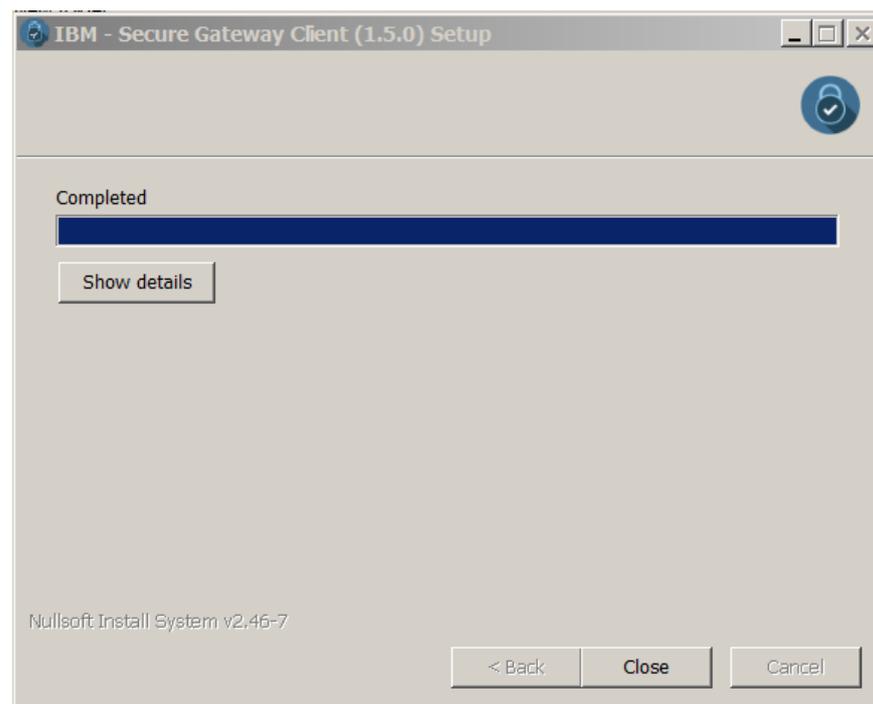
Provide a name for this file in the prompt and then proceed click **Next**.



- Click **Install**, accepting the default to use the Client UI.



- Installation will take a few minutes, during which time you will see a command window open up temporarily. When it has completed, click Close.



- When installation has been completed, create your ACL file in the installation folder of the Secure Gateway client. This will be as follows if you have accepted the default location:

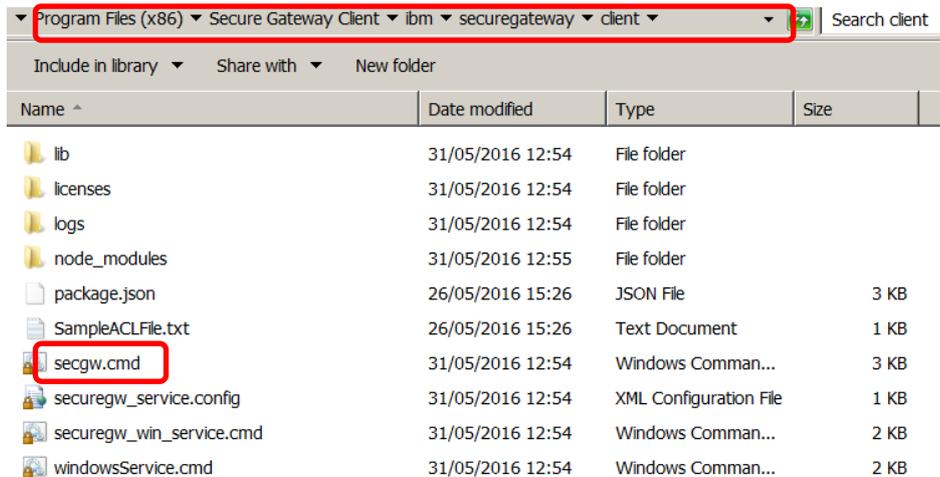
```
C:\Program Files (x86)\Secure GatewayClient\ibm\securegateway\client
```

It should contain this single line to allow inbound connections to port 7800:

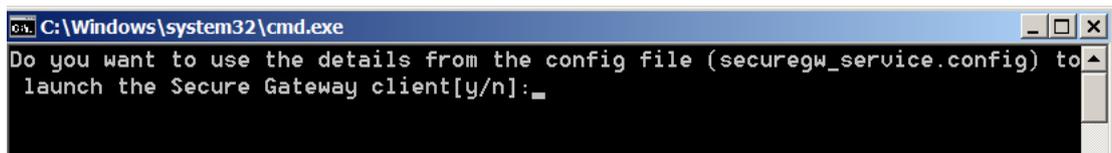
```
acl allow :7800
```

3.3 Start IBM Secure Gateway client

1. Now start the client by double-clicking on file `secgw.cmd` in the sub-folder shown.

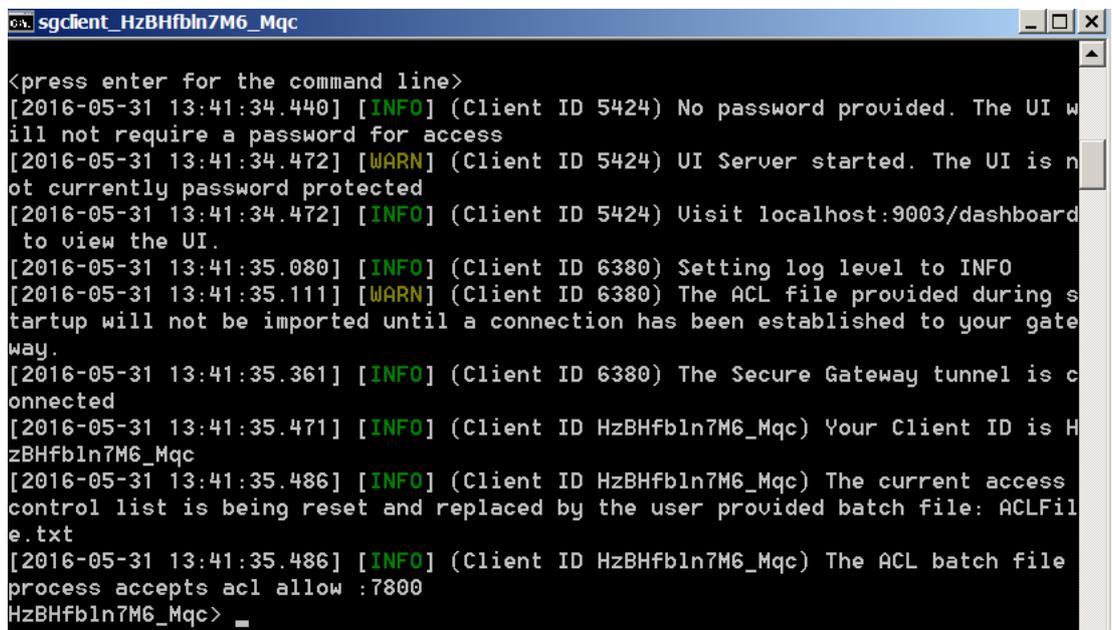


2. At the command line prompt, enter 'y':

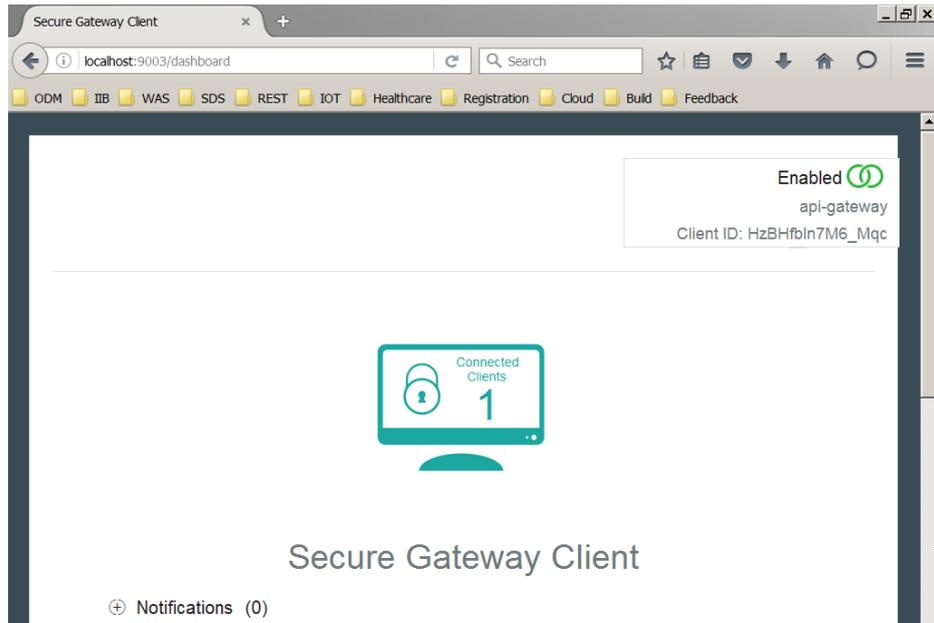


3. The client should start successfully and display output like this, if you scroll down towards the end of the screen.

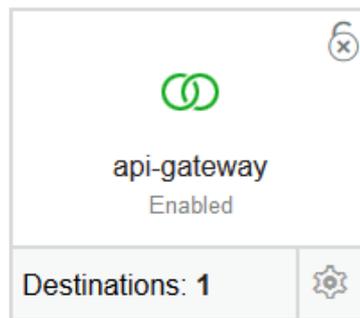
Important – this session needs to be kept running for the duration of this lab, otherwise Bluemix will not be able to reach the REST API. It can be ended later using Ctrl-C.



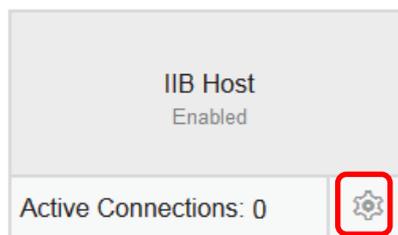
- A browser windows should also open like this to show the client is connected.



- Go back to the Secure Gateway on Bluemix, refresh the screen and you should now see the status indicator is green to indicate an active client:



- Click on this to display the destination and then click on the cog symbol.



- In the resulting display, you will see the cloud host and port that will be substituted for the host and port of the IIB host in the REST API UTL. You will need these details later on when you are logged on as a different user. Copy these into a file e.g. in C:\user\temp. (In the VM environment, the **COPY** button may not copy the text to the clipboard and you may have to use Ctrl-C instead).



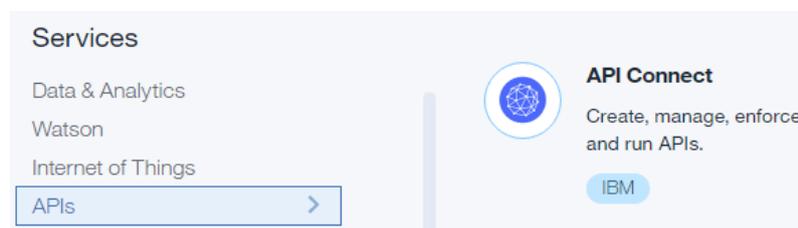
4. Register for API Connect on Bluemix

For this section, you can remain logged in to Windows as the user "iibadmin."

1. Now select the **Catalog** option once more:



2. Scroll to the bottom of the page and then select **API Connect** under **Services, APIs**:

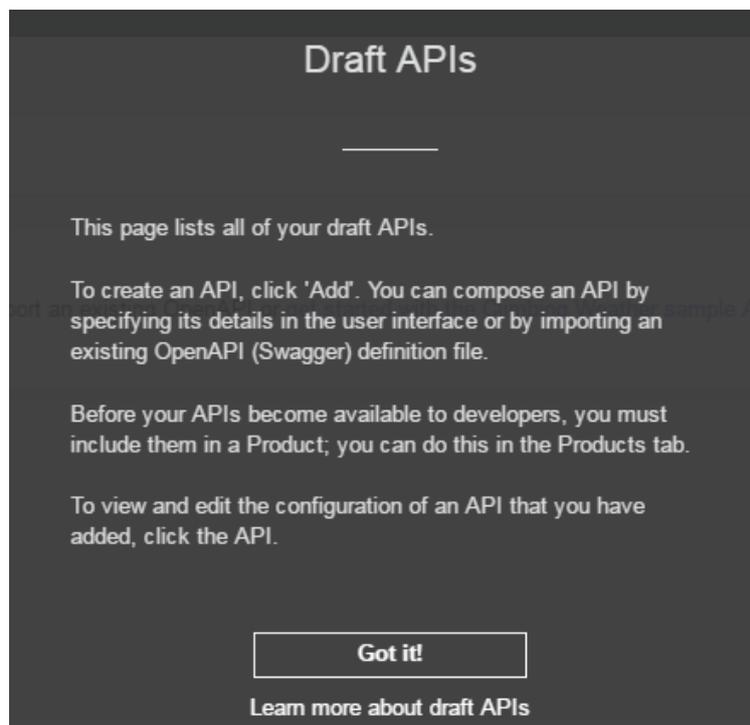


At the prompt, click **CREATE** to create the service.

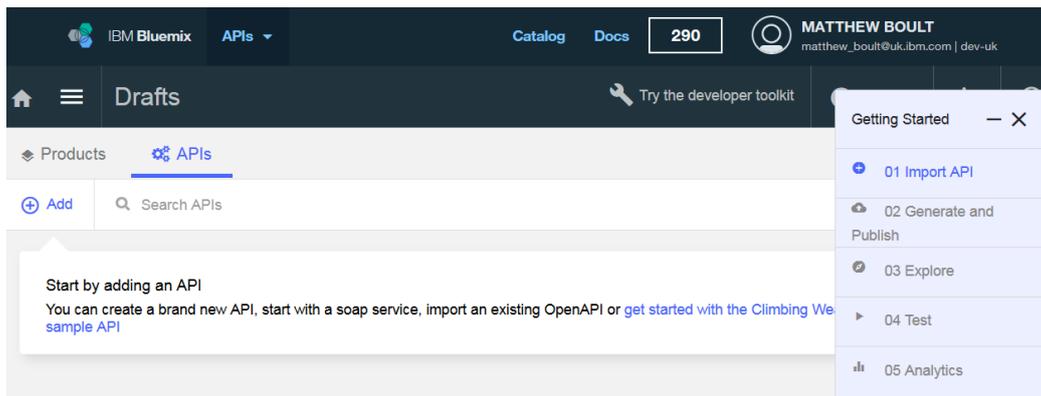
3. This may take a minute or so to set up and to log you on to API Connect.

You should now see the following information about Draft APIs.

Click **Got it!** when you have finished reading it.



4. You should now see the Draft API view in API Connect. Bookmark this or leave the tab open.



5. Prepare IIB environment

5.1 Configure TESTNODE_iibuser for REST applications

The IIB support for the REST API requires some special configuration for the IIB node and server. Cross-Origin Resource Scripting (CORS) must be enabled for the IIB node to execute REST applications. This is also required when testing with the SwaggerUI test tool. See http://www.w3.org/TR/cors/?cm_mc_uid=09173639950214518562833&cm_mc_sid_5020000=1452177651 for further information.

In Windows, switch user and log in as "iibuser", password = "passw0rd".

1. Ensure that TESTNODE_iibuser is started.
2. Check that CORS has been enabled on the IIB node by running the following command in an Integration Console:

```
mqsireportproperties TESTNODE_iibuser
-e default
-o HTTPConnector
-r
```

3. If CORS is enabled, you will see the following lines (amongst others):

```
corsEnabled='true'
corsAllowOrigins='*'
corsAllowCredentials='false'
corsExposeHeaders='Content-Type'
corsMaxAge='-1'
corsAllowMethods='GET,HEAD,POST,PUT,PATCH,DELETE,OPTIONS'
corsAllowHeaders='Accept,Accept-Language,Content-Language,Content-Type'
```

4. If CORS has not been enabled, run the following commands:

```
mqsichangeproperties TESTNODE_iibuser
-e default
-o HTTPConnector
-n corsEnabled -v true

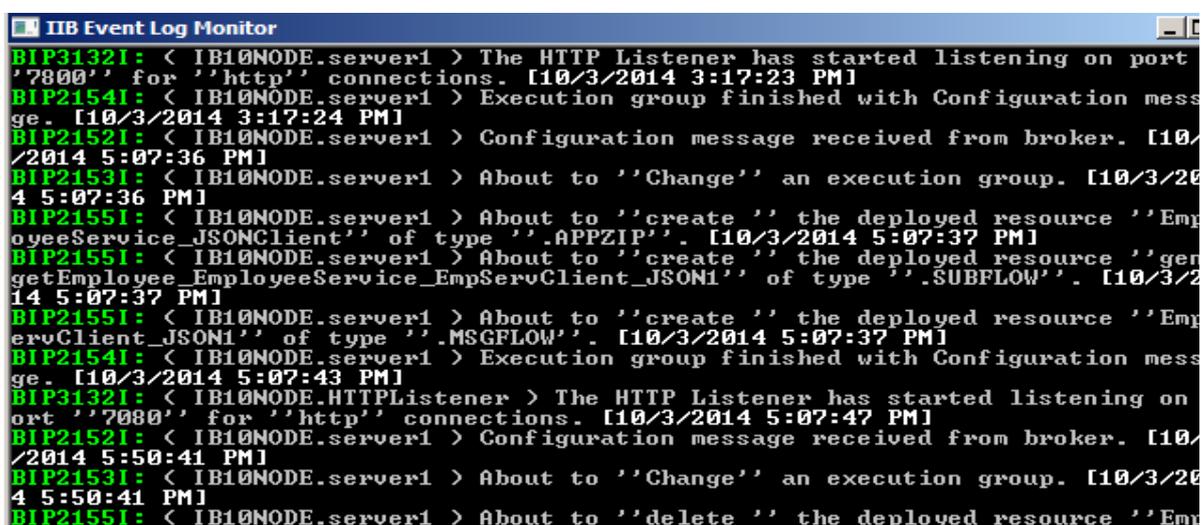
mqsistop TESTNODE_iibuser

mqsistart TESTNODE_iibuser
```

5.2 Open IIB Log Monitor

A useful tool for IIB development on Windows is the IIB Log Monitor. This tool continuously monitors the Windows Event Log, and all messages from the IIB and WMB log are displayed immediately.

From the Start menu, click IIB Event Log Monitor. The Monitor will open; it is useful to have this always open in the background.



```

IIB Event Log Monitor
BIP31321: < IB10NODE.server1 > The HTTP Listener has started listening on port
'7800' for 'http' connections. [10/3/2014 3:17:23 PM]
BIP21541: < IB10NODE.server1 > Execution group finished with Configuration mess
ge. [10/3/2014 3:17:24 PM]
BIP21521: < IB10NODE.server1 > Configuration message received from broker. [10/
/2014 5:07:36 PM]
BIP21531: < IB10NODE.server1 > About to 'Change' an execution group. [10/3/20
4 5:07:36 PM]
BIP21551: < IB10NODE.server1 > About to 'create' the deployed resource 'Emp
oyeeService_JSONClient' of type '.APPZIP'. [10/3/2014 5:07:37 PM]
BIP21551: < IB10NODE.server1 > About to 'create' the deployed resource 'get
EmployeeService_EmpServClient_JSON1' of type '.SUBFLOW'. [10/3/2
14 5:07:37 PM]
BIP21551: < IB10NODE.server1 > About to 'create' the deployed resource 'Emp
erovClient_JSON1' of type '.MSGFLOW'. [10/3/2014 5:07:37 PM]
BIP21541: < IB10NODE.server1 > Execution group finished with Configuration mess
ge. [10/3/2014 5:07:43 PM]
BIP31321: < IB10NODE.HTTPListener > The HTTP Listener has started listening on
ort '7800' for 'http' connections. [10/3/2014 5:07:47 PM]
BIP21521: < IB10NODE.server1 > Configuration message received from broker. [10/
/2014 5:50:41 PM]
BIP21531: < IB10NODE.server1 > About to 'Change' an execution group. [10/3/20
4 5:50:41 PM]
BIP21551: < IB10NODE.server1 > About to 'delete' the deployed resource 'Emp

```

This tool is not shipped as part of the IIB product; please contact us directly if you would like a copy.

5.3 Configure TESTNODE_iibuser to work with DB2

If you have already done Lab 1 in this series, you can skip this section.

To run this lab, the Integration Bus node must be enabled to allow a JDBC connection to the HRDB database and create the necessary security credentials for TESTNODE_iibuser to connect to the database.

1. Open an IIB Command Console (from the Start menu), and navigate to

```
c:\student10\Create_HR_database
```

2. Run this command and accept the defaults:

```
3_Create_JDBC_for_HRDB
```

3. Run this command and accept the defaults:

```
4_Create_HRDB_SecurityID
```

4. Stop and restart the node to enable the above definitions to be activated

```
mqsistop TESTNODE_iibuser
```

```
mqsistart TESTNODE_iibuser
```

6. Prepare IIB REST Application

6.1 Import and deploy pre-requisite IIB projects

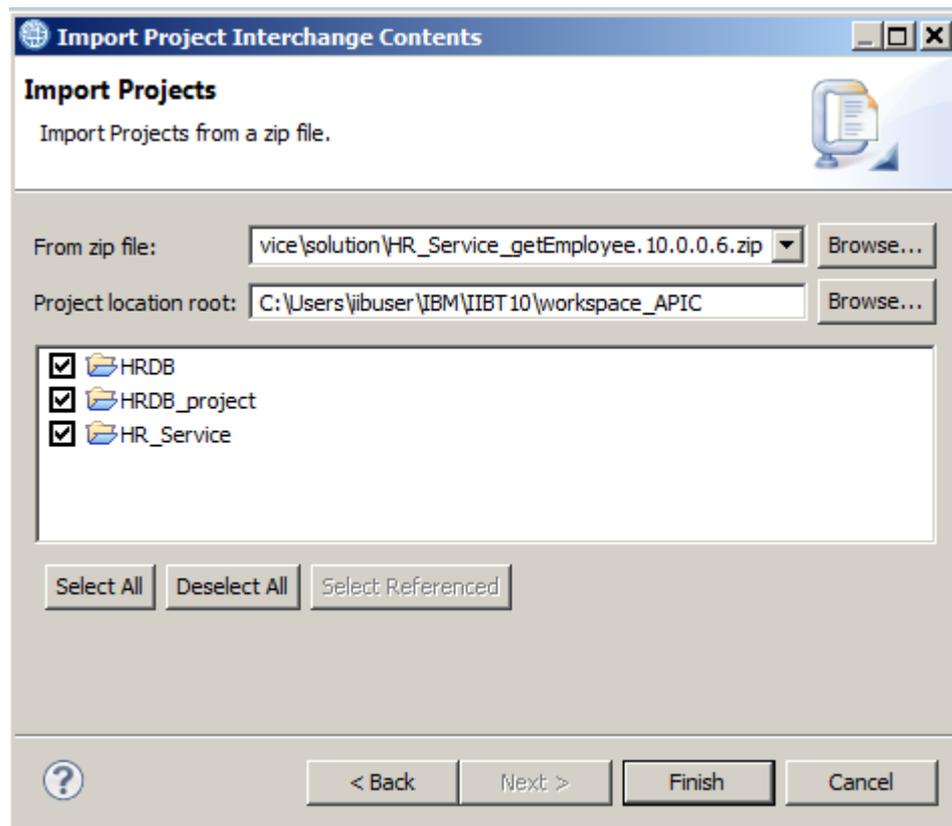
1. Start the IIB toolkit from the Start menu and to ensure there are no conflicts with other components, switch to a new workspace, e.g.

```
c:\users\iibuser\IBM\IIB 10\workspace_APIC
```

2. In the IIB Toolkit, import the Project Interchange file:

```
C:\student10\REST_API_HR_Service\solution\
HR_Service_getEmployee.10.0.0.6.zip
```

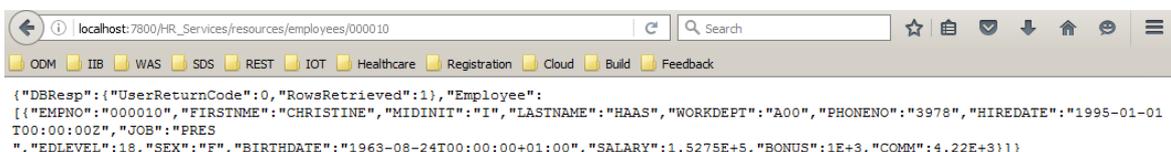
Import all 3 projects in the PI file.



3. Delete all deployed resources in the default server and then deploy first **HRDB** and then **HR_Service** to server **default** on TESTNODE_iibuser.
4. To test the REST API, use the following URL. A shortcut for this is provided in the REST folder in Firefox.

```
http://localhost:7800/HR_Services/resources/employees/000010
```

If the REST API is working correctly, the details for employee 000010 should be returned:



- Now change the URL, substituting the cloud host and port that you saved in section 3.3, for `localhost:7800`, so that the REST API can be accessed from outside the network. The format will be, e.g.

`http://<cloud host>:<cloud port>/HR_Services/resources/employees/000010`

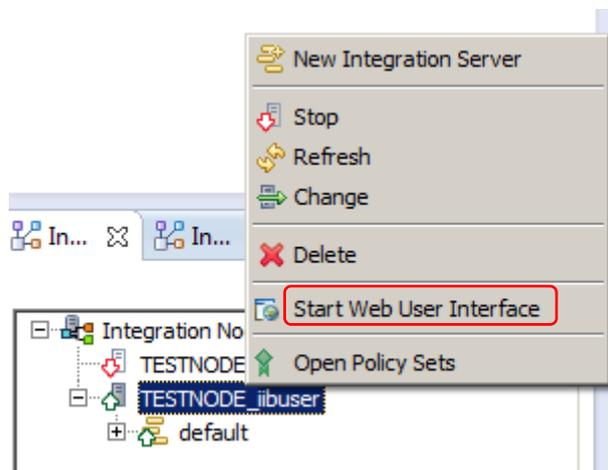
Paste the resulting URL into a browser and confirm that it returns the same details as before. This shows that the Secure Gateway connection is working and that API Connect on Bluemix will be able to reach this REST API.

6.2 Push REST API definition to IBM API Connect – Web UI

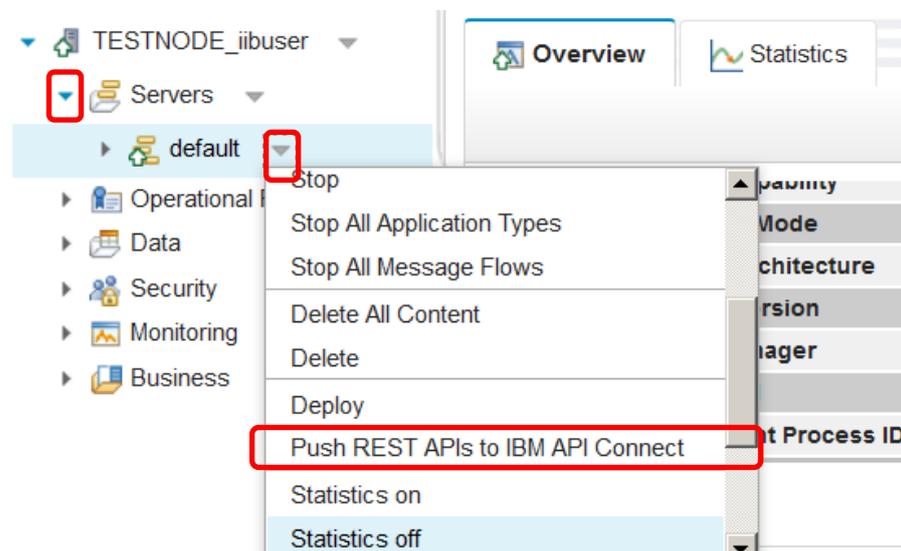
To push the REST API from definition using the Web User Interface, follow this section.

Alternatively, to push the definition from the Integration Console, skip this section and follow the next one instead.

- From the Integration Toolkit, start the WebUI for node TESTNODE_iibuser:



- Expand **Servers** under node TESTNODE_iibuser by clicking the twisty at the left and then click the down arrow to the right of the default server and select the option **Push REST APIs to IBM API Connect**.



3. The Push to API Management wizard will appear. Provide the following details:
 - Host. Enter one of the following, depending on the Bluemix region you are currently logged into:
 - **eu.apiconnect.ibmcloud.com** (United Kingdom)
 - **us.apiconnect.ibmcloud.com** (US South)
 - **au.apiconnect.ibmcloud.com** (Sydney)
 - Port: leave this as 443.
 - User ID: your IBM ID
 - Password: your IBM password

and then click **Connect to IBM API Connect**.

You should see a successful connection message as shown here.

Now click **Next**.

The screenshot displays a wizard window titled "Push REST APIs to IBM API Connect" with the subtitle "Define a connection to the IBM API Connect system". The form is divided into two sections: "Management Cluster / Server Address" and "Authentication".

Management Cluster / Server Address

- Host: eu.apiconnect.ibmcloud.com (highlighted with a red box)
- Port: 443

Authentication

- UserID: iibtester99@gmail.com (highlighted with a red box)
- Password: [Redacted with dots] (highlighted with a red box)

A "Connect to IBM API Connect" button is visible below the authentication fields. Below the button, a green checkmark icon is followed by the text "Successfully connected to IBM API Connect".

At the bottom of the wizard, there are four buttons: "Back", "Next", "Push to IBM API Connect", and "Cancel".

- In API Connect, a product is a set of APIs and Plans in one offering that you make available to your developers.

At the next screen, you are prompted to add a new product that will contain the REST API.

Give a title for the product.

The name and version will be generated for you.

Do not stage the product.

When the details have been entered, click **Next**.

Push REST APIs to IBM API Connect

Specify the product title, name, and version

Select the target organization

Organization

Specify the product title, name, and version

Title

Name

Version

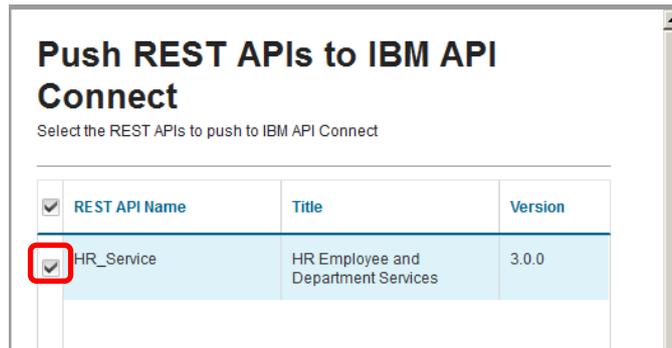
The draft product 'Employee' with version '1.0.0' will be created

Select the catalog where the product will be staged

Catalog

The product 'Employee' with version '1.0.0' will not be staged.

5. At the next screen, select the REST API, scroll down and then click **Next**.



Push REST APIs to IBM API Connect
Select the REST APIs to push to IBM API Connect

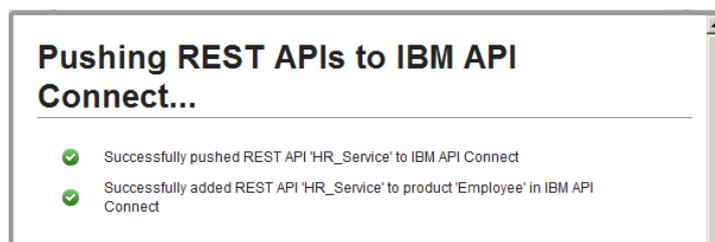
<input checked="" type="checkbox"/>	REST API Name	Title	Version
<input checked="" type="checkbox"/>	HR_Service	HR Employee and Department Services	3.0.0

6. At the next prompt, enter the cloud host name and port from the Secure Gateway connection (saved in section 3.3) in the appropriate places for the http connection.

These details will be used in the configuration of the REST API in API Connect on Bluemix, so that inbound connections are permitted to the IIB host system.

Click **Push to IBM API Connect**.

7. The push should be successful and a message displayed like the one shown below. Close the window when you are ready and skip the next section.



6.3 Push REST API definition to IBM API Connect – CLI

Do not perform this section if you have pushed the REST API definition using the Web UI.

To push the REST API definition using the Integration Console, follow this section.

We will use the **mqsipushapis** command, which has the following syntax:

```
mqsipushapis <integration node>
  -e <integration server>
  -t <API Connect host>
  -g <API Connect port>
  -u <API Connect user ID>
  -a <API Connect password>
  -o <API Connect organization>
    -r <product name>
    -k <API Name>
  -x <API host>:<API port>
```

Note:

- the **product name** specified with the **-r** flag is a name assigned by the user to the new product in API Connect that will contain the API.
- **API Connect host**: you will use one of the following, depending on the Bluemix region you are currently logged into:
 - **eu.apiconnect.ibmcloud.com** (United Kingdom)
 - **us.apiconnect.ibmcloud.com** (US South)
 - **au.apiconnect.ibmcloud.com** (Sydney)
- the **API Connect organization** is constructed in our case as follows: **<Bluemix Organization> (<Bluemix Space>)**. See section 3.1, if you don't have this information. Note the space between the two parts and that **Bluemix Space** needs to be placed in brackets, e.g, for a user with Bluemix Organization *iibttester99_workshop* and Bluemix Space *dev*, this would be: *iibttester99_workshop (dev)*.

1. To help you, we have provided a script, `deploy_to_APIConnect.cmd`, in `C:\student10\IIB_APIC\commands`.

In an Integration Console, change to this directory and run the command, entering appropriate values in the prompt. For a fictitious user, IBM ID `iibtester99@gmail.com`, in Bluemix Organization `iibtester99_workshop` and Bluemix Space `dev`, the output could look like this.

```
C:\student10\IIB_APIC\command>deploy_to_APIConnect.cmd
*****
*** This CMD file pushes an IIB REST API definition to API Connect. ***
*** Please provide values for each parameter at the prompt. ***
*** or allow the CMD file to use the following default values: ***
*****
IIB node name: "TESTNODE_iibuser"
IIB server name: "default"
IIB REST API: "HR_Service"

API Connect server hostname: "eu.apiconnect.ibmcloud.com"
API Connect server port: "443"
API Connect username: "betaworks_user@uk.ibm.com"
API Connect password: "xxxxxxx"
API Connect Organisation/space: "BetaWorks (dev)"
API Connect Title: "Employee"
API Connect Product: "Employee"
API Connect version for IIB REST API: "1.0.0"
API Connect Proxy url: "caplonsgprd-5.integration.ibmcloud.com:15002"
*****
Enter IIB node (default is "TESTNODE_iibuser"):
Enter IIB server where REST API is running (default is "default"):
Enter name of IIB REST API to be published (default is "HR_Service"):
Enter API-Connect server hostname (default is "eu.apiconnect.ibmcloud.com"):
Enter API-Connect server port (default is "443"):
Enter API-Connect username (default is "betaworks_user@uk.ibm.com"): iibtester99@gmail.com
Enter API-Connect password for user (default is "xxxxxxx"): passw0rd
Enter API-Connect Organisation and space (default is "BetaWorks (dev)"): "iibtester99_workshop (dev-uk)"
Enter API-Connect Title (default is "Employee"):
Enter API-Connect Product (default is "Employee"):
Enter API-Connect Version (default is "1.0.0"):
Enter name of HTTP Proxy for "HR_Service" (default is "caplonsgprd-5.integration.ibmcloud.com:15002"): caplonsgprd-3.integration.ibmcloud.com:15186
Thankyou, about to run the command below. Ok to proceed? Use Ctrl-C to terminate.

mqsipushapis "TESTNODE_iibuser" -e "default" -t "eu.apiconnect.ibmcloud.com" -g "443" -u iibtester99@gmail.com -a passw0rd -o "iibtester99_workshop (dev-uk)" -r "Employee" -d "Employee" -s "1.0.0" -k "HR_Service" -x caplonsgprd-3.integration.ibmcloud.com:15186

Press any key to continue . . .
```

This results in the following command being run:

```
mqsipushapis TESTNODE_iibuser
-e default
-t eu.apiconnect.ibmcloud.com
-g 443
-u iibtester99@gmail.com
-a passw0rd
-o "iibtester99_workshop (dev-uk)"
-r Employee
-d Employee
-s 1.0.0
-k HR_Service
-x caplonsgprd-3.integration.ibmcloud.com:15186
```

2. Check that the command runs successfully. If it does not, make corrections and try again. The response should look something like this:

```
BIP9356I: Successfully created API 'HR_Service' in IBM API Connect.
BIP9358I: Successfully created Product 'Employee' and added API 'HR_Service' to it in IBM API Connect.
BIP9366I: Product 'Employee' was not staged for Organization 'matthew_boult@uk.ibm.com (dev-uk)' in IBM API Connect as staging was not requested when the command was run.
BIP8071I: Successful command completion.
```

7. View REST API in IBM API Connect

For this section, you can remain logged in to Windows as the user "iibuser" or switch back to user "iibadmin" and continue with your running Bluemix session.

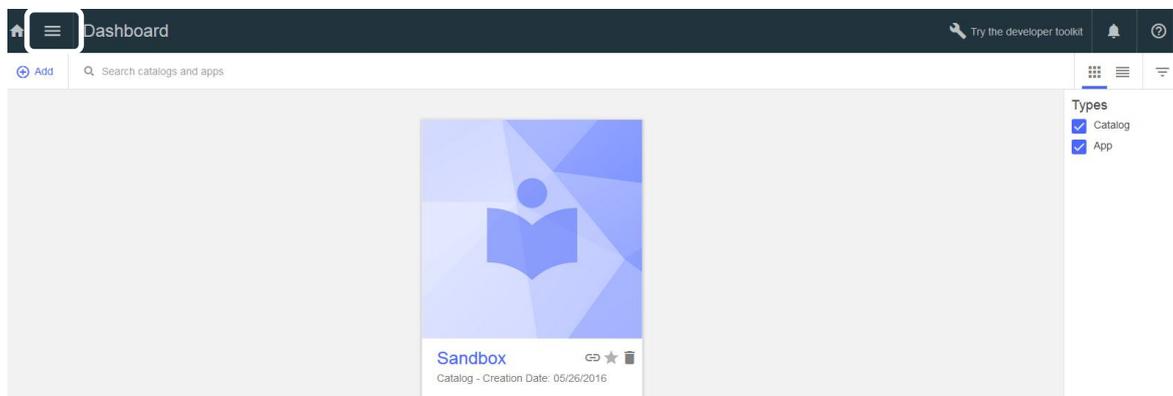
1. If you are still logged into API Connect from before, then jump to step 6.

Go to <https://new-console.eu-gb.bluemix.net/apis/apiconnect>

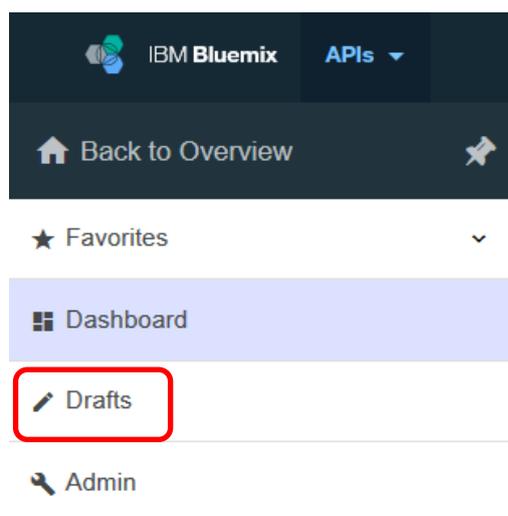
Log on with your IBM ID and password.

You are now logged on to IBM API Connect as an API manager.

2. Click on the menu symbol ()

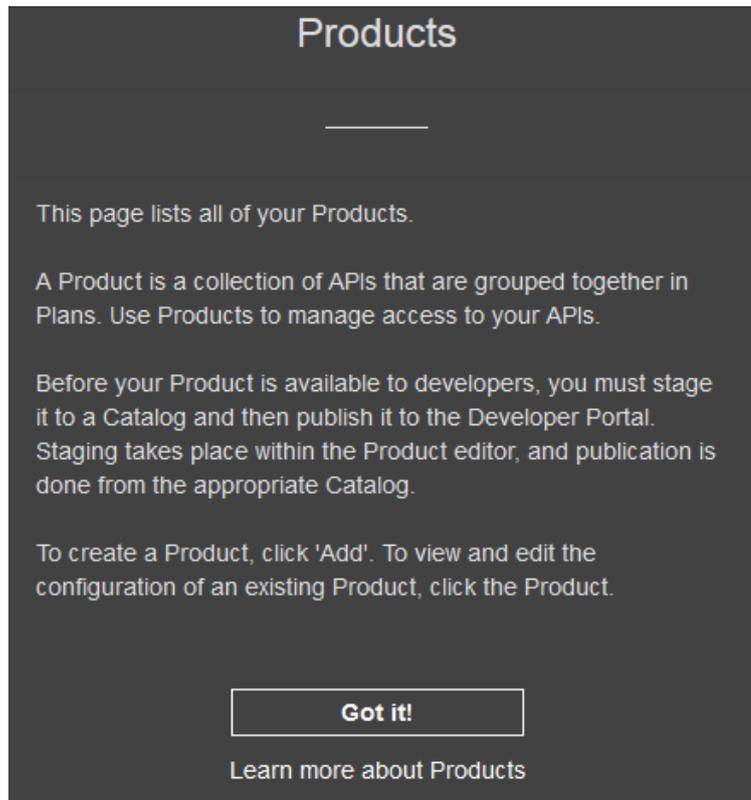


3. Then **Drafts**

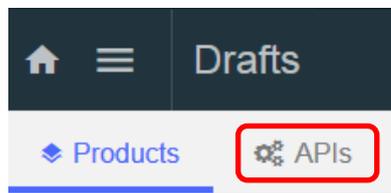


- You should now see the following screen with information on API Connect Products.

Click **Got it!** when you have finished reading it.



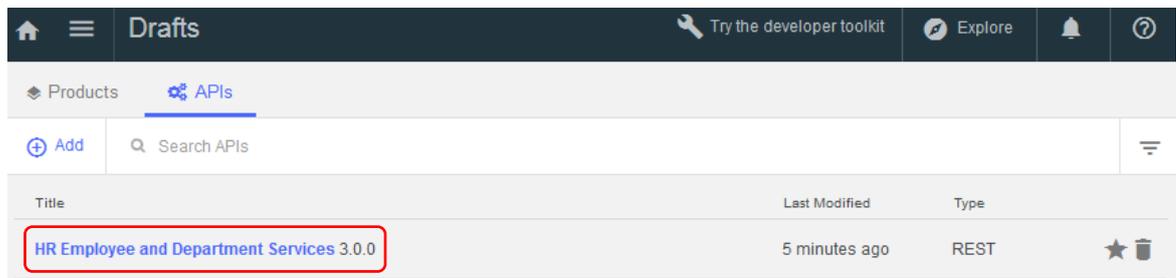
- At the next screen, click **APIs**.



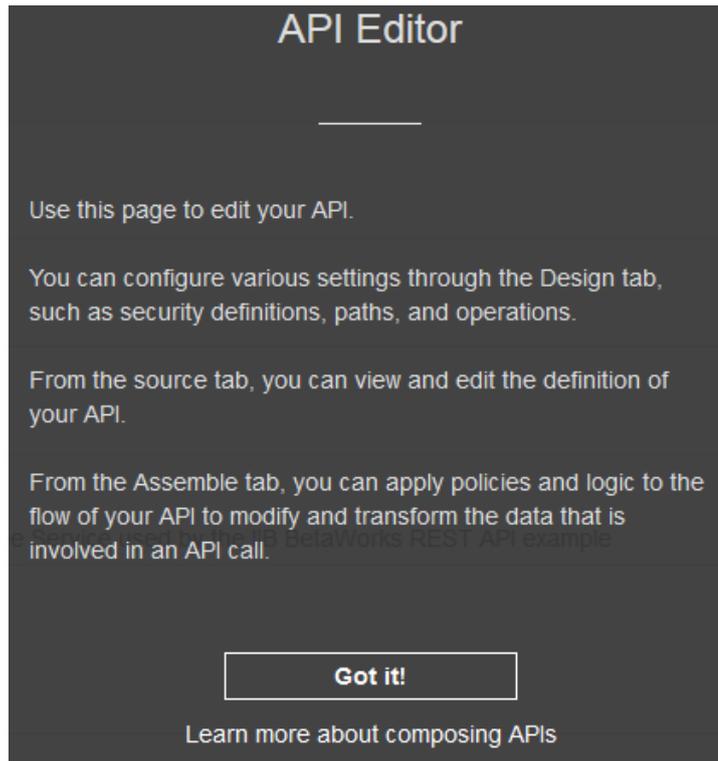
- If you were already logged on to API Connect from before, refresh the screen.

The **Employee Service** API should be displayed. This is the API that you pushed into IBM API Connect.

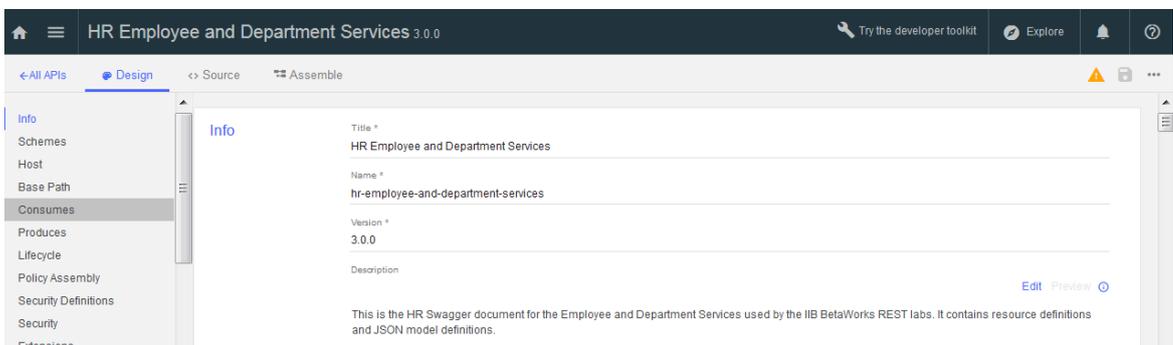
Click on **HR Department and Employee Services** to show the details of the API.



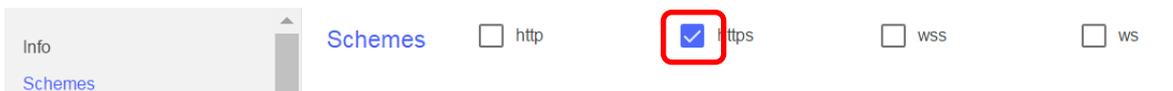
- 7. Before you get to the API Editor, you will see the following pop-up. Again, click **Got it!** after reading.



- 8. Explore the design of the API by navigating the left panel, or scrolling down the page at the right.

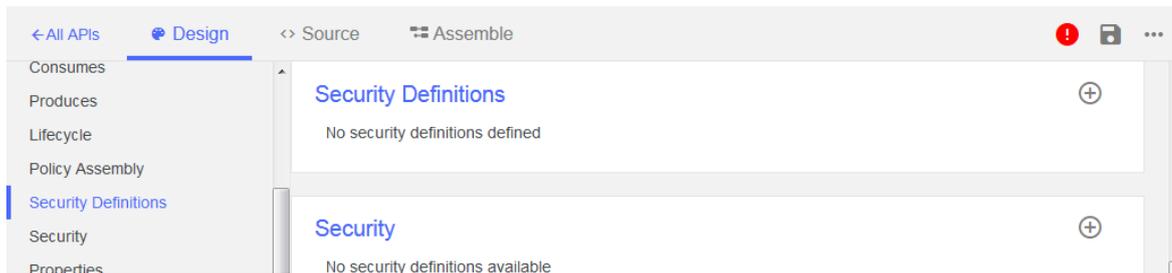


- 9. Navigate to the **Schemes** section under the **Design** tab and note the **https** tick box alone is selected, as shown below. This ensures the published URL will be secured using https.



10. Access to the REST API may be restricted by use of a Client ID and Client Secret which can be defined in the Security Definitions section and referenced in the Security section below it.

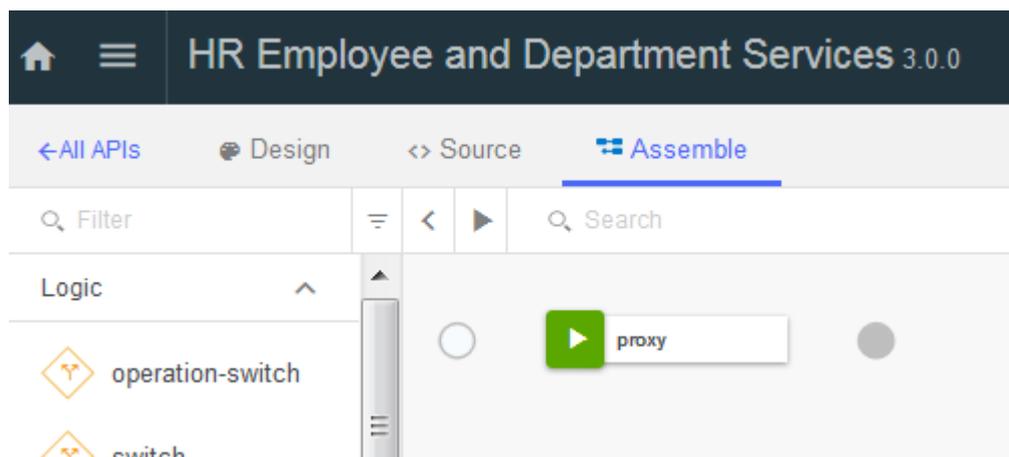
Scroll past these, as we shall not be using security definitions in this lab.



11. Click the **Assemble** tab.



12. You will see that an assembly flow has been created with a proxy node. Click on this node.

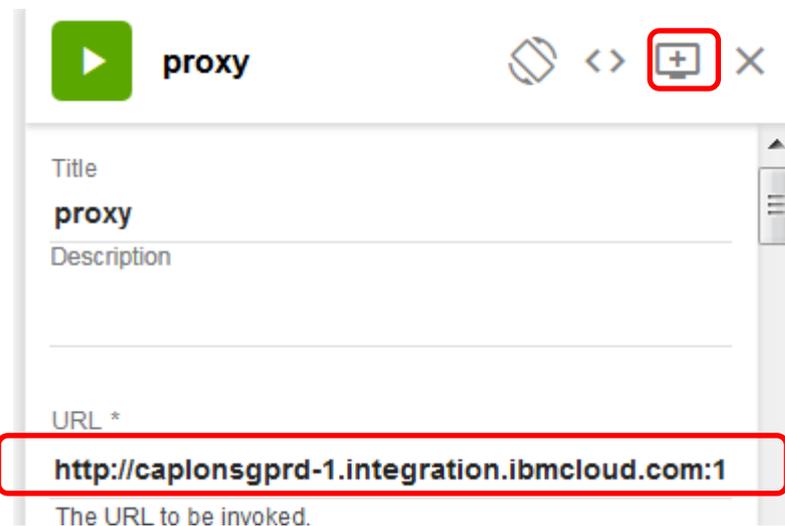


13. IBM API Connect must proxy the requests for the REST API into IBM Integration Bus. The details of the cloud host and port were given when the REST API was pushed from IIB and will be of the form:

`http://<cloud host>: <cloud port>$(request.path)`

To see the full URL, click the maximize symbol as shown.

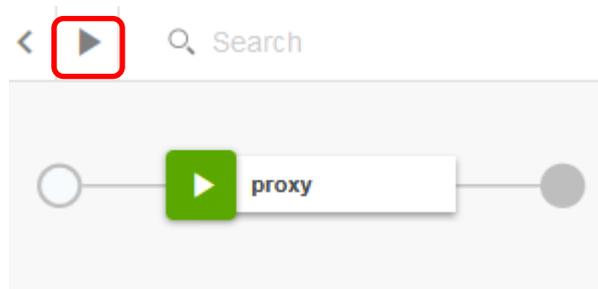
When you have finished, close the window.



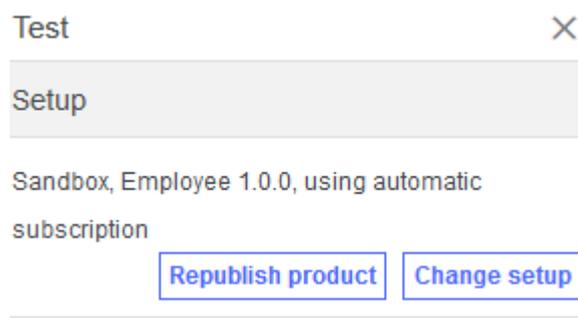
8. Test REST API

8.1 Set up and run the test

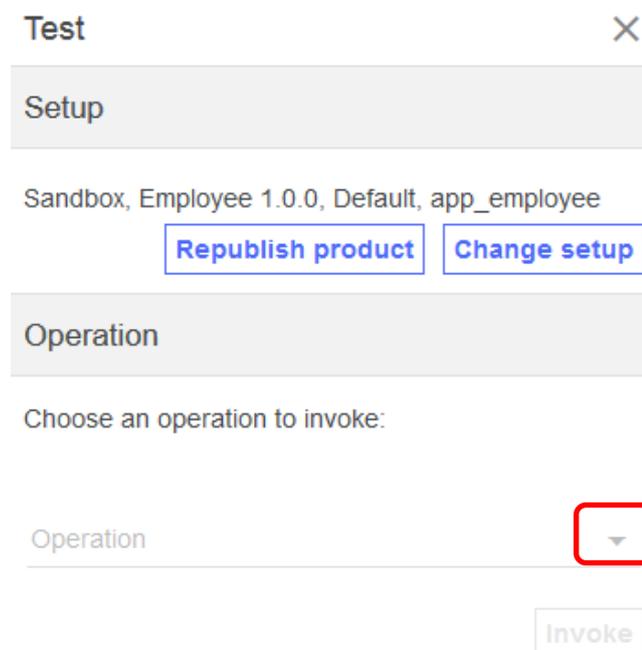
1. Click the arrow button to start the test.



2. In the navigation pane at the left, click **Republish product**.



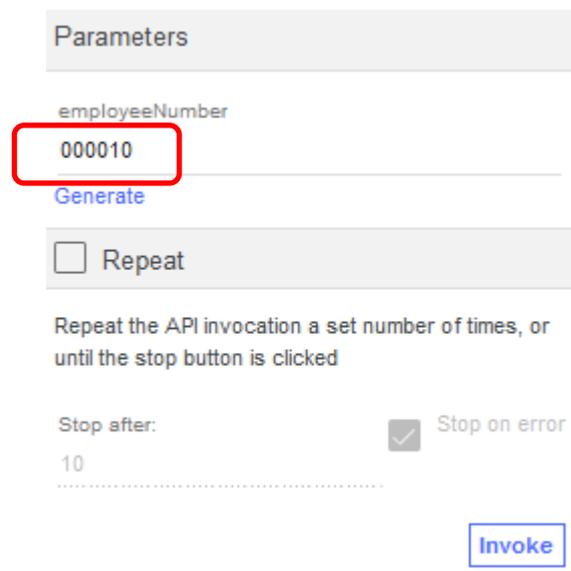
3. Now choose an operation to invoke by clicking the down arrow at the right.



4. Scroll down and choose the **get /employees/{employeeNumber}** operation.



5. Now, under the Parameters section, enter an employee number to retrieve, e.g.: 000010 and click **Invoke**.



6. If successful, data will be returned in the body of the response:

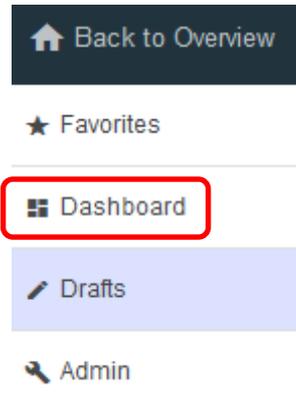


8.2 Determine URL of published REST API

1. Navigate to the Sandbox catalog by clicking on the menu icon ()

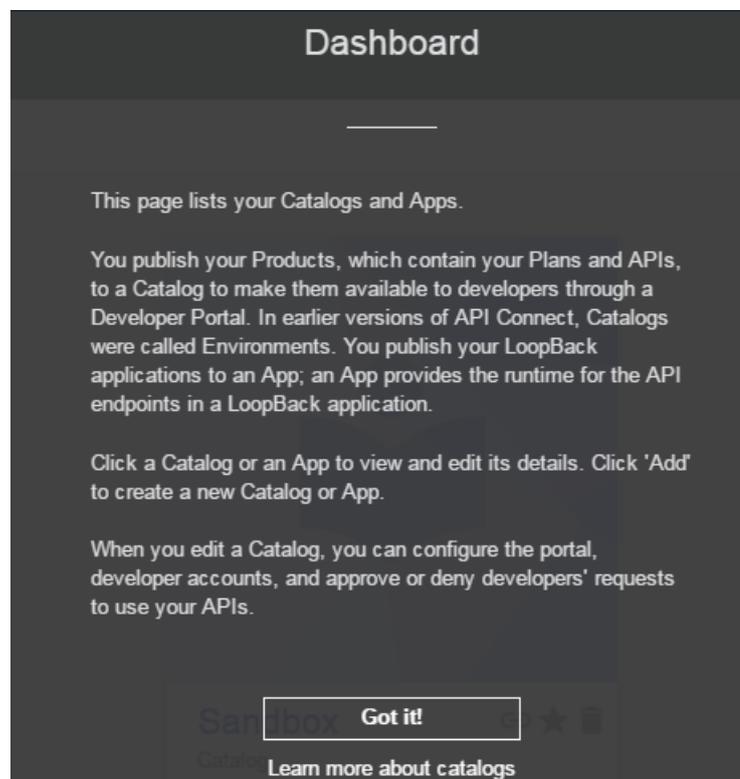


and then selecting Dashboard.

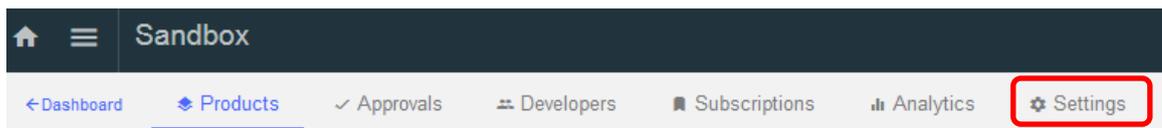


2. If this is the first time you have navigated here, you will see the following screen with information about the Dashboard.

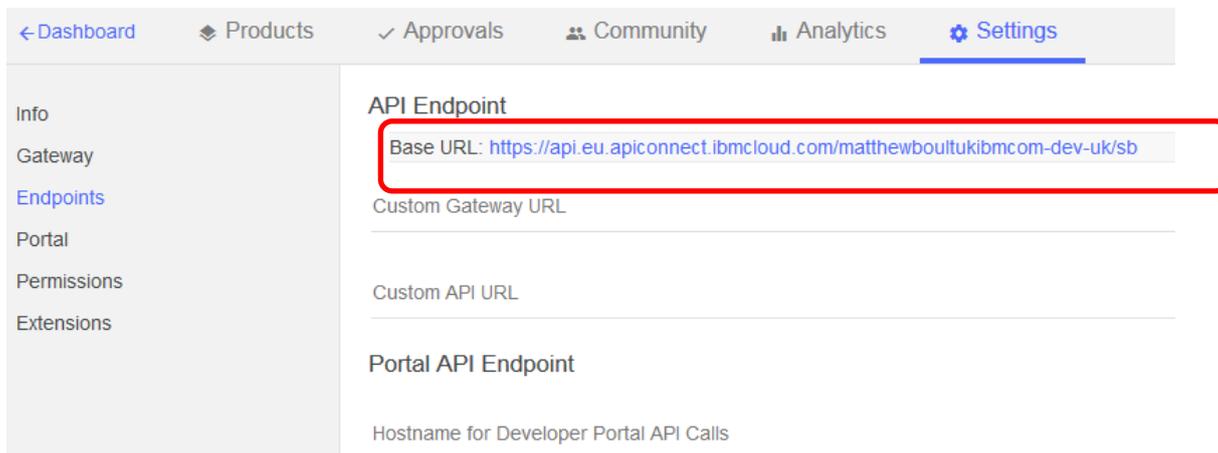
Scroll down, if necessary to Click **Got it!** when you have finished reading it.



- Click on the Sandbox icon and then **Settings**.



- Select the **Endpoints** option from the menu at the left and note the API Endpoint Base URL.



- The published URL will be of the form.

<API Endpoint Base URL>/HR_Services/resources/employees/{employeeNumber}

Construct a URL in this form, using your API Endpoint Base URL and a test employee number.

An example URL with a test employee number of 000020 is:

**https://api.eu.apiconnect.ibmcloud.com/iibttester99workshop-dev/sb/
HR_Services/resources/employees/000020**

Enter this URL in a browser and you should see a response showing the relevant employee's details:



You have now accessed the published REST API successfully as an API user.

END OF LAB GUIDE