Table of Contents

[Credentials 1](#_Toc25752329)

[Verify Initial Setup 1](#_Toc25752330)

[Import Template 2](#_Toc25752331)

[Create/Move/Read/Delete a TCP application (API) 2](#_Toc25752332)

[Create/Delete a TCP application (GUI) 4](#_Toc25752333)

[Additional Practice 5](#_Toc25752334)

[Other applications (HTTP/HTTPS) 6](#_Toc25752335)

# Credentials

**### Component Info & Credentials**

| Host | Management IP | Subnet 10 IP | Username | Password |

| ----------------- | ------------- | --------------- | -------- | -------- |

| BIG-IP | 10.1.1.11 | 10.1.10.40 | admin | admin |

| BIG-IQ CM | 10.1.1.5 | 10.1.10.17 | admin | Admin12# |

| Linux WebServer | 10.1.1.6 | 10.1.10.100-105 | ubuntu | key-only |

| Windows Jump Host | 10.1.1.7 | 10.1.10.200 | user | user |

# Verify Initial Setup

* Verify that the UDF deployment has started correctly (green).
* RDP into the Windows host using the credentials ‘user’ ‘user’
* Open ‘postman’ using the desktop icon, check that the following collections are present.
  + Sample BIG-IQ templates
  + Big-iq apps tcp
  + Big-iq apps http
  + Big-iq apps https offload
  + Big-iq apps https bridge
  + BIG-IQ API
* Verify that the environment is set to ‘sample apps’ (Top right of the screen)
* Open the environment (top right gear icon and click on sample apps). Verify that all values are populated (except ‘taskid’ need not be)
* Login into the BIG-IQ CM (IQ -7) using the credentials ‘admin’ ‘Admin12#’.
* Go to applications tab, then go to application templates (left side column). Verify that the ‘AS3 templates’ section is empty.

# Import Template

* Open the postman collection Sample BIG-IQ templates
* Go to Create a template (TCP) and observe its body. What fields does it have other than the actual schema? Go over this link for more details. <https://clouddocs.f5.com/products/big-iq/mgmt-api/v7.0.0/ApiReferences/bigiq_public_api_ref/r_as3_template.html>
* Click Send.
* Observe the response that you get.
* From the BIG-IQ, go to Applications -> Application templates. Do you see a new template? You should see ‘tcp-app’ template.
* Click on the template.
* You should see the following screen.
* A screenshot of a cell phone

  Description automatically generated
* Try to decipher the schema section. If you want to learn more about JSON schemas, please use this link - <https://json-schema.org/understanding-json-schema/>
* Click on Service\_TCP, what do you see? What are some things that are hard coded/ overridden? What is the thinking behind doing so? In certain objects you see “Use” and “Bigip” (e.g SNAT pool). What is the difference between these two options?
* Similarly, click on ‘Pool’ and observe what parameters have been overridden.
* Go to ‘Classes’ and click on Add/Remove. What do you see?
* Publish the template by selecting it and clicking publish.

# Create/Move/Read/Delete a TCP application (API)

Creating the application.

* Go to the collection big-iq apps tcp
* You should have 4 requests, Create App, get app status, check app config (read app), Delete app.
* Click on Create app. Click on the body section and observe the JSON body. Study the naming convention used. Notice that ‘app\_id’ field. What could have been the significance of this field? Also, click on the environment “sample apps” to see the names used to populate the variables inside the body.
* Click Send and observe the response. What is the significance of ‘id’ in the response? How can this field be used?
* Click on the request ‘check task status’ and click ‘send’. What is the response that you see (may vary depending upon when you check)?
* Go to BIG-IQ -> applications -> applications. What do you see?
* Click on unknown applications, what do you see? How is the application(configSet) named? Click on the application. Study the different parts of the screen. What can you see in the ‘configuration’ section? Are there things What do you see in the analytics section? What do you see in the middle of the screen? Click on each of the circles in the middle and observe.

Moving the application.

* When the applications are initially created using the API, they fall into the category of ‘unknown applications’. We will be moving it to a different globalApp using the API.
* Click on the app by clicking on ‘unknown applications’ and clicking on the application itself. From the browser’s URL bar, grab the ‘id’ of the ‘configSet’ of the application. You will see two ‘id’s’. The second one is the id of the app.
* 
* Click on the collection named ‘BIG-IQ API’ and click on the request named ‘Move/Merge apps’. In the body section, replace the ‘id’ by our applications ‘id’ (which we got above)
* A screenshot of a social media post

  Description automatically generated
* Click send and observe the response.
* Go to the BIG-IQ -> applications and look at the applications. Do you see a new application-name? Do you see our app inside? BIG-IQ provides this one level of hierarchy to organize our applications. What are some ways you can think of to leverage this feature?

Reading the application

* Go back to the ‘big-iq apps tcp’ collection. Click on the request named ‘check app config’ and click send.
* Observe the response. What did we do in our request to make sure that only our application is returned in the response?
* Login to the BIG-IP (admin,admin) and verify that our application exists there. Which partition has it been deployed?

Deleting the application

* Go back to the ‘big-iq apps tcp’ collection. Click on the request named ‘Delete app’ and click send.
* Check the status of the task
* Go to BIG-IQ -> applications. Verify that the application has been deleted. Is the empty app named ‘tcp\_AS3LAB\_apps’ still there?
* Login to the BIG-IP, verify that the application is deleted.

# Create/Delete a TCP application (GUI)

Creating the application

* From the GUI, go to Applications and click on ‘Create’
* A screenshot of a cell phone

  Description automatically generated
* Select Application name to be our existing application and template type to be ‘tcp-app’
* A screenshot of a cell phone

  Description automatically generated
* Provide the application service a name (no dots or dashes, can’t begin with number), e.g. ‘test\_app2’
* Select the target from the drop-down.
* Provide the tenant a name (no dots or dashes, can’t begin with number), e.g. ‘t\_1111’
* Enter server address (10.1.10.100 in the lab)
* Enter Virtual address (10.1.10.42-45 in lab)
* Click on create and wait for a few moments. Is your application created?
* Login to the BIG-IP and look at the configuration. Does your configuration look different from the one created using the API?

Deleting the application

* From the GUI, go to Applications -> tcp\_AS3LAB\_apps , select our application and click delete. Wait for a few moments.
* Verify that the application got deleted from the BIG-IQ and BIG-IP

# Additional Practice

* Your customer wants to use clone pools. For **every** TCP application, they want to mirror the client-side traffic and send it to the address 1.1.1.1. They already have a pool created on the BIG-IP named ‘/Common/clientside\_clone\_pool’. What modifications would you make to make this possible for all tcp applications?
* You currently are using base ‘tcp’ monitor for all your tcp applications. Your customer wants custom tcp monitor when creating each new application. He wants to be able to modify the interval and timeout for each application separately. What modifications would you make to make this possible?

# Other applications (HTTP/HTTPS)

* Use the collections provided, perform the above steps for HTTP, HTTPS offload and HTTPS bridge applications
  + Import template
  + Create/Delete (API)
  + Create/Delete (GUI)