00b Python Requests Core Server Concepts

When connecting to a remote SystemLink Server using native REST web service communication in python, you have 2 connection authentication options, either a valid username/password pair or a valid API key or Session key for the SystemLink Server with which you want to communicate.

If you know a valid username/password pair, you can send repeated requests to the same remote server url, appending the desired api route, and using the same username/password credentials each time

session = requests.Session()

url = "https://demo.systemlink.io/nitag/v2/tags-count"  
auth = ("lvadmin", "LabVIEW===")  
response = session.get(url, json={}, verify=False, auth=auth)  
print(str(response.json()))

If you can find or create an API key or Session key, you can send repeated requests to the same remote server url, appending the desired api route and using the same api\_key authentication each time

session = requests.Session()

url = "https://demo.systemlink.io/niuser/v1/workspaces"  
api\_key = "AU2hWR8LfzeeszVg1R0ZviS\_tlwmf8XjQF3JFGUkrG"  
headers = {'x-ni-api-key':api\_key, 'Content-Type':'application/json'}  
response = session.get(url, json={}, verify=False, headers=headers)  
print(str(response.json()))

You can look for an API key on the computer you’re running the python script, in this folder  
"C:\ProgramData\National Instruments\Skyline\HttpConfigurations\"  
Both the “http\_master.json” and the “http\_localhost.json” file have an “ApiKey” field

{"Id":"systemlink\_master","DisplayName":"Master","ConnectionType":"Master","Uri":"https://demo.systemlink.io/","ApiKey":"x-3ZRpGONcWUzEVKOmvX4mIS\_fzCP-4aB5ijAabPHR","CertPath":"http-client/http-client.cer"}

You can also run the below notebook code in a JupyterHub page of the SystemLink Server you want to communicate with—it will return a temporary session key you can use in your python code

api\_key = os.getenv("SYSTEMLINK\_API\_KEY")

