CIS3534C - Scripting for Network Professionals

Module 10

Application Programming Interfaces (APIs)

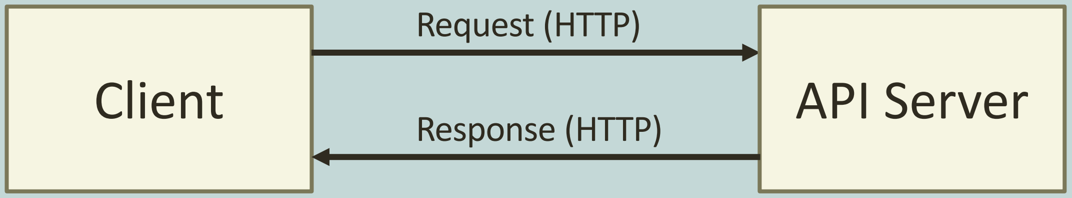
FSCJ
Florida State College at Jacksonville

APIs

- **Application Programming Interfaces** (APIs) are frequently provided that allow data to be accessed programmatically in various formats
- APIs can be accessed through Uniform Resource Locators (URLs), using a programming language with associated software libraries (Java, Python, JavaScript, R, etc), or using an application which has embedded the API in its feature set (e.g. Excel and PowerQuery)
- Data is retrieved in a **payload** which packages the data in a standard format, usually either **JSON** (JavaScript Object Notation) or **XML** (Extensible Markup Language)

REST Web Service APIs

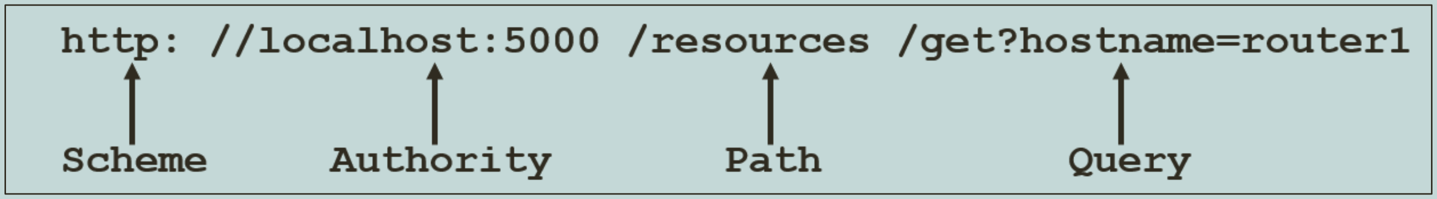
- There are various types of APIs available; one of the most commonly used API architectures for web applications is REST ("Representational State Transfer") which uses common HTTP web operations to obtain data



- REST API requests consist of the following components:
  - URI (Uniform Resource Identifier) (aka URL)
  - HTTP Method (e.g. GET, POST)
  - Header
  - Body

REST API Requests

- The URI identifies which resource the client wants to access.
- URI components are:
  - Scheme: the HTTP protocol (http or https)
  - Authority: host and port
  - Path: resource location on the server
  - Query: additional details for scope, clarity, or filtering



REST API Responses

- HTTP status codes are used to determine the success or failure of requests
    - 1xx – Informational
    - 2xx – Success
    - 3xx – Redirect
    - 4xx – Client Error
    - 5xx – Server Error

REST API Authentication

- REST APIs can require authentication to access data in order to prevent exposure of sensitive information or to prevent malicious behavior
- **Authentication** proves the client's identity
  - **Basic authentication**: credentials are transmitted as username/password pairs
  - **Bearer authentication**: uses a bearer token, a string generated by an identity service
  - **API Key**: a unique alphanumeric string generated by a server and assigned to a user
- **Authorization** determines the client's level of access

Flask and Flask_API

- Flask is a "micro web framework" written in Python
- Here's an example of a simple Flask application:

```
    from flask import Flask

    app = Flask(__name__)

    @app.route("/")
    def hello_world():
        return "<p>Hello, World!</p>"
```

- The @app.route() decorator tells Flask what URL should "trigger" the hello_world function
- Flask_API is an add-on module which provides API access to Flask

FSCJ
Florida State College at Jacksonville

The Python **requests** Module

- The requests module allows you to send client-side HTTP requests using Python
- Requests can be sent as HTTP **GET** or **POST** requests

  - GET requests include parameters appended in the URL, e.g.

```
response =
requests.get('http://localhost:5000/get?hostname=router1')
```

  - POST requests include parameters in the message body, e.g.

```
api_url = 'http://127.0.0.1:5000/set'
pload = {'token':token,
         'hostname':'router1',
         'attribute': 'ipaddr',
         'value':'127.10.10.1'}
r = requests.post(api_url,data = pload)
```

- Requests return a **Response Object** containing the response data

FSCJ
Florida State College
at Jacksonville

Python Virtual Environments

- Python 3 provides a virtual environment feature which allows a non-admin user to control their configuration without installing modules in obscure locations (e.g. a "roaming" folder)
- The **venv** module is included in the standard Python library on Windows (it has to be installed on the Linux version)

```
C:\>cd \Users\<userid>\Documents
C:\Users\<userid>\Documents>python -m venv myenv
```

- This creates a local folder named "myenv" which is used to store configuration information and packages as they are installed

Python Virtual Environments (Slide 1)

- The virtual environment must be activated after creating it by calling the activate script:

```
C:\Users\<userid>\Documents>myenv\scripts\activate
(myenv) C:\Users\<userid>\Documents>
```

- When you are done with the virtual environment, deactivate it by calling the deactivate script:

```
(myenv) C:\Users\<userid>\Documents>myenv\Scripts\deactivate
```

Florida State College at Jacksonville

Python Virtual Environments (Slide 2)

- The **(myenv)** prompt prefix indicates the virtual environment is active
- We can activate the environment in multiple windows
  - Other resources can be modified without activating the environment
- Use pip to list the installed packages (by default you will get pip and setuptools)

```
(myenv) C:\Users\<userid>\Documents>pip list
Package    Version
---------- -------
pip        20.2.3
setuptools 49.2.1
WARNING: You are using pip version 20.2.3; however,
version 22.3.1 is available.
```

FSCJ
Florida State College at Jacksonville

Python Virtual Environments (Slide 3)

- We can update pip (and install/update any other package) in the virtual environment without touching the system installation

```
(myenv) C:\Users\<userid>\Documents>python -m pip install --upgrade pip
Collecting pip
  Downloading pip-22.3.1-py3-none-any.whl (2.1 MB)
Installing collected packages: pip
...
Successfully installed pip-22.3.1

(myenv) C:\Users\<userid>\Documents>pip list
Package    Version
---------- -------
pip        22.3.1
setuptools 49.2.1
```

FSCJ Florida State College at Jacksonville

Installing Required Packages (Slide 1)

- Now we can install the necessary packages in the virtual environment to run Flask

```
(myenv) C:\Users\<userid>\Documents>pip install flask
  Collecting flask
  Downloading Flask-2.2.2-py3-none-any.whl (101 kB)

(myenv) C:\Users\<userid>\Documents>pip install flask_api
  Collecting flask_api
  Downloading Flask_API-3.0.post1-py3-none-any.whl (139 kB)

(myenv) C:\Users\<userid>\Documents>pip install requests
  Collecting requests
  Downloading requests-2.28.1-py3-none-any.whl (62 kB)
```

FSCJ
Florida State College
at Jacksonville

Installing Required Packages (Slide 2)

- Run pip list again to verify (you will also see the dependencies that were installed):

```
(myenv) C:\Users\<userid>\Documents>pip list
Package            Version
------------------ ---------
certifi            2022.9.24
charset-normalizer 2.1.1
click              8.1.3
colorama           0.4.6
Flask              2.2.2
Flask-API          3.0.post1
idna               3.4
importlib-metadata 5.0.0
itsdangerous       2.1.2
Jinja2             3.1.2
MarkupSafe         2.1.1
pip                22.3.1
requests           2.28.1
setuptools         49.2.1
urllib3            1.26.12
Werkzeug           2.2.2
zipp               3.10.0
```

FSCJ
Florida State College at Jacksonville

pip freeze (Slide 1)

- **pip freeze** also lists installed modules

```
(myenv) C:\Users\<userid>\Documents>pip freeze
certifi==2022.9.24
charset-normalizer==2.1.1
click==8.1.3
colorama==0.4.6
Flask==2.2.2
Flask-API==3.0.post1
idna==3.4
importlib-metadata==5.0.0
itsdangerous==2.1.2
Jinja2==3.1.2
MarkupSafe==2.1.1
requests==2.28.1
urllib3==1.26.12
Werkzeug==2.2.2
zipp==3.10.0
```

FSCJ
Florida State College at Jacksonville

pip freeze (Slide 2)

- pip freeze can be used to create a list of installed modules to recreate the configuration in a new virtual environment
- Record the current venv and deactivate:

```
(myenv) C:\Users\<userid>\Documents>pip freeze >requirements.txt
(myenv) C:\Users\<userid>\Documents>myenv\Scripts\deactivate
```

- Create a new venv and install packages from saved list

```
C:\Users\<userid>\Downloads>python -m venv newenv
C:\Users\<userid>\Documents>newenv\Scripts\activate
(newenv) C:\Users\<userid>\Documents> pip install -r requirements.txt
```

FSCJ
Florida State College at Jacksonville

pip freeze (Slide 3)

- Run pip freeze in the new environment to verify

```
(newenv) C:\Users\<userid>\Documents>pip freeze
...
Flask==2.2.2
Flask-API==3.0.post1
...
requests==2.28.1
...
WARNING: You are using pip version 20.2.3; however, version
22.3.1 is available.
```

- (notice that pip does not install the newer version from the original environment, it must be updated separately)

FSCJ
Florida State College
at Jacksonville