ITRI615 - Computer Security
Project Documentation

Affaan Muhammad - 33016763

Joshua Esterhuizen - 30285976

Contents

Section 1

Installation and setup

1.1 Project files

The project files can be found on the following GitHub link:
https://github.com/AM-ops/SecurityProject

This was our main code repository. We both have been updating the code as we went along and added details and bug fixes to the project.

To copy the code to your own machine, follow the following steps:

1. Make sure Git is installed. If not it can be downloaded from here:
   https://git-scm.com/

2. Create an empty directory where the code can be copied to

3. Run the following command:

```
git clone https://github.com/AM-ops/SecurityProject.git
```

1.2 Virtual Environment

There are multiple advantages of using virtual environments when creating software. The primary reason being we create a layer of separation and abstraction between our host machine's files and our software project.

We made use of a Python virtual environment which was handled by Anaconda. This can be downloaded from the following link:
https://www.anaconda.com/products/individual

1.2.1 Creating a virtual environment

Once Anaconda was installed the following commands were run in the terminal to create a virtual environment called `myDjangoEnv`.

```
conda create --name myDjangoEnv
```

Depending on the version of Anaconda installed you might have to use a leading underscore on Windows machines. The same will apply for commands further down. Below is a demonstration.

```
_conda create --name myDjangoEnv
```

1.2.2 Listing virtual environments

To list all virtual environments on your host machine run the following command.

```
conda info --envs
```

or

```
conda env list
```

1.2.3 Deleting a virtual environment

To delete a virtual environment run the following commands.

```
conda remove --name <name_of_virtual_environment> --all
```

or

```
conda env remove --name <name_of_virtual_environment>
```

1.2.4 Activating and deactivating virtual environments

To activate an environment run the following commands for Windows.

```
conda activate <name_of_virtual_environment>
```

For Linux and MacOS the command is as follows.

```
source activate <name_of_virtual_environment>
```

Once the environment is activated your terminal should change. By default, the active environment, is shown in parentheses () or brackets [] at the beginning of your command prompt as shown below.

```
(<name_of_virtual_environment>) >_
```

Depending on your version of Anaconda to deactivate your environment the commands for Windows is.

```
deactivate
```

or

```
conda deactivate
```

For Linux and MacOS the command will be

```
source deactivate
```

1.2.5 Listing Packages installed

To list all the packages you have installed in an environment there are two methods of listing them. First, if the environment is not activated run the following.

```
conda list -n <name_of_virtual_environment>
```

Secondly, if the environment is activated, then simply run the following.

```
conda list
```

1.2.6 Using pip

Due to the fact that Python is being used for the project it is always necessary to make sure pip is installed and functioning. If it is not then run the following commands.

```
conda install -n <name_of_virtual_environment> pip
```

1.3 Frameworks and other packages

1.3.1 Django

The primary framework used for development in this project was `Django`. This is a python based Web framework. The documentation for the it can be found here:
https://docs.djangoproject.com/en/3.2/

1.3.2 Bootstrap

Section 2

Sources

https://conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.
html
https://www.udemy.com/course/python-and-django-full-stack-web-developer-bootcamp/
https://docs.djangoproject.com/en/3.2/