
ATTACK
DEFENSE
by PentesterAcademy

Name	Assertion bypass
URL	https://www.attackdefense.com/challengedetails?cid=584
Type	Secure Coding : Python

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic.

A vulnerable python source code file (script.py) is given in the student home directory.

Objective: Execute the script in such a way that it prints "Congratulations, you have reached the unreachable code".

Solution:

According to the logic, the assert statement in the code should hit every time.

```
student@attackdefense:~$ cat script.py
def print_flag(show_flag):
    assert show_flag != False
    print("Congratulations, you have reached the unreachable code !!")

show_flag=False
print_flag(show_flag)
student@attackdefense:~$
```

Verify this by running the script

```
student@attackdefense:~$ python script.py
Traceback (most recent call last):
  File "script.py", line 6, in <module>
    print_flag(show_flag)
  File "script.py", line 2, in print_flag
    assert show_flag != False
AssertionError
student@attackdefense:~$
```

This happens because the __debug__ is always true by default when python script is executed. However, on disabling __debug__, the interpreter should bypass the assert statement.

The following command can be used to do that:

Command: python -O script.py

```
student@attackdefense:~$ python -O script.py
Congratulations, you have reached the unreachable code !!
student@attackdefense:~$
```