Demonstration

* To set up the environment to run the code and execute the attack I downloaded PyCharm which is an integrated development environment (IDE) specifically designed for Python programming.
* I also installed the soco library, which is a Python package that provides a high-level interface for controlling Sonos speakers and devices.
* "SoCo" stands for "Sonos Controller" and it allows you to interact with Sonos devices on your local network
* Once we had everything installed, we needed to search the network to find the Sonos device that we were going to attack.
* Once I figured out the subnet and the ports that the Sonos bar used, I used Nmap to scan and find the Sonos bar and to see if the port is open. I used Nmap -p 1400 192.168.1.0/24 to find the device.
* I have already run the scan and as you can see, here is the Sonos Device and we can see the port is open.
* Now that we know that the port is open, we will be able to execute our attack.
* Now it is time to write a script that will allow us to take control of the device.
* We needed to write a script that would not only stop the device from playing but continue to run the command so that the Sonos bar was inoperable.
* usings the socos help you can find a list of commands to control the sonos device.
* This means that anyone that would have the app on the phone or other device would not be able to control the device. On the user side it would seem that the device is not working.
* Using documents for the Soco library, that were already created, we were able to write code to execute the attack
* We used a while true statement that would create an infinite loop to execute our DoS attack.
* The line if device is not None: checks if a Sonos device was discovered. If the condition is true (i.e., a device is found), the code block inside the if statement is executed. Otherwise, if no device is found, the code jumps to the else block.
* The line while True: starts an infinite loop that will continue indefinitely until the program is interrupted or terminated. This means that the code inside the loop will be executed repeatedly and for our case it will continue to stop the device from playing.

**Testing the Vulnerability**

* Now that we have a code written, that should act out a DoS attack so let’s try it!
* I will go into my Sonos app and start playing some music. I will then run the script and you will see the DoS attack in progress. Once the music has stopped I will explain what is happening.
* As you can hear the music has stopped and DoS attack is in progress
* I will try to hit play in the app on my phone and you will see the command pause for a second and then go back to continuing the stop action.
* On the user side they hit play and it goes back to pause. The user will see nothing except that the Sonos bar is not working.
* As soon as I hit stop the DoS attack stops and the Sonos bar can go back to being used as normal.