1st_function_probe_port-Socket (For this code)

In the provided Python script, the `socket` module is used to create and manage network sockets. Sockets are a fundamental concept in network programming and allow communication between processes over a network. Here's how the `socket` module is used in this script:

1. **Socket Creation:**

- `sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)`
  - `socket.AF_INET`: This specifies the address family, in this case, IPv4.
  - `socket.SOCK_STREAM`: This specifies the socket type, which is a TCP socket for stream-oriented communication.
- **Setting Timeout:**
- `sock.settimeout(0.5)`
  - This line sets a timeout of 0.5 seconds for the socket connection attempt. If the connection cannot be established within this time, the socket operation will raise an exception.
- **Connecting to a Remote Host:**
- `r = sock.connect_ex((ip, port))`
  - The `connect_ex` method is used to attempt a connection to the specified IP address (`ip`) and port number (`port`).
  - If the connection is successful, `r` will be set to 0. If the connection fails, it will return an error code that indicates the reason for the failure.
- **Closing the Socket:**

1. `sock.close()`
   - After attempting the connection and obtaining the result, the socket is closed using the `close` method. This releases the resources associated with the socket.
2. **Exception Handling:**

- The `probe_port` function is wrapped in a try-except block to handle any exceptions that might occur during the socket operations. If an exception is caught, it is ignored (`pass`).

3. **Reuse of Socket for Each Port:**
    - The script creates a new socket (`sock`) for each port it attempts to probe. This is a common approach in simple port scanning scripts.

The primary purpose of the `socket` module in this script is to facilitate the creation of TCP sockets for attempting connections to different ports on the specified IP address. The result of these connection attempts is then used to identify whether a particular port is open or closed.