SOP 3: Custom Python Tool SOP

Authorship: Ethan Pham

Objective: To provide instructions on developing, executing, and documenting a custom Python tool.

Tools Needed:

- Kali Linux
- Python 3.x

Steps:

1. **Setup Development Environment:**
    - Open a terminal on Kali Linux.
    - Ensure Python 3.x is installed: python3 --version
2. **Develop the Custom Python Tool:**
    - Create a new Python script: nano custom_tool.py
    - Example script for simple port scanning: import socket
      def scan_ports(ip): n_ports = [] for port in range(1, 1025): s = socket.socket(socket.AF_INET, socket.SOCK_STREAM) s.settimeout(1) result = s.connect_ex((ip, port)) if result == 0: open_ports.append(port) s.close() return open_ports
      if **name** == "**main**": target_ip = "192.168.1.10" ports = scan_ports(target_ip) print(f"Open ports on {target_ip}: {ports}")
3. **Execute the Custom Python Tool:**
    - Run the script: python3 custom_tool.py
4. **Documentation:**
    - Take screenshots of the script and its execution output.
    - Document the purpose, functionality, and results of the custom tool.

Output:

- Successful execution of the custom Python tool.
- Screenshots and documentation of the tool’s purpose and functionality.