CE 340 Cryptography & Network Security Assignment 2

Title: Building a simple pentest tool
Defined by: Süleyman Kondakcı (Instructor)
Date to start: 05.12.2022
Date to deliver: 22.12.2022, 17:00
Project members: Max. 2 students

You will write a script (pentest.py) containing a set of **Python** functions (tasks), which will be invoked via a main menu. This script is your source file, which should be executed on a command console without use of any Python IDE. Your script will contain the tasks that are explained in the below table. When you execute the script, a menu will be displayed from which you will choose a task to perform.

ICMP ping	Ping an IP range and collect IP addresses of the hosts that are alive and save the result in a text file, call this **icmp.dat**.
Port identification	1) Get the IP addresses from the **imcp.dat** file and scan and validate these IP addresses. If an IP address is a valid live host address append it to a string or list (live hosts) that will contain the IP addresses of the live hosts. A live host is an active host that can be monitored by **nmap** or **Wireshark**. 2) Now perform port scan on the live hosts. The scanning must find and identify ports on each host and save the results into a text file, call this **openPorts.dat**. This text file will contain Host IPs, ports numbers, and service/application names (if any) of each port.
OS Fingerprint identification	This function will get the host IPs from the text file (**openPorts.dat**) and identify operating systems (OS) and OS versions of the hosts.
Web server detection	Scan the Internet and discover 10 web-server addresses, protocols, and ports of each web server. Save the result into text file, call this **web.dat**.
SYN_flood	This function will launch SYN-flood attack to a given destination (IP) and port(s). This tool must also enable you to choose the number of flooding, e.g., 10.000 SYN attacks. While running the SYN-flood attack use some Python codes to capture and decode TCP and IP packet headers and save them in a text file called **SYNresults.txt**. If you want, you can also use **Wireshark** or **tcpdump** to capture the attack packets and save the packets to a **pcap** file. Thereafter, use a Python code to decode and save the results.
Show	This function will ask and display the contents of the files that your tools have created so far.

What to upload on Blackboard?

Source files, User guide, and execution trace (e.g., screenshots).
Make sure that you can present the project in the classroom.

Good Luck! *S. Kondakci*