PSG College of Technology

Department of Applied Mathematics and Computational Sciences

18XW46 - Computer Networks and TCP/IP Lab

PACKAGE DOCUMENTATION

Team Members:

19PW13 - Madhumitha S

19PW27 - Sakthi S

Problem Statement:

The package deals with Data-exfiltration using ICMP tunneling with analysis of .pcap file.The application aims to send files from one system to another using ICMP ping packets which pings messages to single or multiple hosts.

Abstract:

The ICMP protocol is a pretty simple layer 3 protocol and does not contain a lot of information. However, it includes a data field used notably in error messages. This data field can also be used for creating an ICMP tunnel between two remote computers where hosts inject data into icmp echo packets. This method is often used to bypass firewall rules in the aim of exfiltrating data. Such process can be detected by analyzing ICMP echo packets' size.

Here, the transfer of file content is from the host to destination IPs that are scanned by a user defined program *Network_scanner.py*.

The application first converts the contents of the file(an .exe, image, document etc) to base64 encoded text. This will then send the ping requests each with 64 characters of data taken from the base64 encoded text file.

The application needs a packet capture software on the other side to capture and record all the ping packets as a .pcap file. Use the *parser.sh* to quickly parse the .pcap file and obtain the text file with base64 encoded data.

Once this is done, base64 decoding is used to get back the transmitted text file into respective file format and view original content of message .

The application also analyses the previously mentioned .pcap file and graphically interprets the relation between the number of packets sent and the destination IPs.

Contributions:

1. **19PW13 Madhumitha S:**

   Developed programs that Scan the network for IPs (*Network_scanner.py*), parse the .pcap file to retrieve data and perform base64 decoding (*parser.sh*), analyse the .pcap file *(analysis.py*) and some modules in the ICMP exfiltrate program namely *receive_ping* and *verbose_ping* (these functions deal with the receiving and displaying of the ping echo request).

2. **19PW27 Sakthi S:**

   Setting up the host , creating a ICMP echo request packet, sending the file, error detection during transmission using checksum and

pinging the clients with the packet.(*create_packet, do_one, checksum, PingQuery, multi_ping_query*)

Tools used:

1. Wireshark - for packet sniffing
2. scapy - for scanning purposes

References:

1. scapy :https://scapy.readthedocs.io/
2. Asyncore : https://docs.python.org/3/library/asyncore.html
3. plotly: https://plotly.com/