***Name :- Ranjan Kumar***

***Roll no. :- 2201CS59***

**Assignment 4:**

**Objective 1: Tracert Utility Analysis**

**1. Tracert Basics**

**Purpose of the Tracert Utility:**

- The tracert (short for "trace route") utility is a network diagnostic tool used to trace the path that packets take from your computer to a destination host. It provides a list of hops (routers) between your computer and the target, helping to identify where delays or failures occur.

**Basic Syntax:**

- The basic syntax of the tracert command is:

  tracert [options] target_host

  - target_host: The domain name or IP address of the destination.

**Examples:**

- To trace the route to a website (e.g., google.com):

  Command :- tracert google.com

```
C:\Users\Ranjan Kumar>tracert google.com

Tracing route to google.com [142.250.193.14]
over a maximum of 30 hops:

  1    29 ms     7 ms    15 ms  10.15.6.1
  2     6 ms     1 ms    15 ms  172.29.1.17
  3     6 ms    38 ms    17 ms  172.16.0.22
  4    19 ms     6 ms    31 ms  ws240-251-252-122.rcil.gov.in [122.252.251.241]
  5    72 ms    69 ms    43 ms  ws197-251-252-122.rcil.gov.in [122.252.251.197]
  6    13 ms    15 ms     *     172.31.251.85
  7   131 ms     *       33 ms  172.31.251.84
  8   141 ms     *        *     136.232.74.101
  9     *        *        *     Request timed out.
 10    44 ms    94 ms    69 ms  10.119.234.162
 11    94 ms    99 ms    89 ms  72.14.195.22
 12    96 ms    72 ms   106 ms  72.14.234.223
 13    65 ms    64 ms    66 ms  142.251.54.87
 14    99 ms    44 ms    49 ms  del11s14-in-f14.1e100.net [142.250.193.14]

Trace complete.
```

### To trace the route to a local host:

Code :- tracert 127.0.0.1

```
C:\Users\Ranjan Kumar>tracert 127.0.0.1

Tracing route to LAPTOP-9UNCUVUF [127.0.0.1]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  LAPTOP-9UNCUVUF [127.0.0.1]

Trace complete.
```

**2. Tracert Output Analysis**

**Running Tracert:**

- **Command:** tracert google.com
- **Sample Output:**
-

css

Copy code

Tracing route to google.com [142.250.193.238] over a maximum of 30 hops:

**Output Explanation:**

- **Hop Number:** The sequence number of the router the packet passes through.
- **IP Address:** The IP address of the router.
- **RTT (Round-Trip Time):** The time it takes for a packet to go from the source to the destination and back, measured in milliseconds.

**Local Host Tracert:**

- **Command:** tracert 127.0.0.1
- **Output :**
- **Output Explanation:**
    - Since 127.0.0.1 is the loopback address, the output will typically show just one hop with minimal RTT.

**3. Tracert Options**

**-d (Do not resolve hostnames):**

- **Description:** This option prevents the tracert utility from resolving IP addresses to their corresponding domain names, which can speed up the trace process.
- **Example:** tracert -d google.com
- **OUTPUT**

```
C:\Users\Ranjan Kumar>tracert -d google.com

Tracing route to google.com [142.250.206.142]
over a maximum of 30 hops:

  1    13 ms    55 ms     4 ms  10.15.6.1
  2     2 ms     2 ms     2 ms  172.29.1.17
  3    29 ms    52 ms    63 ms  172.16.0.22
  4    30 ms    54 ms    13 ms  122.252.251.241
  5   130 ms   330 ms   333 ms  10.118.248.49
  6   354 ms    40 ms     *     172.31.251.85
  7     *       20 ms     *     172.31.251.84
  8     *       21 ms     *     136.232.74.101
  9     *        *        *     Request timed out.
 10   365 ms   408 ms     *     10.119.234.162
 11    77 ms    79 ms    66 ms  74.125.147.192
 12    64 ms    84 ms    50 ms  192.178.80.159
 13    88 ms    57 ms    58 ms  142.251.76.197
 14    84 ms    64 ms    83 ms  142.250.206.142

Trace complete.
```

**-h (Maximum number of hops):**

- **Description:** This option allows you to set the maximum number of hops (routers) to be traced before the utility stops.
- **Example:** tracert -h 5 google.com
- **OUTPUT**

```
C:\Users\Ranjan Kumar>tracert -h 5 google.com

Tracing route to google.com [142.250.195.14]
over a maximum of 5 hops:

  1     9 ms    14 ms     3 ms  10.15.6.1
  2   117 ms     8 ms     2 ms  172.29.1.17
  3   932 ms  3296 ms   144 ms  172.16.0.22
  4     5 ms    81 ms   109 ms  ws240-251-252-122.rcil.gov.in [122.252.251.241]
  5   407 ms   984 ms   417 ms  ws197-251-252-122.rcil.gov.in [122.252.251.197]

Trace complete.
```

**-w (Timeout in milliseconds):**

- **Description:** This option sets the wait time in milliseconds for each reply before moving on to the next hop.
- **Example:** tracert -w 500 google.com
- **OUTPUT**

```
C:\Users\Ranjan Kumar>tracert -w 500 google.com

Tracing route to google.com [142.250.195.14]
over a maximum of 30 hops:

  1     *        3 ms     3 ms  10.15.6.1
  2     6 ms    13 ms     3 ms  172.29.1.17
  3     5 ms    12 ms    83 ms  172.16.0.22
  4     8 ms     8 ms    12 ms  14.139.194.1
  5    13 ms     4 ms    17 ms  ws197-251-252-122.rcil.gov.in [122.252.251.197]
  6     *       29 ms     *     172.31.251.85
  7    86 ms    18 ms   282 ms  172.31.251.84
  8     *        *      182 ms  136.232.74.101
  9     *        *        *     Request timed out.
 10     *       33 ms    30 ms  10.119.234.162
 11    87 ms    55 ms   101 ms  72.14.195.22
 12    60 ms    91 ms    83 ms  142.251.226.85
 13    65 ms    48 ms    62 ms  142.251.52.213
 14   136 ms    85 ms     *     del12s09-in-f14.1e100.net [142.250.195.14]
 15    50 ms    67 ms    92 ms  del12s09-in-f14.1e100.net [142.250.195.14]

Trace complete.
```

**4. Troubleshooting with Tracert**

**Scenario:**

- **Problem:** A user is experiencing slow network speeds when accessing a particular website.
- **Using Tracert:**
  - **Command:** tracert google.com
  - **Analysis:** The tracert output can show if there is a specific hop that is causing delays, indicating a possible network bottleneck or misconfiguration at a specific router.

**Options to Use:**

- **-h:** To limit the number of hops traced if the destination is known to be within a few hops.
- **-d:** To speed up the process by skipping hostname resolution.

**5. Conclusion**

**Summary:**

- The tracert utility is a powerful tool for network diagnostics, helping identify where delays or failures occur along a packet's route to its destination.

**Limitations:**

- Tracert may not work effectively if ICMP traffic is blocked by routers, or if the destination is unreachable, leading to incomplete or misleading results.

---

**Objective 2: Scapy-based Tracert Utility**

**1. Basic Functionality**

**Testing:**

- Ensure the provided Scapy-based tracert code works with various inputs, such as different destination IPs, max TTL values, packet sizes, timeouts, and source IPs.

**2. Additional Features**

**Implementation:**

- **Number of pings per hop:**

  ping_per_hop = 3  # Number of pings

- **Delay between pings:**

  delay_between_pings = 0.5  # Delay in seconds

- **Save output to a file:**

  with open("tracert_output.txt", "w") as file:

  file.write(output)

**3. Error Handling**

**Try-Except Blocks:**

- **Invalid Destination IP:**

  try:

  ip = socket.gethostbyname(destination)

  except socket.error:

  print("Invalid IP address.")

- **Invalid Max TTL Value:**

  if not (1 <= max_ttl <= 255):

  raise ValueError("Invalid TTL value.")

- **Invalid Packet Size:**

  if packet_size < 0:

  raise ValueError("Packet size must be a positive integer.")

**4. Output Formatting**

**Improved Output:**

- **Example:**

  print(f"Hop {hop}: {ip} | RTT: {rtt:.2f} ms | Loss: {loss}%")

**Submission**

- **Modified Code:**
  - Include the updated Scapy-based tracert code with the additional features.
- **Brief Report:**
  - Describe the new features, error handling approach, and sample outputs.