The ICMP Pinger Lab is a computer networking project that involves implementing a basic ping application using the Internet Control Message Protocol (ICMP). In this lab, you will create a simple program that sends ICMP Echo Request messages (pings) to a remote host and measures the round-trip time for each ping. This project helps you understand the fundamentals of ICMP and network latency measurement.

**Project Title: ICMP Pinger Lab in Python**

**Project Description:**

**Overview:**

The goal of this project is to implement a basic ICMP ping application. The program sends ICMP Echo Request messages to a specified remote host and measures the time it takes for each message to travel to the host and receive a corresponding Echo Reply.

**Components:**

1. **ICMP Ping Client:**
   * The client sends ICMP Echo Request messages to a specified remote host.
   * Measure the round-trip time for each ping.
   * Implement a timeout mechanism to handle cases where the host does not respond.
2. **ICMP Ping Server (Optional):**
   * Implement a simple ICMP server that listens for incoming Echo Request messages and responds with Echo Replies.
   * The server may optionally introduce random delays to simulate network variability.
3. **Round-Trip Time Calculation:**
   * Calculate and display the round-trip time for each ping.
   * Measure the time elapsed between sending a ping and receiving the corresponding reply.
4. **Statistics:**
   * Track statistics such as average round-trip time and packet loss.
   * Display these statistics at the end of the ping session.

**Requirements:**

1. **Programming Language:**
   * Use Python for both the client and server implementations.
2. **ICMP Protocol:**
   * Utilize the scapy library or another suitable library for sending and receiving ICMP packets.
3. **Ping Client:**
   * Implement the client program to send ICMP Echo Request messages and measure round-trip times.
4. **Ping Server (Optional):**
   * Implement an optional server that responds to ICMP Echo Request messages.
5. **Round-Trip Time Calculation:**
   * Calculate and display the round-trip time for each ping.
6. **Statistics Calculation:**
   * Compute and display statistics such as average round-trip time and packet loss.

**Sample Workflow:**

1. The user runs the ICMP ping client script, specifying the remote host's IP address or domain name.
2. The client sends a series of ICMP Echo Request messages to the remote host.
3. The client measures the round-trip time for each ping.
4. The client displays statistics such as average round-trip time and packet loss at the end of the session.

**Additional Features (Optional):**

* **Random Delays (Server):** Introduce random delays on the server side to simulate network latency.
* **Graphical Interface:** Create a graphical user interface to visualize the ping statistics.
* **Multithreading:** Implement multithreading to handle multiple ping sessions simultaneously.

**Note:**

Before starting the implementation, ensure that you understand the ICMP protocol and the basics of network latency measurement using ping. Familiarize yourself with libraries like scapy for working with ICMP packets in Python. Additionally, consider error handling and logging for a more robust implementation.