```java
import java.io.*;
import java.net.Socket;
import java.nio.charset.StandardCharsets;
import java.util.zip.GZIPOutputStream;

public class client {
    public static void main(String[] args) {
        try {
            System.out.println("Connecting to the server...");

            // Create a socket and connect to the server on localhost, port 8080
            Socket socket = new Socket("localhost", 8080);

            System.out.println("Connected to the server.");

            // Create output stream for communication with the server
            ObjectOutputStream objectOutputStream = new ObjectOutputStream(
                    new GZIPOutputStream(socket.getOutputStream()));

            // Specify the folder path on the client side
            String desktopPath = System.getProperty("user.home") + "/Desktop" + "/Files";
            File folder = new File(desktopPath);

            // List files in the folder
            File[] files = folder.listFiles();
            if (files == null) {
                System.err.println("No files found in the folder: " + desktopPath);
                return;
            }

            // Send the number of files to expect
            objectOutputStream.writeInt(files.length);
            objectOutputStream.flush();

            // Send up to 20 files at a time
            int filesToSend = Math.min(20, files.length);
            for (int i = 0; i < filesToSend; i++) {
                File file = files[i];

                // Send the compressed file content to the server
                byte[] compressedContent = compress(readFileContent(file));
                objectOutputStream.writeObject(compressedContent);
                objectOutputStream.flush();
            }

            // Close the streams and socket
            objectOutputStream.close();
            socket.close();

        } catch (IOException e) {
            System.err.println("Connection failed. Make sure the server is running and check your firewall settings.");
            e.printStackTrace();
        }
    }

    private static byte[] compress(String content) throws IOException {
        ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
        try (GZIPOutputStream gzipOutputStream = new GZIPOutputStream(byteArrayOutputStream)) {
            gzipOutputStream.write(content.getBytes(StandardCharsets.UTF_8));
        }
        return byteArrayOutputStream.toByteArray();
    }

    private static String readFileContent(File file) throws IOException {
        StringBuilder fileContent = new StringBuilder();
        try (BufferedReader fileReader = new BufferedReader(new FileReader(file))) {
            String line;
            while ((line = fileReader.readLine()) != null) {
                fileContent.append(line).append("\n");
            }
        }
        return fileContent.toString();
    }
}

```