```java
/**
 * SendEmail.java
 * Created on March 31, 2013 by Matthew A. Crist.
 *
 * This is an updated module from old shell scripts to the new Java method.

##############################################################################
# Created By: Wesley R. Howell
# Modified by: Matthew A. Crist (2013-03-08)
# Shell Script to Process an email message
#
# invocation: ./route-email.sh "inputFile"
#       inputFile - is the email message in the incoming/email folder on the
#                   server.
#
# Server listening port: 20005
#
# CHANGE LOG:
#----------------------------------------------------------------------------
# 2013-03-31    -   Modified file to use new Java invocation.
# 2013-03-23    -   Changed path to module from send-email to send.
# 2013-03-08    -   Added network listening capabilities to script and changed
#                   loop to infinite loop until kill command receive as well as
#                   removed the need for multiple files.
##############################################################################

 */

package modules.email.send;

import java.io.*;
import java.net.*;
import java.util.*;

class SendEmail {
    public static void main(String[] args) {
        boolean listening = true;

        try {
            // Listen for connection via server port 20005
            ServerSocket server = new ServerSocket(20005);

            while(listening) {
                // Wait until the client connects
                Socket client = server.accept();

                // Handles for input and output streams relating to the socket connection
                PrintWriter out = new PrintWriter(client.getOutputStream(), true);
                BufferedReader in = new BufferedReader(new InputStreamReader(client.getInputStream()));

                // Buffers
                String input, store="", request="";

                // Read the input from the connection
                while((input = in.readLine()) != null) {
                    request += input;
                }   // end while

                // Write the input out to a temporary file for shifting.
                String fileName = (new Date()).toString();
                // Since I do not like whitespace and : will make the console go apeshit
                fileName = fileName.replace(':', '_');
                fileName = fileName.replace(' ', '_');

                // The ACL2 string that will be executed.
                String acl2 = "(in-package \"ACL2\")" +
                            "(include-book \"modules/email/send/route-email\" :uncertified-okp t)" +
                            "(rwEmail \"" + request + "\" \"" + fileName + "\" state)";
```

```java
            // Initialize ACL2 and dump its output to the log
            System.out.println("Executing ACL2 runtime for SendEmail...");
            ProcessBuilder processBuilder = new ProcessBuilder("acl2");
            File log = new File("logs/email/send/acl2_log.txt");
            processBuilder.redirectErrorStream(true);
            processBuilder.redirectOutput(log);

            Process process = processBuilder.start();
            PrintWriter procIn = new PrintWriter(process.getOutputStream());

            // Write the ACL2 to the process, close ACL2
            procIn.println(acl2);
            procIn.println("(good-bye)");
            procIn.flush();
            procIn.close();
        }    // end while loop
    } catch(Exception e) {
        e.printStackTrace();
    }    // end try-catch
    }    // end function main
}   // end class SendEmail
```