**POTENTIAL ISSUE: Potentially Unsafe Code - Public Class Not Declared as Final**

Line: 35 - C:\Users\Jeremy\Downloads\RecoverController(1).java

The class is not declared as final as per OWASP recommendations. It is considered best practice to make classes final where possible and practical (i.e. It has no classes which inherit from it). Non-Final classes can allow an attacker to extend a class in a malicious manner. Manually inspect the code to determine whether or not it is practical to make this class final.

public class RecoverController {

**STANDARD: Potentially Unsafe Code - getParameter**

Line: 53 - C:\Users\Jeremy\Downloads\RecoverController(1).java

This function extracts data directly from the HTTP request. Manually check the code to ensure the data obtained is correctly validated and/or sanitised.

String usernameOrEmail = StringUtils.trimToNull(request.getParameter("usernameOrEmail"));

**HIGH: Potentially Unsafe Code - Poor Input Validation**

Line: 53 - C:\Users\Jeremy\Downloads\RecoverController(1).java

The application appears to use data contained in the HttpServletRequest without validation or sanitisation. No validator plug-ins were located in the application's XML files.

String usernameOrEmail = StringUtils.trimToNull(request.getParameter("usernameOrEmail"));

**STANDARD: Potentially Unsafe Code - getParameter**

Line: 62 - C:\Users\Jeremy\Downloads\RecoverController(1).java

This function extracts data directly from the HTTP request. Manually check the code to ensure the data obtained is correctly validated and/or sanitised.

String recaptchaResponse = request.getParameter("g-recaptcha-response");

**HIGH: Potentially Unsafe Code - Poor Input Validation**

Line: 62 - C:\Users\Jeremy\Downloads\RecoverController(1).java

The application appears to use data contained in the HttpServletRequest without validation or sanitisation. No validator plug-ins were located in the application's XML files.

String recaptchaResponse = request.getParameter("g-recaptcha-response");

**LOW: Potentially Unsafe Code - Operation on Primitive Data Type**

Line: 77 - C:\Users\Jeremy\Downloads\RecoverController(1).java

The code appears to be carrying out a mathematical operation on a primitive data type. In some circumstances this can result in an overflow and unexpected behaviour. Check the code manually to determine the risk.

for(int i=0; i<PASSWORD\_LENGTH; i++) {