HCL Deliverables and Documentation

Group A: Alexander Molina, Marshall Pennington, Matthew Rebeles, Matthew Yonsetto

Vulnerabilities:

Vulnerability 1 - Insecure direct object references (IDOR): A type of access control vulnerability that occurs when an application exposes internal object identifiers (such as a file or URL path) to users without proper access controls. This vulnerability was introduced by and developed by **all four members of the group.**

Vulnerability 2 - Unvalidated Redirects and Forwards: Unvalidated redirects and forwards vulnerability is a web application security issue where a website or web application redirects users to another page or forwards them to a different URL without properly validating or sanitizing the target destination. This type of attack is typically used by attackers to redirect users to phishing pages in order to steal user credentials. This vulnerability was introduced by and developed **by all four members of the group.**

Minor Vulnerabilities:

We also introduced some smaller vulnerabilities that we didn't want to classify as a full vulnerability as they are better described as violations of best practices.

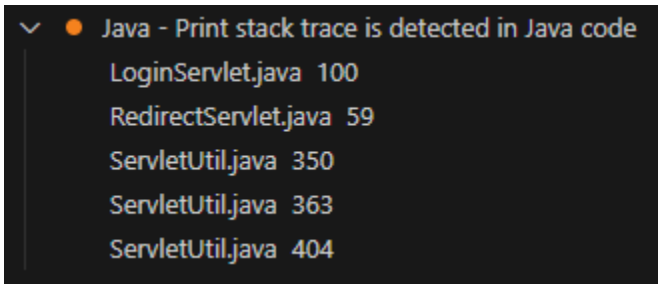
- Password stored in Java String object

  -This is where the password input is kept in memory until it is cleared by the garbage

collected.

- printStackTrace is used in the program

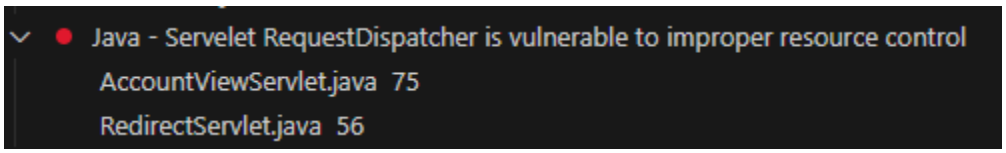
  -printStackTrace prints a trace of all of the method calls leading up to a crash. This can be used to find out information on the program and see if there are any known vulnerabilities.



AppScan Screenshot (If Applicable):

Vulnerability 1: Unfortunately, after multiple attempts, AppScan did not recognize the vulnerability, despite existing within the code. **This vulnerability is very similar to the second vulnerability and is contained under the term as shown here*

Vulnerability 2:



Where in Code:

Vulnerability 1: File path for the vulnerability location is:

AltoroJ/src/com/ibm/security/appscan/altoromutal/servlet/RedirectServlet.java

AltoroJ-Altoro-3.4\WebContent\WEB-INF\web.xml

Vulnerability 2: File path for the vulnerability location is:

AltoroJ/src/com/ibm/security/appscan/altoromutal/servlet/RedirectServlet.java

AltoroJ-Altoro-3.4\WebContent\WEB-INF\web.xml

Possible Solutions:

Vulnerability 1: There are a few ways to prevent the IDOR vulnerability such as the implementation of strict access controls such as the use of role-based access control (RBAC) or attribute-based access control (RBAC). Additionally, instead of directly referencing sensitive objects or data, the use of unique identifiers or tokens can prevent attackers from manipulating direct object references.

Vulnerability 2: In order to mitigate this vulnerability, web developers should implement proper input validation and sanitization for redirect parameters, validate and verify redirect URLs against a whitelist of trusted domains or paths, and avoid user-controlled input for redirects whenever possible. Additionally, the use of server-side validation and avoiding relying solely on client-side verification will help in preventing this vulnerability.