CSOC 1030: Assignment #1

July 05, 2023

**Prepared By**: *Ashish Kishor Hedau*

Table of Contents

[Insecure Direct Object Reference in NHS terminal 2](#_Toc137283759)

[Description 2](#_Toc137283760)

[Impact 2](#_Toc137283761)

[Recommendations 2](#_Toc137283762)

[Steps to Reproduce 3](#_Toc137283763)

# Insecure Direct Object Reference in NHS terminal

## Description

The assessed “http://10.5.30.10” application was found to implement a login-type functionality that required users to submit a password to gain access to the terminal and see the secret message. This login-type type functionality was found to be implemented as a client-side control, and the protected content was found to be delivered to the users as part of the application’s front end HTML file through insecure direct object reference.

## Impact

An attacker could modify the client-side behavior of the web page to bypass the authentication, and/or view the protected content by inspecting the page’s source code. The attacker could see the “secret message” due to the reference mentioned in the client-side HTML.

## Recommendations

Implement the secret message and delivery of the protected content as server-side functions combined with indirect object reference. This would ensure the users don’t have access to the reference that directs to the “secret message”. It would also ensure the content is not tampered with or modified because of the exposed reference.

For more information on IDOR related Common Weaknesses, refer to CWE-639 and CWE-862.

Reference: <https://cheatsheetseries.owasp.org/cheatsheets/Insecure_Direct_Object_Reference_Prevention_Cheat_Sheet.html>

## Steps to Reproduce

The target application was found to be protected by a code that required users to submit the correct password before accessing the protected content on the site, as pictured below:

![A computer screen with white text

Description automatically generated]()

Inspecting the page source code using the browser’s built-in developer tools revealed that the protected code “secret message” could be accessed through the fetch details mentioned in the HTML script as “/api/v3/message”.

![A computer screen with white text

Description automatically generated]()

By inserting the Direct Object Reference in front of the URL in scope produces the “secret message” as shown in the image below.

![A screenshot of a computer

Description automatically generated]()