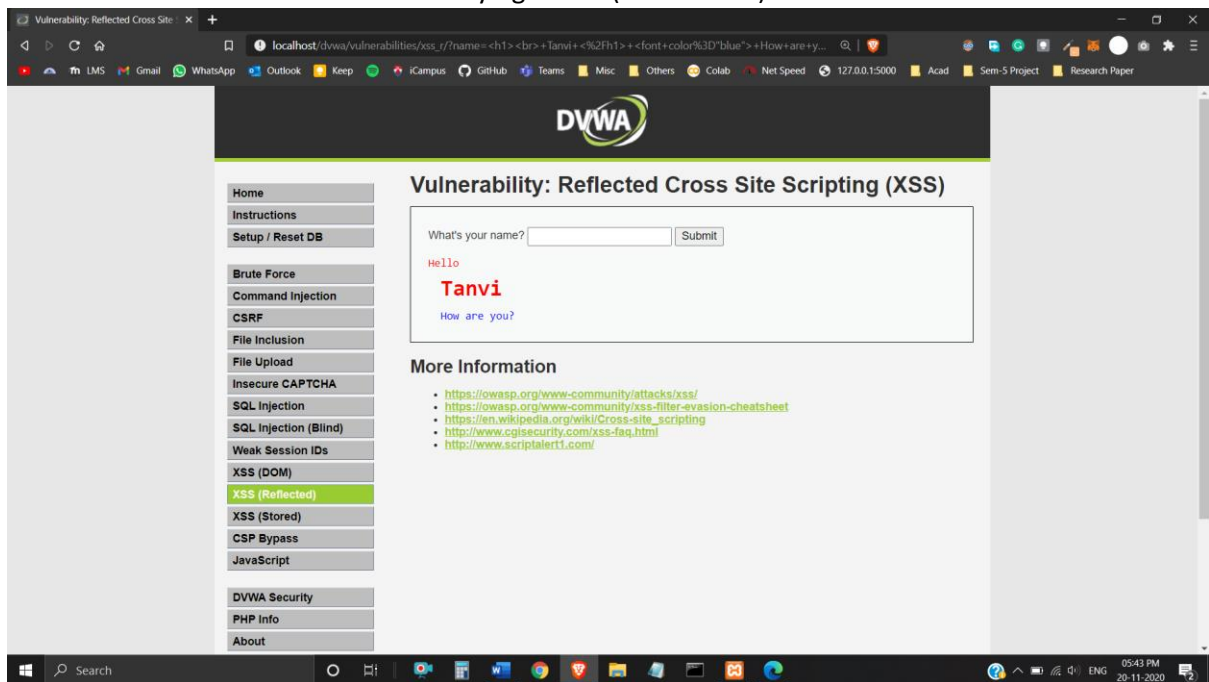


Lab-12 Cryptography and Network Security

Name: Tanvi Penumudy (E18CSE187)

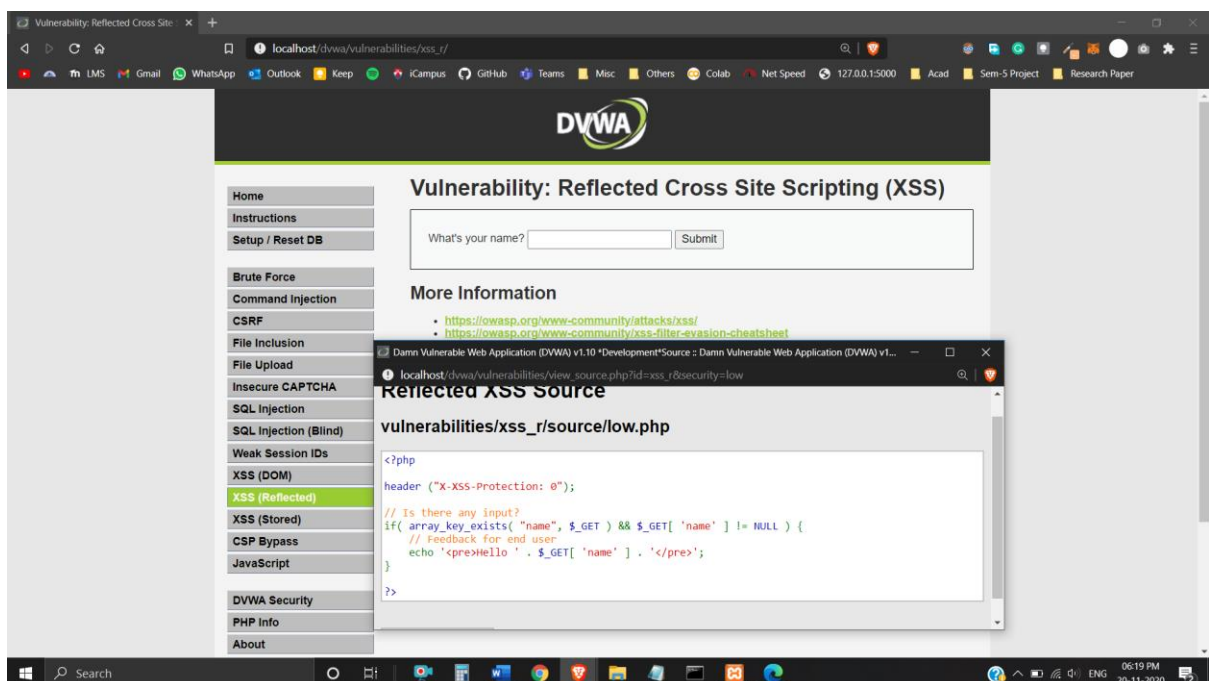
Objective: To Perform Reflected Cross-Site Scripting (XSS) Attack on the damn vulnerable web application (DVWA). Set the security level “Low, Medium, High” to inject malicious scripts into the web application.

Trying Basics (HTML Code)

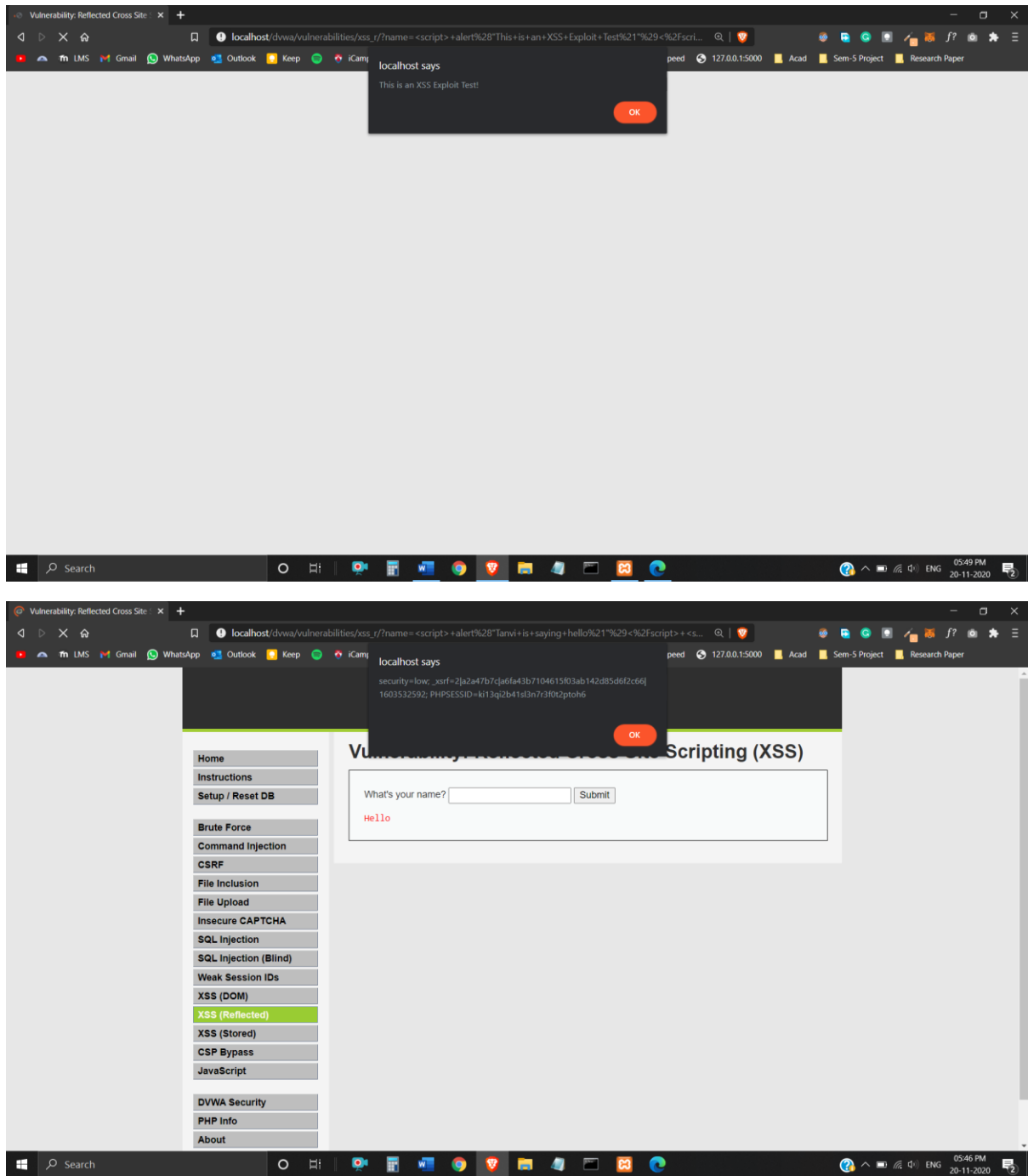


Security Level: Low

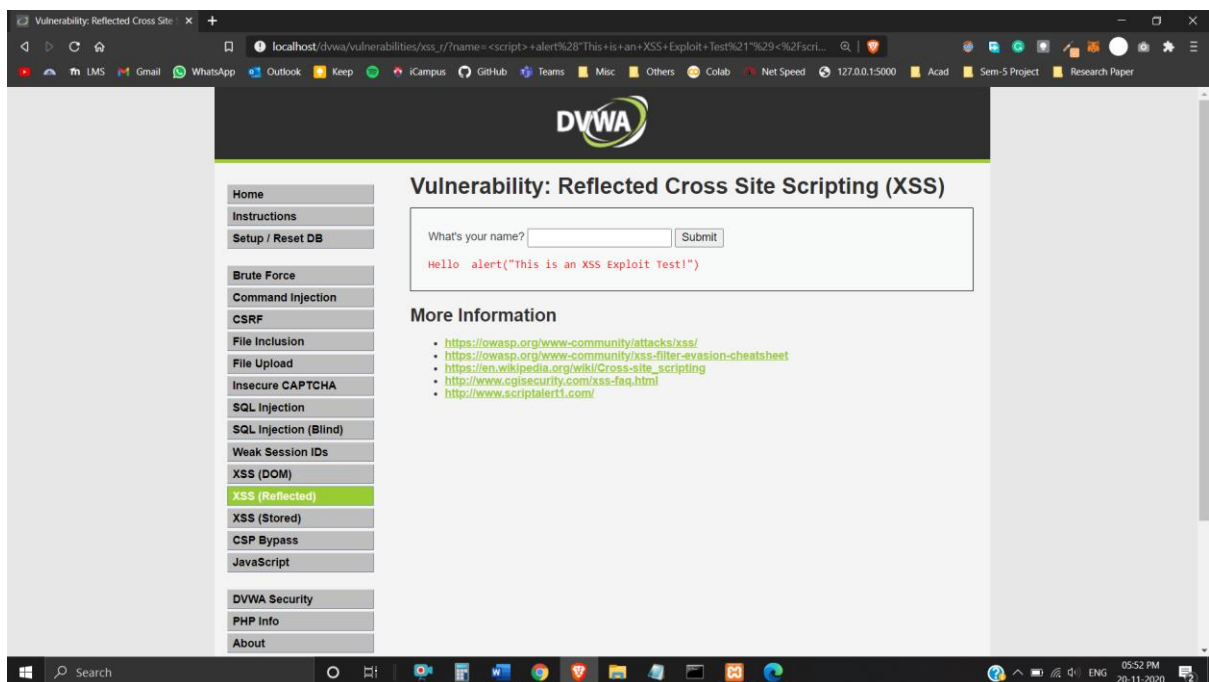
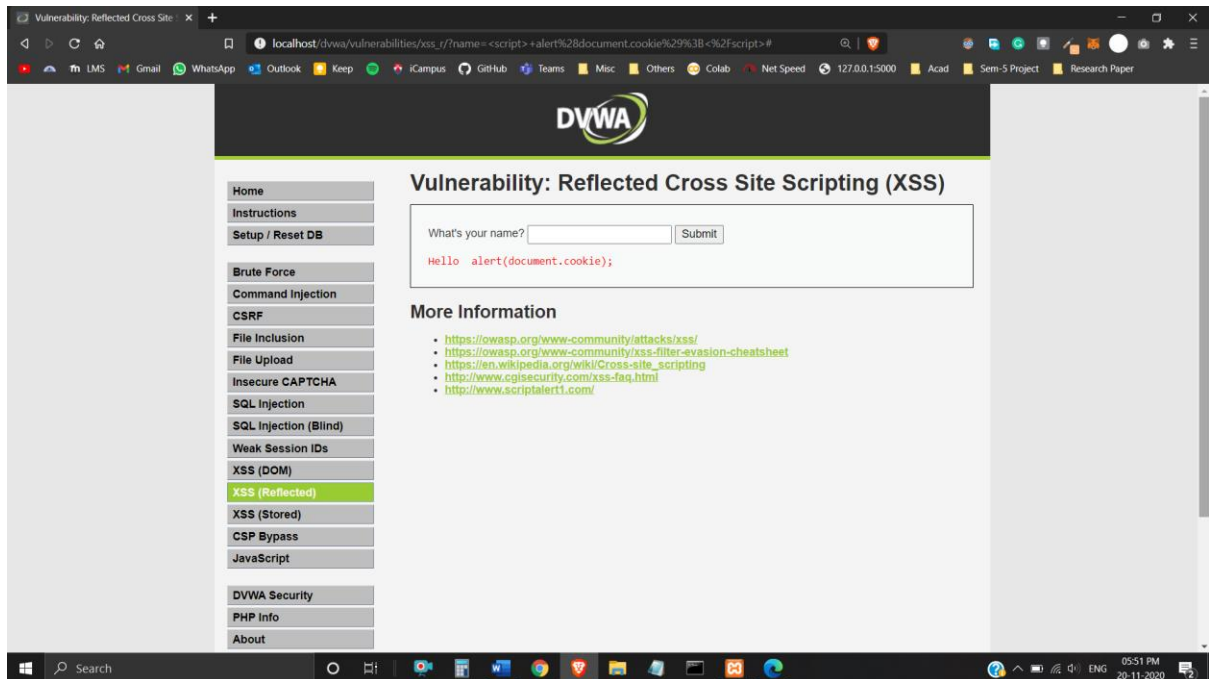
Behind the Scenes: Retrieving the text from the Input Box and echoing it in a pre-tag



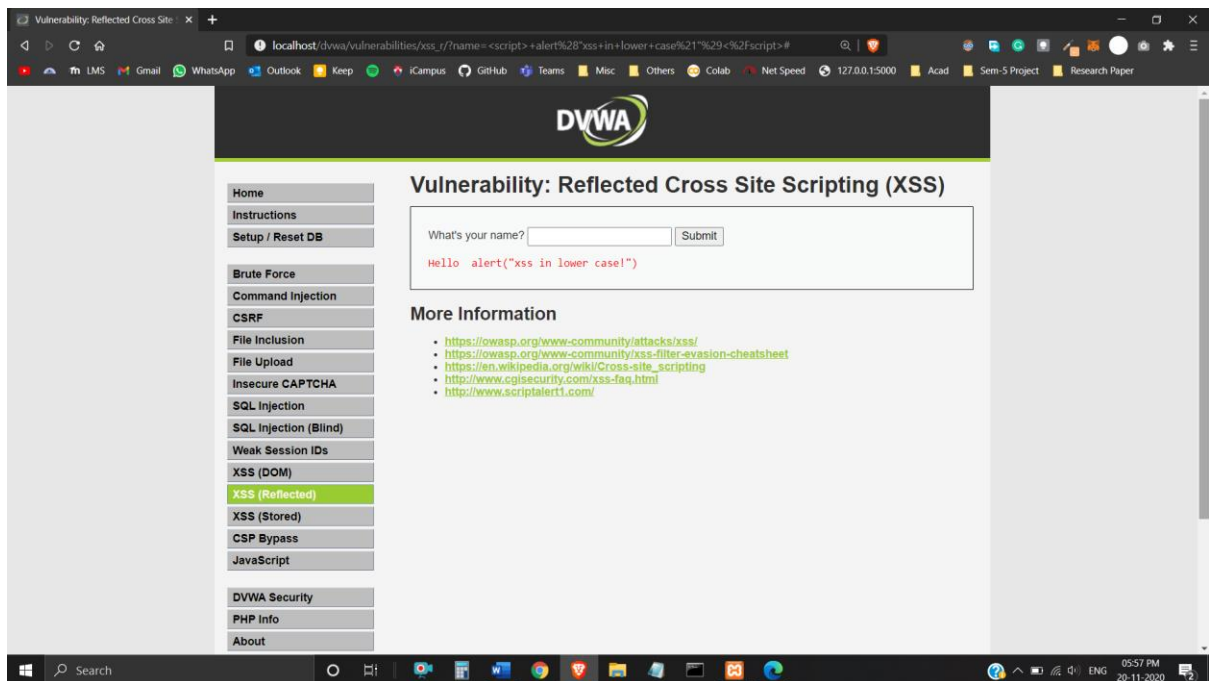
Attack Successful!



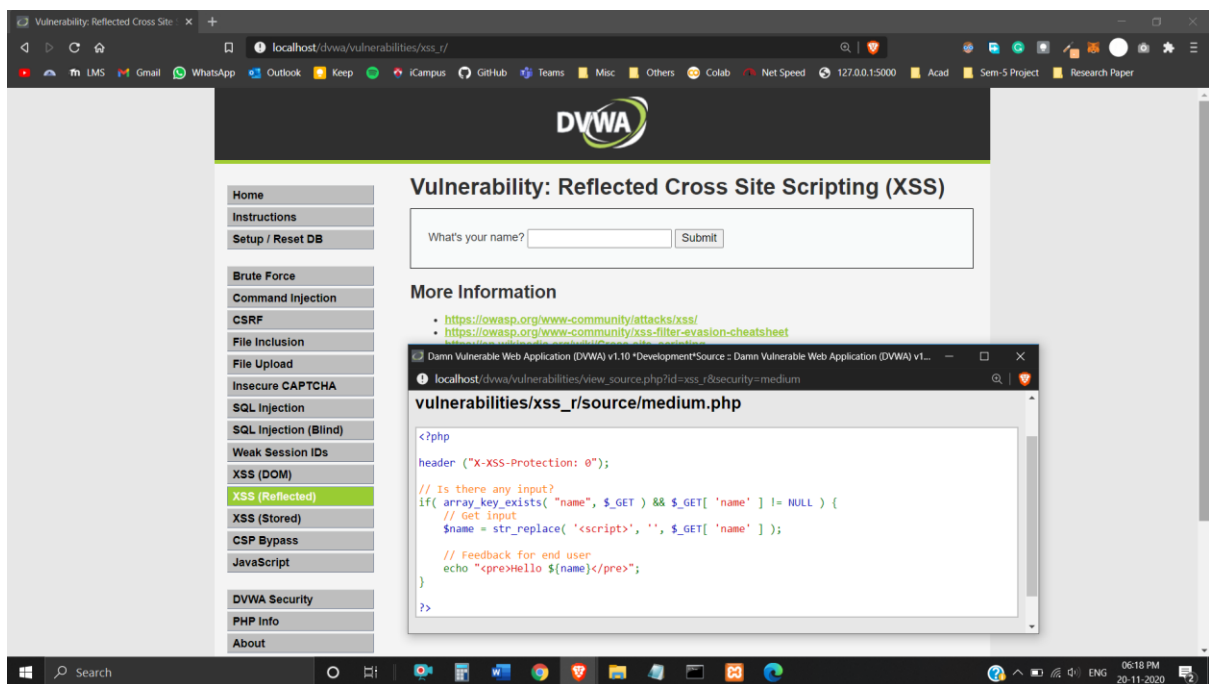
Trying earlier mechanisms (Not Working!)



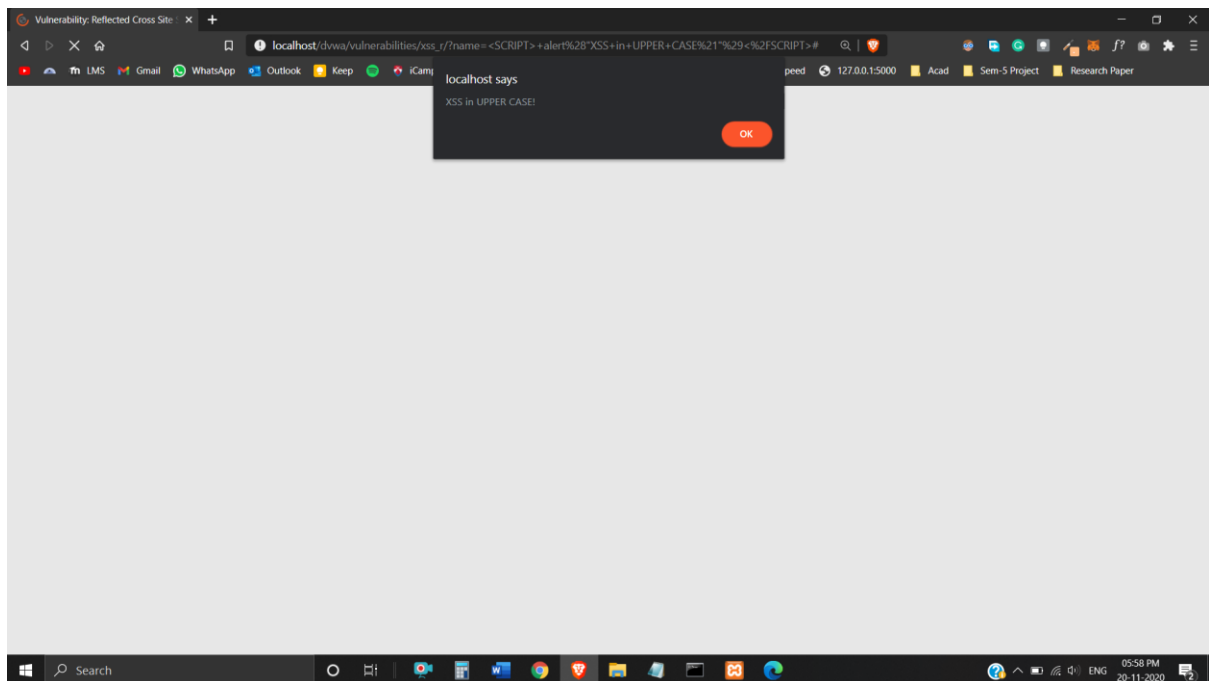
Using 'script' in lower case (Not Working!)



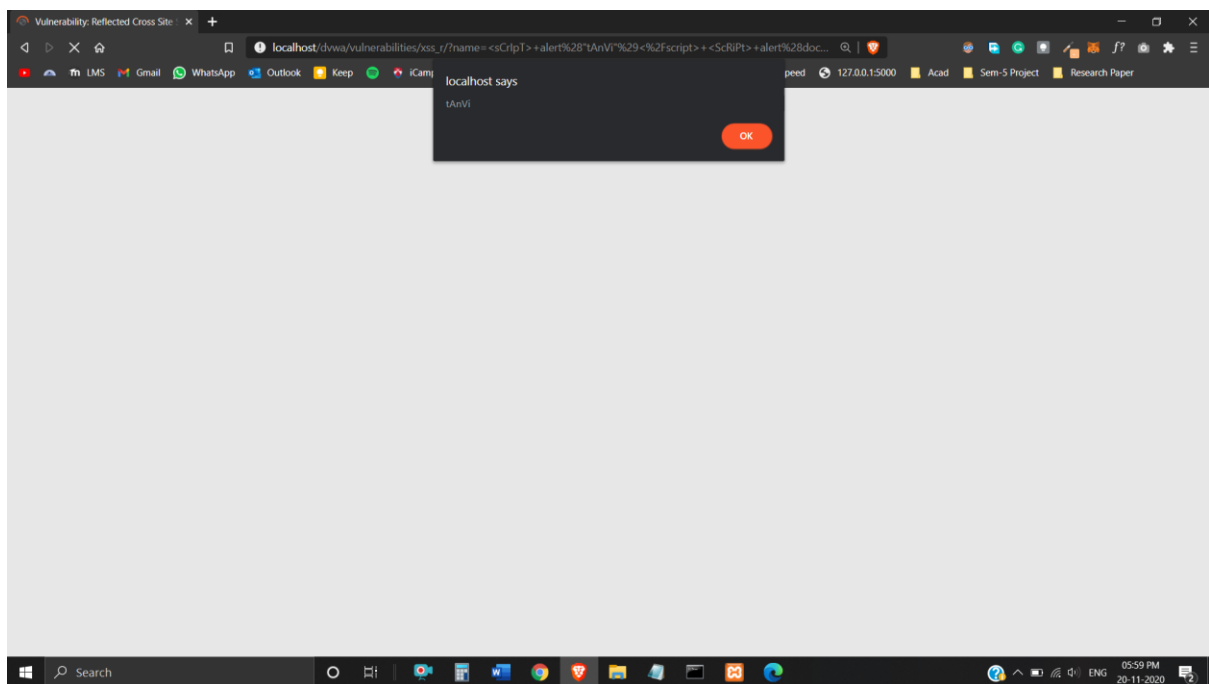
Behind the Scenes: Replacing <script> tag with blank space

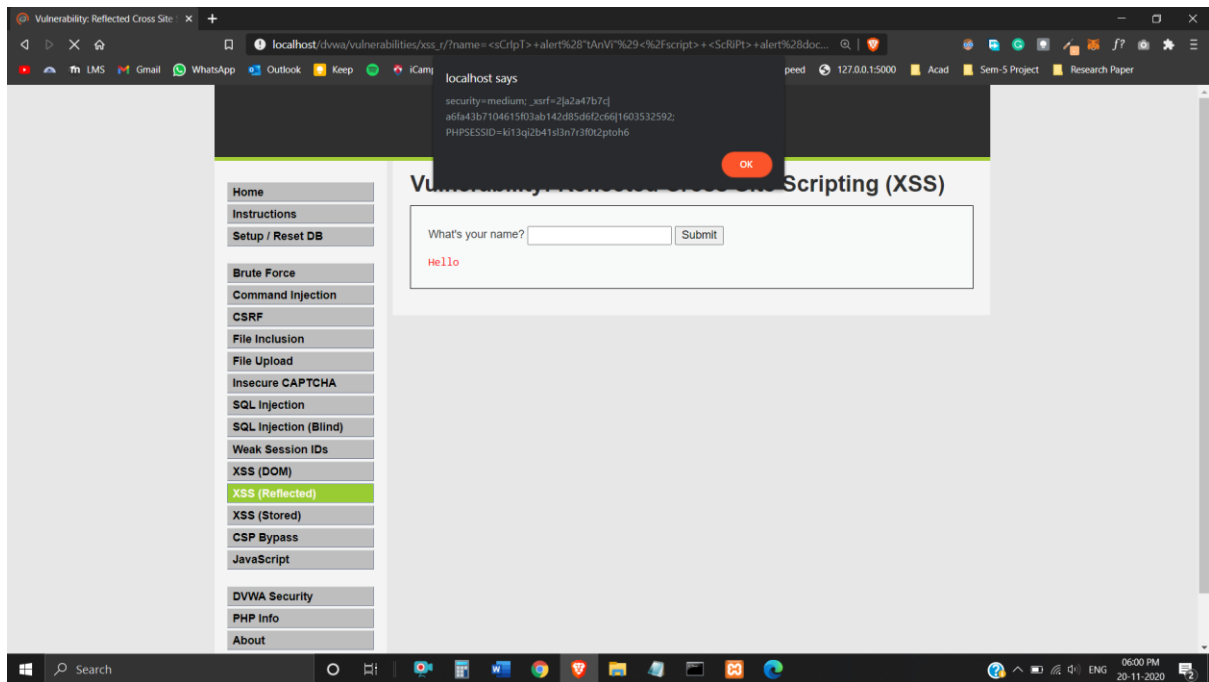


Using 'SCRIPT' in UPPER CASE (Attack Successful!)



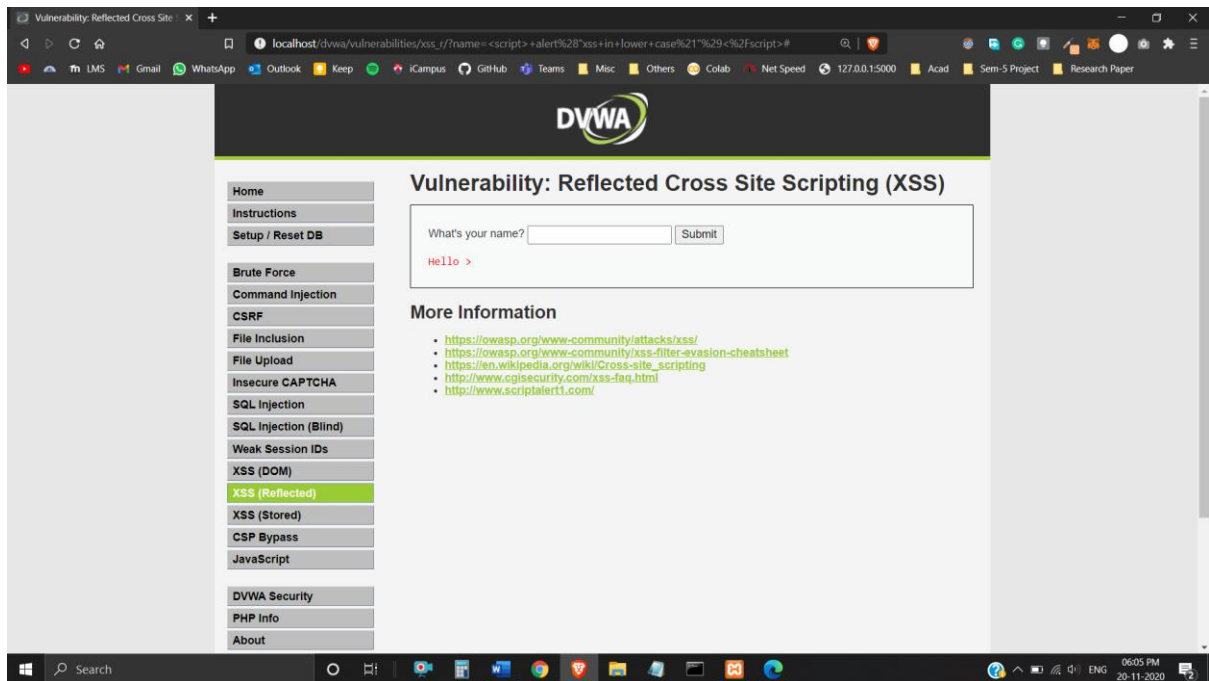
Using 'ScRiPt'/'sCrIpT' in MiXeD cAsE (Attack Successful!)



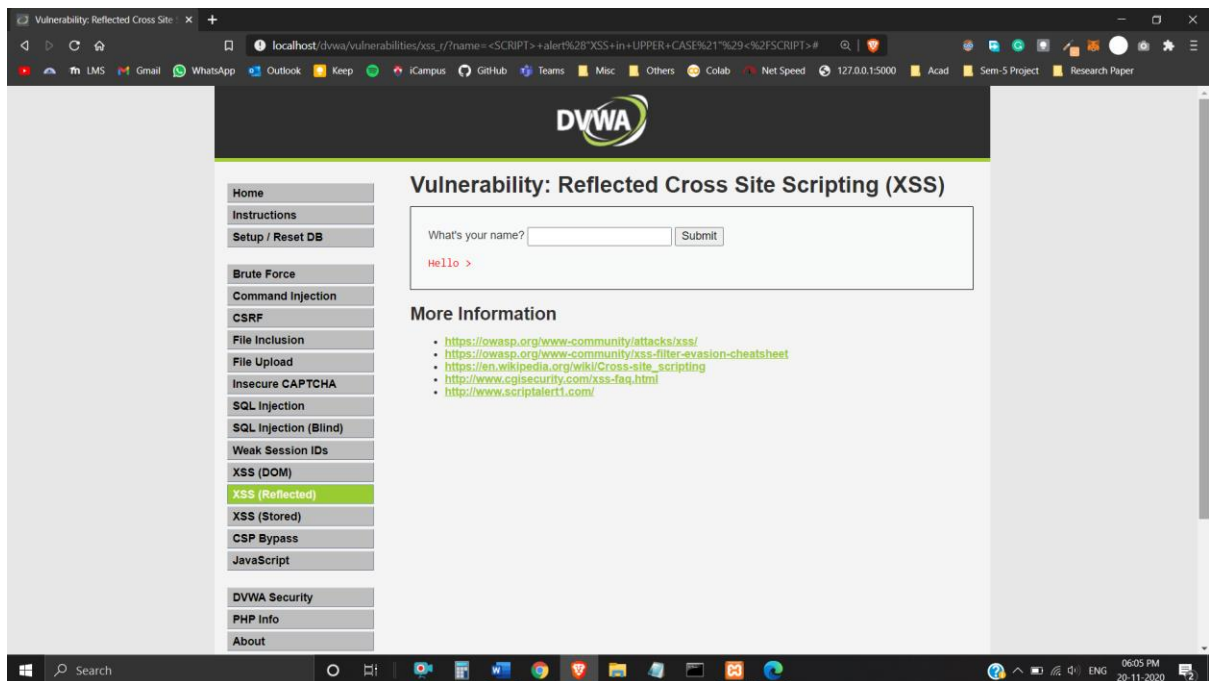


Security Level: High

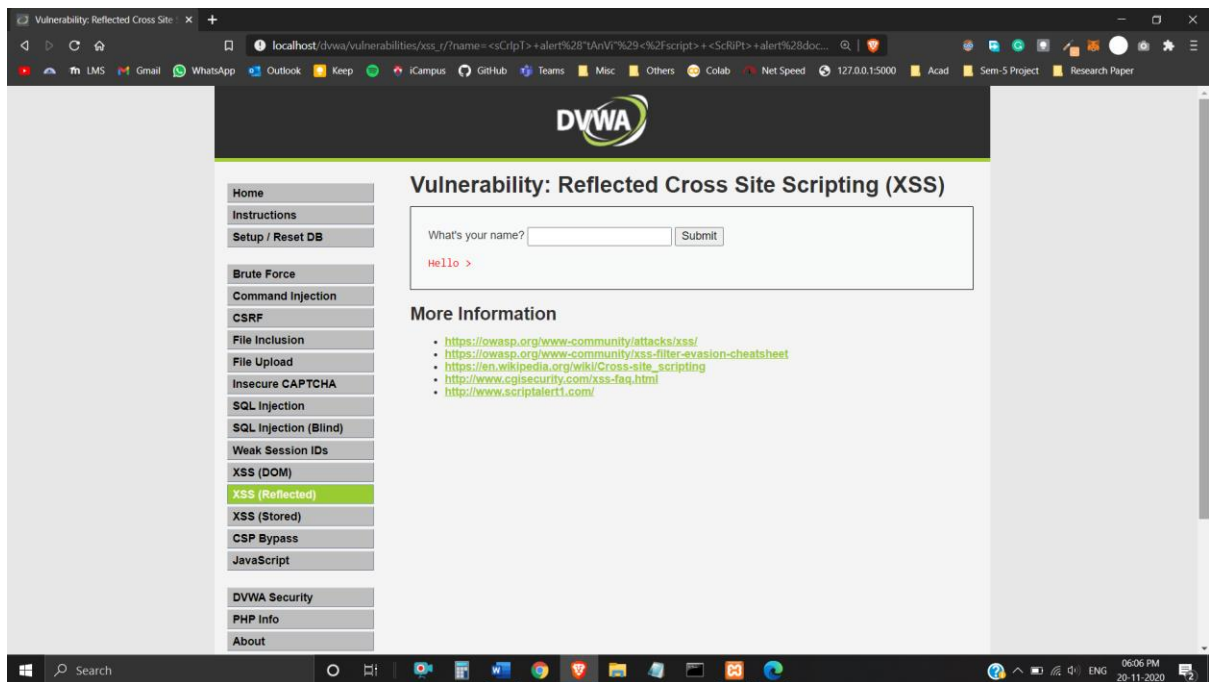
Trying earlier mechanisms – Using 'script' in lower case (Not Working!)



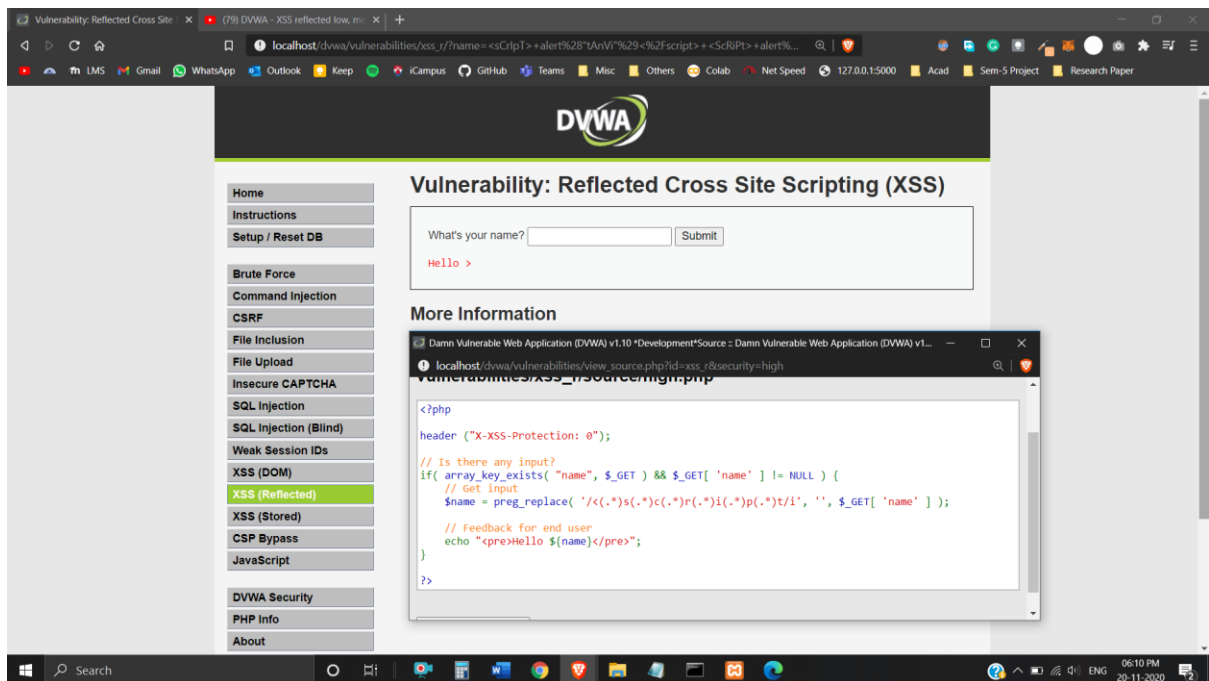
Using 'SCRIPT' in UPPER CASE (Not Working!)



Using 'ScRiPt'/'sCrIpT' in MiXeD cAsE (Not Working!)



Behind the Scenes: Using Regular Expression (regex) to detect the <script> tag



The screenshot shows the DVWA (Damn Vulnerable Web Application) interface. The main heading is "Vulnerability: Reflected Cross Site Scripting (XSS)". Below it is a form with the label "What's your name?" and a "Submit" button. The output area shows "Hello >". A sidebar on the left lists various vulnerabilities, with "XSS (Reflected)" selected. An overlay window titled "More Information" displays the source code of the page, which includes a PHP script that checks for the presence of a "name" parameter and echoes it back, demonstrating the vulnerability.

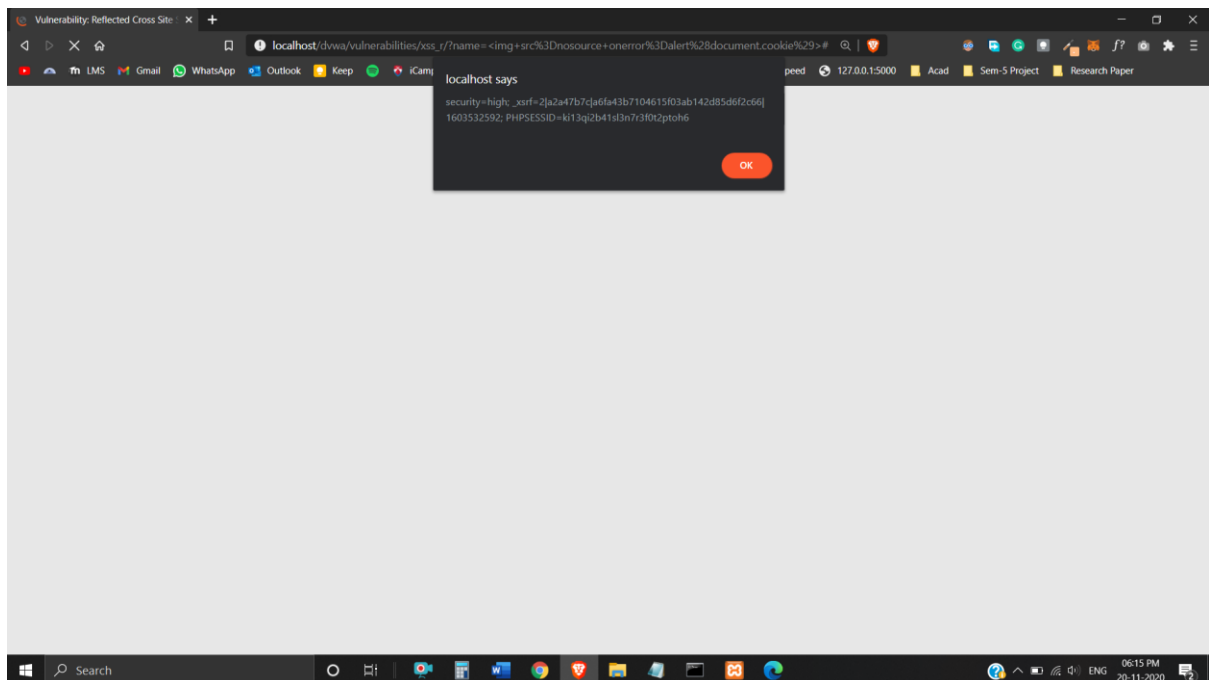
```
<?php
header ("X-XSS-Protection: 0");

// Is there any input?
if( array_key_exists( "name", $_GET ) && $_GET[ 'name' ] != NULL ) {
    // Get input
    $name = preg_replace( '/<(.*)(.*)r(.*)i(.*)p(.*)t/i', '', $_GET[ 'name' ] );

    // Feedback for end user
    echo "<pre>Hello ${name}</pre>";
}

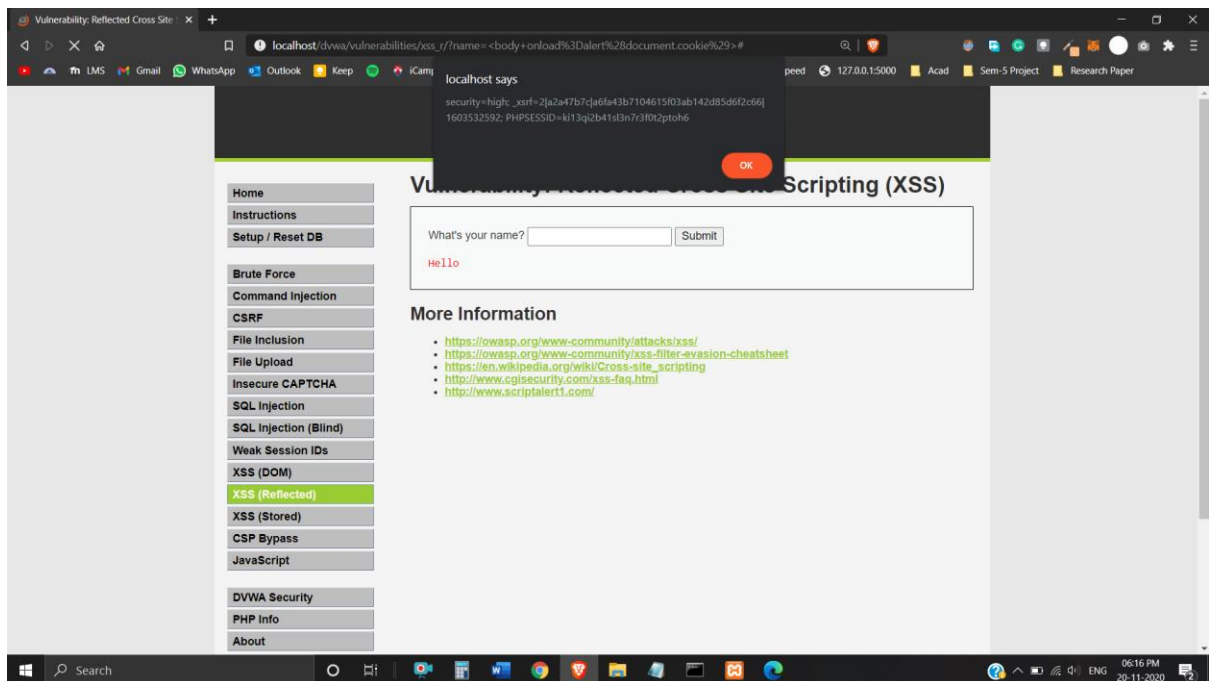
?>
```

Using tag (Attack Successful!)



The screenshot shows the DVWA interface after a successful attack. The URL in the browser's address bar is `localhost/dvwa/vulnerabilities/xss_r/?name=<img+src%3Dnosource+onerror%3Dalert%28document.cookie%29>#`. The output area displays a message from the browser: "localhost says security=high; _xsrf=2js247b7d46f43b710461503ab142d85d6f2c66j 1603532592; PHPSESSID=ki13q2b41s3n7r3f02ptoh6". An "OK" button is visible in the bottom right corner of the output area.

Using <body> tag (Attack Successful!)



Hence, successfully performed Reflected Cross-Site Scripting (XSS) Attack on DVWA in 'Low, Medium and High' Security Levels.