LAB06B: Bypassing preg\_replace(pattern, null, )

[Simon X. Camilo. Cybersecurity Student]

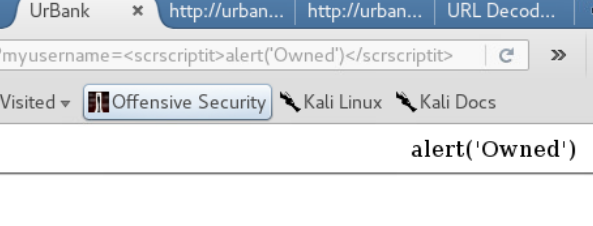
[BHCC CIT-275-WB Professor Philip Kazanjian]

Introduction

preg\_replace(pattern, null, ) is a function that is preventing us from running script on the address bar. This document tries to finds ways to bypass that function and run scripts on the address bar

**What is preg\_replace?**

preg\_replace is a function that replaces a string when it follows a pattern. In this case it’s being used to replace any S that is located in any element that contains an i. This makes it so attackers can’t use the most common methods of attack, and it requires a better understanding of code than str\_replace and str\_ireplace, it was more tedious.



**Image 1**

*Demonstration of preg\_replace(pattern, null, )*

A demonstration of how this function works can be seen in image 1. In this case, the function is preg\_replace(pattern, null, $\_REQUEST[‘myusername’]). This make it so when the url above is entered, instead of showing an input field, the only thing that’s shown is the name of the variable.

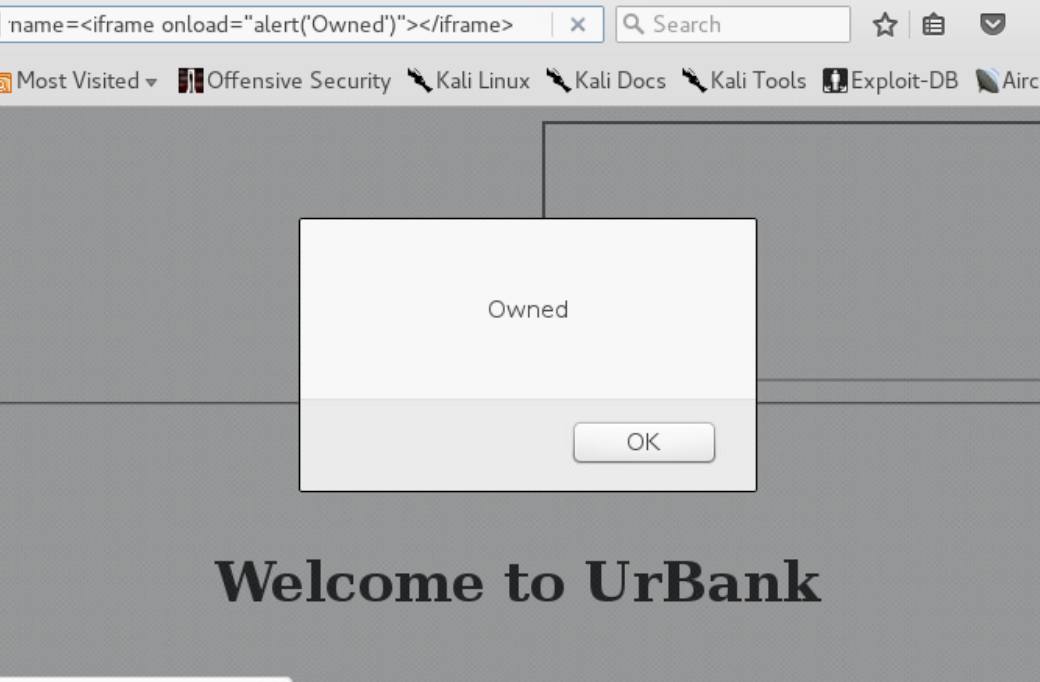
The page source as shown in image 2 also shows that it’s not there anymore, it has been replaced with nothing. This is how preg\_replace works.

  
**Image 2**  
*Demonstration of preg\_replace(pattern, null, ) in html source*

# Analysis

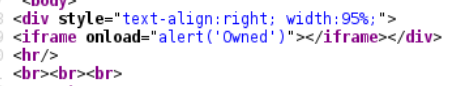
There are many ways to bypass this filter, the only thing needed is to execute something that doesn’t use “<script>” and doesn’t have an s. There are many things like iframe, img src, body, and many others that can achieve this.

Image 3 shows that using code such as <iframe onload="alert('Owned')"></iframe> it works, simply using code that doesn’t have an S on it is going to work. It has not been altered in any way either, as shown on image 4



**Image 3**

Preg\_replace was successfully bypass



**Image 4**

Code unaltered in html source

# Conclusion

A system administrator needs to knowledge that there are many things other than “<script>” that an attacker can use to prevent an attack from happening. Filtering the word s could also put limitations in development. It’s important to use a script that takes into consideration any piece of code that can be used in an attack

References

Preg\_replace. (n.d.). Retrieved February 26, 2021, from <https://www.php.net/manual/en/function.preg-replace.php>

PHP preg\_replace() Function. (n.d.). Retrieved February 26, 2021, from <https://www.w3schools.com/php/func_regex_preg_replace.asp>

Rizal, M. (2012, September 21). XSS 1. Retrieved March 07, 2021, from <https://hack.me/101020/xss-1.html>