Documenting Findings: XSS Payload Generation and WAF Evasion

Prepared by Febin Jacob

1. Introduction

- This document summarizes the findings from testing a Python script designed to generate Cross-Site Scripting (XSS) payloads capable of evading character-based filters commonly used by Web Application Firewalls (WAFs). The tests were conducted using the Damn Vulnerable Web Application (DVWA) configured with WAF settings to simulate real-world security measures.

2. Test Environment

- - Web Application: Damn Vulnerable Web Application (DVWA)
- - WAF Configuration:
- - Default rules to block common XSS attack patterns.
- - Input validation enabled for form fields and URL parameters.

3. Testing Methodology

- The following approach was taken to test the effectiveness of the generated XSS payloads:
- 1. Payload Generation: A Python script was developed to generate various XSS payloads using different encoding and evasion techniques.
- 2. Submission of Payloads: The generated payloads were submitted through:
- - Form fields in the DVWA application.
- - URL parameters for GET requests.
- 3. Monitoring: WAF logs and DVWA responses were monitored to determine whether the payloads were blocked or executed.

4. Payloads Tested

- The following payloads were generated by the script and tested against the WAF:
- 1. Base Payload: <script>alert('XSS');</script>
- 2. Evasion Payload: alert(String.fromCharCode(60, 115, 99, 114, 105, 112, 116, 62, 97, 108, 101, 114, 116, 40, 39, 88, 83, 83, 39, 41, 59, 60, 47, 115, 99, 114, 105, 112, 116, 62));
- 3. Obfuscated Payload: '';!--"<XSS>=&{()};
- 4. JavaScript URI: javascript:alert(1);
- 5. URL Encoded Payload: %3Cscript%3Ealert%281%29%3C%2Fscript%3E

5. Results

- The outcomes of the tests conducted are summarized below:
- - The Base Payload and Obfuscated Payload were successfully blocked by the WAF due to their recognizable patterns and tags.
- - The Evasion Payload and URL Encoded Payload managed to bypass the WAF filters, indicating that encoding techniques can be effective in evading character-based filters.
- - The results highlight the importance of employing comprehensive security measures, as WAFs relying solely on character-based filtering may be vulnerable to encoded attacks.

Summary of Findings

- The testing demonstrated that while many XSS payloads are effectively blocked by WAFs, specific encoding and obfuscation techniques can enable certain payloads to evade detection. Further research could explore additional evasion methods and evaluate their effectiveness against different WAF configurations.

Recommendations

- - Continuous Testing: Regularly test WAFs with updated payloads to identify weaknesses.
- - WAF Configuration: Consider adopting a combination of character-based and context-aware filtering methods for enhanced protection.
- - Awareness and Training: Educate developers about the evolving techniques used in XSS attacks and the importance of secure coding practices.