**Batch: B1 Roll No.: 16010121045**

**Experiment No. 9**

|  |
| --- |
| **Title:**  Buffer Overflow Vulnerability |

**Objective:**

Buffer Overflow Vulnerability

|  |  |
| --- | --- |
| **CO** | **Outcome** |
| **CO3** | Comprehend post exploitation phase of penetration testing. |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

[**https://www.imperva.com/learn/application-security/buffer-overflow/**](https://www.imperva.com/learn/application-security/buffer-overflow/)

**Theory**:-

**WordPress Security Basics:** WordPress is a popular content management system (CMS) powering millions of websites worldwide. However, its popularity makes it a prime target for attackers. Vulnerabilities in WordPress plugins, themes, and the core itself can be exploited to gain unauthorized access to websites, deface them, or steal sensitive data.

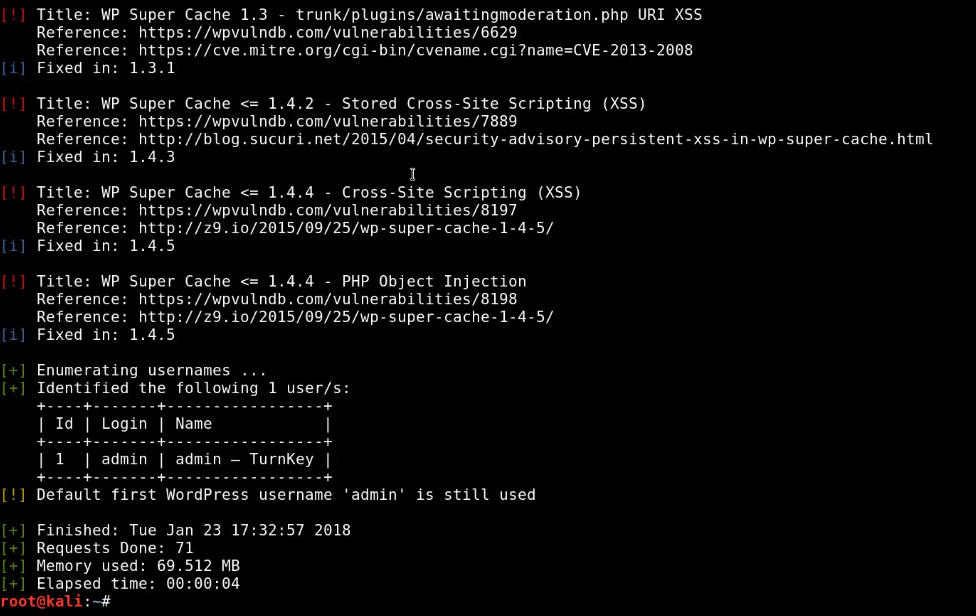
**WPScan:** WPScan is a free, open-source tool specifically designed for WordPress vulnerability scanning. It helps identify security weaknesses in WordPress installations by scanning for vulnerable plugins, themes, and core files. It also enumerates user accounts and performs various other security checks.

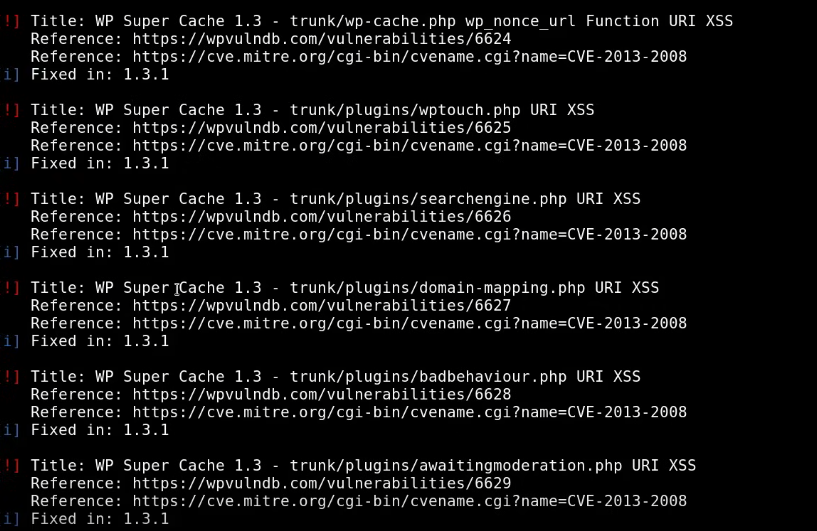
**Types of Vulnerabilities:** Common vulnerabilities in WordPress include SQL injection, cross-site scripting (XSS), cross-site request forgery (CSRF), remote code execution (RCE), and file inclusion vulnerabilities. These vulnerabilities can be exploited to execute malicious code, manipulate databases, or gain administrative access.

**Exploitation Process:** Exploiting a vulnerability typically involves identifying a target, scanning it for vulnerabilities using WPScan, selecting an appropriate exploit, and executing it. Once a vulnerability is successfully exploited, attackers can gain unauthorized access to the target system and perform malicious activities.

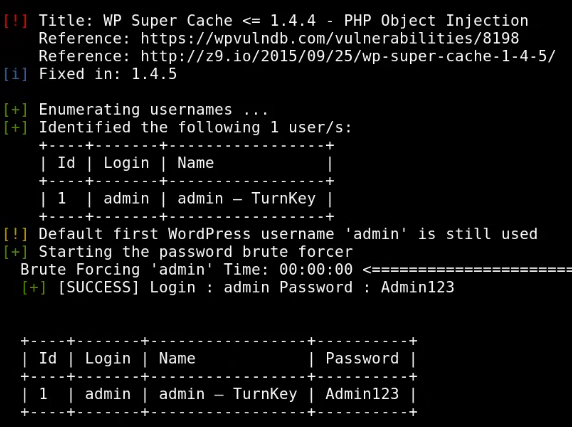
**Implementation:**

****

****

****

**Wordlist attack:**

****

**Conclusion:**

**Successfully implemented and exploited a wordpress server using wpscan.**