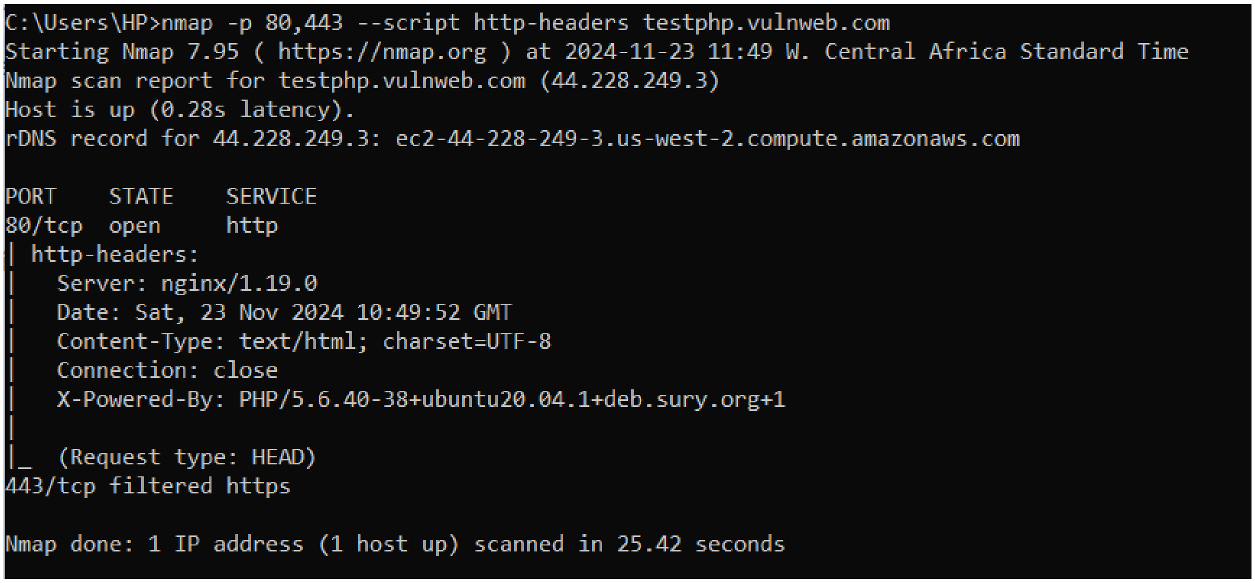
Introduction:

The following is a literature review and vulnerability analysis of the assigned site, http://testphp.vulnweb.com .



```
C:\Users\HP>nmap -p 80,443 --script http-headers testphp.vulnweb.com
Starting Nmap 7.95 ( https://nmap.org ) at 2024-11-23 11:49 W. Central Africa Standard Time
Nmap scan report for testphp.vulnweb.com (44.228.249.3)
Host is up (0.28s latency).
rDNS record for 44.228.249.3: ec2-44-228-249-3.us-west-2.compute.amazonaws.com

PORT    STATE    SERVICE
80/tcp  open     http
| http-headers:
|   Server: nginx/1.19.0
|   Date: Sat, 23 Nov 2024 10:49:52 GMT
|   Content-Type: text/html; charset=UTF-8
|   Connection: close
|   X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
|
|_  (Request type: HEAD)
443/tcp filtered https

Nmap done: 1 IP address (1 host up) scanned in 25.42 seconds
```

Figure 1: Nmap scan of the assigned website

An nmap scan of the assigned website (figure 1) reveals that it is using multiple outdated software which are PHP 5.6.40 and Nginx 1.19.

Based on the above, it is evident that the website has at least 3 layers of vulnerabilities.

Layer 1: PHP 5.6.x

1. Reached end of life more than 5 years ago (in December 31, 2018), and no longer receives any form of updates (PHP, 2024).
2. A basic search for vulnerabilities affecting PHP 5.6.x in the National Vulnerability Database shows 110 matching records (NVD, 2024).
3. Among the vulnerabilities affecting this version of PHP are:
    a. CVE-2014-3622 with a severity score of 9.8 (critical) allows remote code execution from an attacker (CVE, 2014). Other vulnerabilities that allow remote code execution includes CVE-2016-4473, CVE-2016-5773, etc
    b. CVE-2016-5093 with a severity score of 8.6 (High) allows remote attackers to cause a Denial of Service (DoS) (CVE, 2016).
    c. CVE-2016-10712 with a severity score of 7.5 (High) allows attackers to set arbitrary metadata during file uploads.

Layer 2: Nginx 1.19

1. Reached end of life more than 3 years ago (May 25, 2021) and no longer receives security updates (End of Life, 2024).
2. According to the National Vulnerability Database, this version of Nginx is susceptible to a lot of vulnerabilities, including:
   a. CVE-2021-23017 allows attackers to forge UDP packets causing memory corruption and leading to worker process crashes (CVE, 2021).
   b. CVE-2022-41741 allows attackers to carry out memory corruption attacks using specially crafted audio/video files.

Layer 3: The Website Codebase

From layer 1 and 2, we have established that the software is running in an insecure environment. Starting from wrong server configuration that easily gives the names and version of its web server and PHP to the actual vulnerabilities of the aforementioned software applications. A simple interaction with the website itself shows that the code powering it is inherently susceptible to the following vulnerabilities:

1. **SQL Injection**: Risks from unsanitized user input fields (Galluccio, et al, 2020).

2. **Cross-Site Scripting (XSS)**: Vulnerabilities in client-side scripts; old but still present in a large number of web applications (OWASP, 2021).

Conclusion

The vulnerability analysis of http://testphp.vulnweb.com reveals that it is using outdated software, specifically PHP 5.6.40 and Nginx 1.19, both of which no longer receive security updates. This creates multiple layers of vulnerabilities. Additionally, the website's codebase is prone to common vulnerabilities like SQL Injection and Cross-Site Scripting (XSS). To enhance security, it is imperative to update these software components and implement best security practices.

REFERENCES:

PHP (2024) *PHP: Unsupported Branches*. Available from: https://www.php.net/eol.php [Accessed: 23 November 2024].

End of Life (2024) *Nginx End of Life*. Available from: https://endoflife.date/nginx [Accessed: 23 November 2024].

NVD (2024) *NVD Search Results*. Available from: https://nvd.nist.gov/vuln/search/results?form_type=Basic&results_type=overview&query=php%205.6.x&search_type=all&isCpeNameSearch=false [Accessed: 23 November 2024].

CVE (2014) *Common Vulnerabilities and Exposures*. Available from: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3622 [Accessed: 23 November 2024].

CVE (2016) *Common Vulnerabilities and Exposures*. Available from: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-5093 [Accessed: 23 November 2024].

CVE (2021) *Common Vulnerabilities and Exposures*. Available from: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-23017 [Accessed: 23 November 2024].

Galluccio, E., Caselli, E. and Lombari, G. (2020) SQL Injection Strategies: Practical techniques to secure old vulnerabilities against modern attacks. Packt Publishing Ltd.

OWASP Foundation (2021) OWASP Top 10. Available from: https://owasp.org/www-project-top-ten/ [Accessed: 11 November 2024].

Reflection

Reflect on this activity by answering the following questions:

- Did you have any issues or challenges with the literature search/audit on software sites and the national vulnerabilities database?

  Answer: Yes, I initially found it difficult to search the National Vulnerability Database for vulnerabilities affecting a specific version of a software.

- How did you overcome them?

Answer: I overcome the challenge by playing around, then observing that there is an advanced option for the search. In the advanced option you can type a vendor and select from the list of its products.

- How will they affect your final report?

  Answer: Now I can easily filter to the specific software and its version, this will help me in compiling a better final report.