Penetration Testing Terhadap Website Sekolah Menengah Atas ABC dengan Metode NIST SP 800-115

Syania Aulia Maherza[1], Bayu Hananto [2], I Wayan Widi Pradnyana[3]
[1,2,3]Informatika / Fakultas Ilmu Komputer
[1,2,3]Universitas Pembangunan Nasional Veteran Jakarta
[1,2,3]Jl. R.S. Fatmawati, Pondok Labu, Jakarta Selatan, DKI Jakarta, 12450, Indonesia
syaniaam@upnvj.ac.id[1], bayuhananto.2020@gmail.com[2], wayan.widi@upnvj.ac.id[3]

Abstrak, Perkembangan teknologi yang sangat pesat bertujuan untuk mempermudah pekerjaan manusia. Pada saat ini, pengguna internet tidak perlu berjalan untuk berbelanja, belajar, dan bekerja. Tetapi internet tidak sepenuhnya aman untuk digunakan. Salah satu dampak negatif yang dapat disebabkan oleh internet adalah peretasan yang dilakukan oleh orang yang tidak bertanggung jawab. Salah satu cara untuk meningkatkan keamanan dalam penggunaan aplikasi online dengan melakukan *Penetration Testing* dan *Vulnerability Assessment.* Kegiatan ini bertujuan untuk menilai keamanan suatu situs. Dalam penelitian ini, akan dilakukan *penetration testing* dan *vulnerability assessment* pada situs Sekolah Menengah Atas ABC dengan menggunakan metode NIST SP 800-115. Dalam metode ini tahapan yang akan dilakukan adalah *planning, discovery, attack* dan *reporting.* Hasil dari penelitian ini merupakan kerentanan yang ada pada situs SMA ABC dan solusi yang dapat dilakukan untuk memperbaikinya.

Kata Kunci: Situs, *penetration testing, vulnerability assessment,* NIST SP 800-115.

1 Pendahuluan

Peretasan merupakan suatu hal yang sangat sering ditemui dewasa ini. Peretasan itu sendiri adalah kegiatan yang dilakukan oleh seorang yang mampu menumbangkan atau menjatuhkan keamanan sebuah sistem dan mengubah bahkan mengabil data dari target. Sebagai contoh, pada tanggal 30 Juli 2021 situs Sekretaris Kabinet yang beralamat Setkab.go.id diretas oleh dua orang pelaku yang mengakibatkan situs tersebut tidak dapat diakses dan tampilan diubah menjadi hitam sebagai latar belakang, dan foto yang menampilkan demostran membawa bendera merah putih. Kedua pelaku tersebut diduga mempunyai motif peretasan demi mendapatkan keuntungan ekonomi dengan cara menjual *script backdoor* dari website Setkab tersebut. (Kompas, 2021)

Vulnerability Assesment atau VA adalah proses *scanning* untuk menemukan celah dan kerentanan pada sistem, *software,* dan jaringan. Proses ini juga termasuk menentukan ukuran kerentanan yang ditemukan untuk mendapatkan kerentanan yang akan menjadi prioritas. (Yaqoob, 2017)

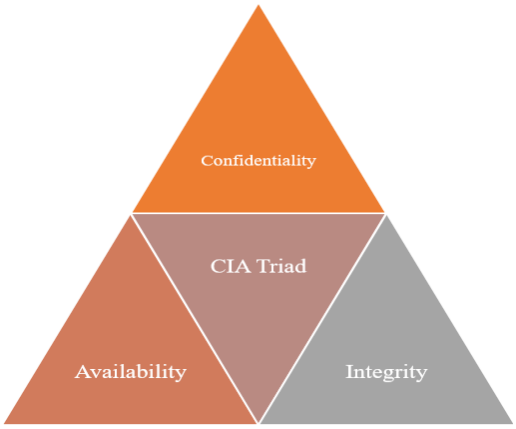
Penetration testing digunakan untuk menemukan kerentanan yang ada pada jaringan atau sistem sebelum terjadinya penyerangan. Kegiatan ini dilakukan tanpa adanya informasi seperti *username* dan *password.* Hasil dari *penetration testing* adalah laporan mengenai kerentanan-kerentanan yang telah ditemukan pada saat pengujian dilakukan dan juga menjelaskan tentang cara perbaikan jaringan atau sistem yang diuji. (Yaqoob, 2017)

Peneliti akan melakukan *Penetration Testing* dan *Vulnerability Assesment* pada situs SMA ABC. Hal tersebut bertujuan untuk mencegah penyerangan yang dapat dilakukan oleh peretas dan juga memaksimalkan keamanan yang telah dimiliki oleh situs tersebut. Peneliti akan melakukan *Penetration Testing* menggunakan metode NIST SP 800-115. Hasil yang diharapkan dari penelitian ini adalah mengetahui jenis serangan yang dapat terjadi beserta solusi yang dapat dilakukan untuk mengurangi kerentanan yang terdapat pada situs SMA ABC.

2 Tinjauan Pustaka

2.1 CIA Triad

Secara umum, keamanan data secara structural berdasarkan *confidentiality, integrity,* dan *availability* atau disebut dengan CIA Triad yang terdapat pada gambar 2.1. (Luma et al., 2018)



Gambar. 1. CIA Triad

1) *Confidentiality*
   Confidentiality atau kerahasiaan dapat diartikan bahwa aset dari sistem komputer hanya dapat diakses oleh pihak yang berwenang. Contohnya seperti membaca, melihat atau bahkan untuk mengetahui informasi tersebut ada.

2) *Integrity*
   Integrity atau integritas dapat diartikan bahwa aset hanya dapat diubah oleh pihak yang berwenang atau dengan cara yang diizinkan. Contohnya seperti menulis, mengubah, dan menghapus informasi tersebut.
3) *Availibility*
   Availability atau ketersediaan dapat diartikan bahwa aset dapat diakses oleh pihak yang berwenang. Pihak yang tidak mempunyai kewewenangan dibatasi dengan hak akses.

2.2 Penetration *Testing*

Penetration Testing digunakan untuk menemukan kerentanan yang ada pada jaringan atau sistem sebelum terjadinya penyerangan. Kegiatan ini dilakukan tanpa adanya informasi seperti *username* dan *password.* Hasil dari *penetration testing* adalah laporan mengenai kerentanan-kerentanan yang telah ditemukan pada saat pengujian dilakukan dan juga menjelaskan tentang cara perbaikan jaringan atau sistem yang diuji. (Yaqoob, 2017)

Penetration Testing adalah proses yang terstruktur untuk menguji basis komputasi organisasi yang meliputi perangkat keras atau *hardware*, perangkat lunak atau *software* dan manusia. Proses ini meliputi analisis seluruh bagian dari sistem untuk mencari kerentanan seperti konfigurasi sistem, kesalahan pada *software* dan *hardware* dan lain-lain. *Penetration testing* juga membantu mengidentifikasi tingkat kesulitan penyerang untuk menembus ke dalam sistem. (al Shebli & Beheshti, 2018)

2.3 Tahapan Penetration Testing

Berikut ini adalah beberapa langkah untuk menjalankan *penetration testing.* (Primartha, 2018)
1) *Reconnaissance*
Reconnaissance (pengintaian) atau dikenal juga dengan pengumpulan informasi adalah tahap yang paling penting dalam melakukan *penetration testing. Reconnaissance* merupakan tahap pertama dalam melakukan *penetration testing.* Ada dua jenis *reconnaissance*, yaitu pasif dan aktif (Primartha, 2018). *Reconnaissance* pasif adalah jenis yang mengumpulkan informasi awal, seperti jam kerja dan istirahat kantor. Metode yang digunakan pada pasif *reconnaissance* yaitu *social engineering. Reconnaissance* aktif adalah jenis yang menyelidiki jaringan seperti mencari *host name, IP address*, dan *service* yang berjalan pada jaringan tersebut. Pada tahap ini, semakin banyak informasi yang didapatkan oleh peneliti maka semakin besar kemungkinan untuk berhasil pada tahap-tahap selanjutnya. Tahap ini menghasilkan informasi yang selanjutnya akan digunakan dalam tahap *scanning.*

2) *Scanning and Enumeration*
Scanning adalah tahap yang dilakukan sesudah mendapatkan informasi yang bertujuan untuk membuat informasi yang didapat lebih akurat. Pada umumnya, *scanning* berkaitan dengan *port* atau bisa disebut dengan *port scanning*. Salah satu *tools* yang dapat digunakan pada tahap *scanning* adalah Nmap. *Enumeration* adalah

melakukan pendaftaran dan identifikasi pada *service* dan sumber target yang akan dilakukan *penetration testing. Enumeration* bertujuan membuat daftar *service* yang dapat dijangkau dan digunakan.

3) *Gaining Access*
Pada tahap ini, penyerang memulai penyerangan dengan cara mengekploit sistem atau jaringan untuk mendapatkan akses.

4) *Maintaining Access and Placing Backdoors*
Jika penyerang berhasil mendapatkan akses, maka penyerang harus mempertahankan akses tersebut. Selain itu penyerang juga membuat *backdoor* untuk keperluan eksploitasi yang akan datang. *Backdoor* ini tidak bisa digunakan oleh penyerang lain, hanya penyerang yang membuat *backdoor* yang dapat mengakses *backdoor* tersebut. Teknik yang biasa dipakai adalah *Trojan* dan *Rootkit.*

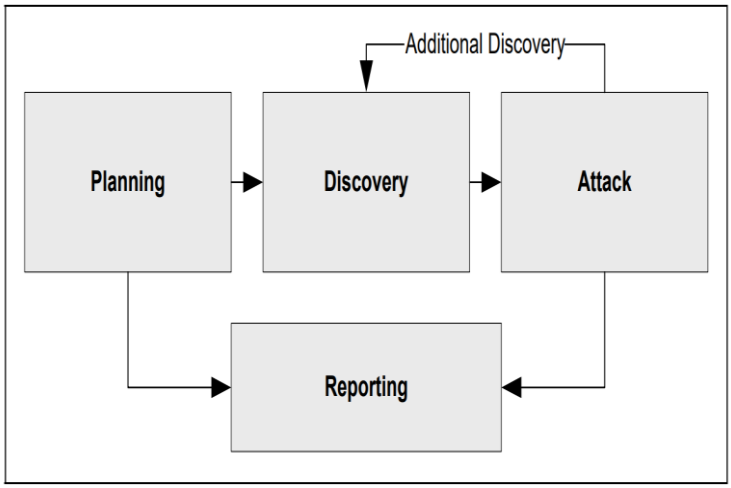
5) *Clearing Tracks*
Tahap ini adalah tahap membersihkan jejak untuk menghindari deteksi pada aktivitas penyerang. Penyerang harus berusaha mengubah atau menghapus *log* dan *record.*

2.4 *Vulnerability Assessment*

Vulnerability Assesment atau VA adalah proses *scanning* untuk menemukan celah dan kerentanan pada sistem, *software*, dan jaringan. Proses ini juga termasuk menentukan ukuran kerentanan yang ditemukan untuk mendapatkan kerentanan yang akan menjadi prioritas. (Yaqoob, 2017)

2.5 NIST SP 800-115

National Institute of Standards and Technology atau NIST adalah sebuah perusahaan keamanan informasi yang dikembangkan oleh pemerintah Amerika Serikat untuk membuat dan mendorong pengukuran, standar, dan teknologi. Peneliti menggunakan NIST SP 800-115 sebagai metodologi dalam penelitian ini. NIST SP 800-115 adalah dokumen yang menunjukan metode dan teknik yang digunakan untuk menguji kerentanan situs dalam *penetration testing* dan rekomendasi solusi dalam menangani kerentanan pada situs. Pada gambar 2.2 dapat dilihat empat tahapan *penetration testing* dalam NIST SP 800-115 yaitu *planning*, *discovery*, *attack*, dan *reporting.* (National Institute of Standards and Technology (NIST), 2020)



(Sumber: NIST SP 800-115)

Gambar. 2. NIST SP 800-115

1) *Planning*
Pada tahap *planning*, peraturan dan hasil yang diharapkan akan didiskusikan dan disetujui oleh kedua pihak, yaitu peneliti dan target. Contoh peraturan yang akan didiskusikan adalah tujuan dilakukannya *penetration testing*, *scope* atau ruang lingkup, rentang waktu pengujian, serta hasil yang diharapkan. Tidak ada pengujian yang dilakukan pada tahap ini.

2) *Discovery*
Tahap *Discovery* mempunyai dua bagian, yaitu mengumpukan informasi tentang target seperti nama Host dan informasi mengenai alamat IP, sistem, dan service dengan cara *scanning* terhadap target. Bagian kedua adalah *vulnerability analysis* atau menganalisis kerentanan yang sudah didapat ketika mengumpulkan informasi.

3) *Attack*
Pada tahap ini, peneliti membuktikan kerentan yang sudah ditemukan pada tahap kedua dengan cara eksploitasi kerentanan tersebut. Pada tahap *attack* atau penyerangan, kemungkinan penguji tidak berhasil menyerang, tetapi penguji menemukan informasi yang lebih dalam mengenai target sehingga penguji kembali ke tahap sebelumnya yaitu *discovery*. Jika ini terjadi, perlu diadakan penambahan analisis dan pengujian untuk menentukan tingkat kerentanan yang sebenarnya.

4) *Reporting*
Tahap *reporting* atau laporan adalah tahap yang menjelaskan tentang kerentanan melalui skala dan solusi penanganan kerentanan. Hasil pengujian harus didokumentasikan dan dijelaskan secara lengkap.

3 Analisis dan *Report*

3.1 *Planning*

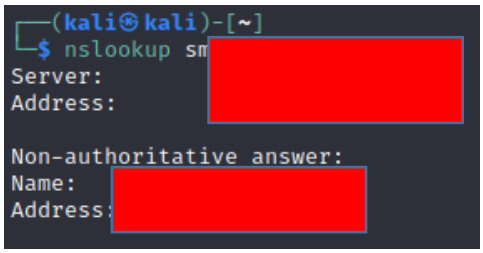
Pada tahap *planning*, penulis mendiskusikan tentang simulasi penyerangan Bersama pihak SMA ABC dan meminta izin untuk melakukan penelitian menggunakan situs SMA ABC.

3.2 *Discovery*

Pada tahap *discovery*, penulis mengumpulkan informasi seputar situs yang akan diuji. Berikut adalah *tools* yang digunakan oleh penulis pada tahap *discovery.*

3.2.1 *Nslookup*

Nslookup merupakan *tools* yang digunakan untuk mencari informasi mengenai DNS *records* pada suatu situs. Berbeda dengan *command* ping yang hanya dapat digunakan untuk melihat *records* suatu situs, Nslookup dapat memberikan informasi berupa *domain name server*, *sub-domain*, ftp, *sitebuilder*, dll. (Pandit, 2021)



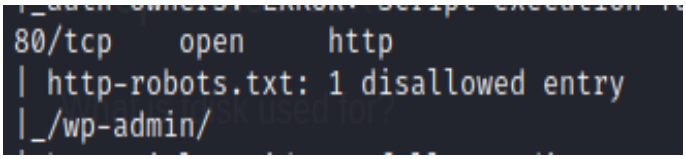
Gambar. 3. Hasil Nslookup

Pada gambar 3.1 terlihat bahwa situs SMA ABC mempunyai *IP Server* 10.21.0.1 dan *IP Address* 10.21.0.1#53. Dan juga terlihat nama *Domain* yaitu smaangkasa2.sch.id dan *IP Address 158.247.217.164.* Dengan *IP Adddress* yang sudah didapatkan bisa dilanjutkan ke tahap *discovery* selanjutnya yaitu *scanning* terhadap jaringan pada situs menggunakan NMAP.

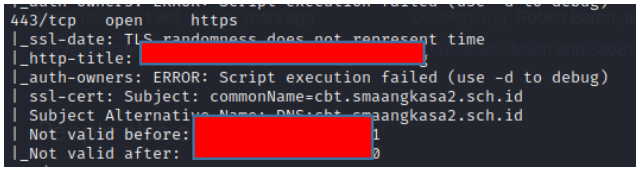
3.2.2 Nmap

NMAP merupakan alat yang berfungsi untuk menemukan *service* pada jaringan dengan cara mengirimkan paket dan akan menganalisis hasil yang didapat. (Gordon Lyon, 2009)

ISSN: 2655-139X (ONLINE)
ISSN: 0216-4221 (PRINT)



Gambar. 4. Hasil Nmap



Gambar. 5. Hasil Nmap

Pada gambar 3.2 dan 3.3 terlihat bahwa ada kemungkinan situs SMA ABC memiliki kerentanan *information disclosure* atau bocornya informasi mengenai suatu situs. Pada gambar 4.4 terlihat bahwa Sertifikat SSL akan segera kadaluarsa yang dapat mengakibatkan situs menjadi rentan dikarenakan informasi yang dikirim atau yang diterima tidak terenkripsi

3.2.3 Nikto

Hasil yang didapatkan setelah melakukan *scanning* menggunakan Nikto adalah SMA ABC menggunakan WordPress dan nginx. Berikut adalah beberapa temuan saat *scanning* menggunakan Nikto. Jenis serangan pada Nikto bersifat *non-intrusive* atau tidak mengganggu jaringan pada server atau target. Penulis menggunakan dua macam *command,* yaitu tidak menggunakan -ssl dan menggunakan -ssl.

3.2.3.1 Nikto -host SMA ABC

Command yang pertama *scanning* Nikto secara menyeluruh yaitu dengan cara tidak menambahkan *scan option* yang dimiliki oleh Nikto. Pada *command* ini, Nikto menggunakan *port* 80 untuk melakukan *scanning*. Hasil *scanning* dengan menggunakan *command Nikto -host SMA ABC* dapat dilihat pada Tabel 3.1.

Tabel 1. *Hasil Temuan Nikto*

No	Hasil Temuan	Jenis Serangan
1.	The anti-clickjacking X-Frame-Options header is not present.	Click jacking
2.	The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS	Cross-Site scripting (XSS)
3.	The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type	Cross-Site scripting (XSS)
4.	Root page / redirects to: https://smaangkasa2.sch.id/	Phising & Cross-Site Scripting
5.	No CGI Directories found	Path Traversal
6.	Uncommon header 'x-redirect-by' found, with contents: WordPress. Uncommon header 'link' found, with contents:<https://smaangkasa2.sch.id/wp-json/>; rel="https://api.w.org/" Uncommon header 'x-tec-api-root' found, with contents: https://smaangkasa2.sch.id/wp-json/tribe/events/v1/	Cross-Site Scripting (XSS)

	Uncommon header 'x-tec-api-origin' found, with contents: https://smaangkasa2.sch.id Uncommon header 'x-tec-api-version' found, with contents: v1	
7.	Cookie PHPSESSID created without the httponly flag	Cross-Site Scripting

3.2.3.2 Nikto -host SMA ABC -ssl

Command yang kedua *scanning* Nikto dengan menggunakan *force ssl* atau *secure sockets layer*. Pada *command* ini, Nikto menggunakan *port* 443 untuk melakukan *scanning*. Hasil *scanning* dengan menggunakan *command Nikto -host SMA ABC -ssl* dapat dilihat pada Tabel 3.2.

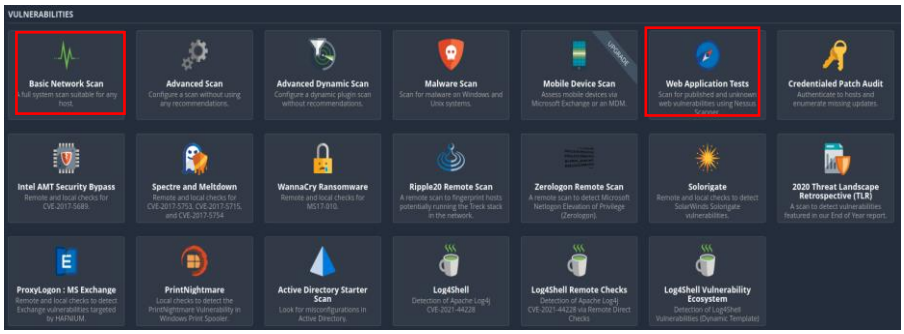
Tabel 2. Hasil temuan Nikto

No	Hasil Temuan	Jenis Serangan
1.	Uncommon header 'x-tec-api-version' found, with contents: v1 Uncommon header 'x-tec-api-root' found, with contents: https://smaangkasa2.sch.id/wp-json/tribe/events/v1/ Uncommon header 'x-tec-api-origin' found, with contents: https://smaangkasa2.sch.id Uncommon header 'link' found, with contents:<https://smaangkasa2.sch.id/wp-json/>; rel="https://api.w.org/" Uncommon header 'x-tec-api-origin' found, with contents: https://smaangkasa2.sch.id Uncommon header 'x-redirect-by' found, with contents: WordPress. Cookie ci_session_zyacbt created without the secure flag	Cross-Site Scripting (XSS)
2.	The site uses SSL and the Strict-Transport-Security HTTP header is not defined. The site uses SSL and Expect-CT header is not present.	Man-in-The-Middle (MiTM)
3.	Cookie PHPSESSID created without the secure flag Cookie PHPSESSID created without the httponly flag. Cookie wordpress_test_cookie created without the httponly flag	Cross-Site Scripting (XSS)
4.	No CGI Directories found	Path Traversal
5.	Entry '/wp-admin/' in robots.txt returned a non-forbidden or redirect HTTP code (302) /wp-login.php: WordPress login found	Information Disclosure & Brute Force
6.	"robots.txt" contains 2 entries which should be manually viewed.	Information Disclosure
7.	The Content-Encoding header is set to "deflate" this may mean the server is vulnerable to the BREACH attack.	Breach Attack
8.	OSVDB-3092: /foto/: This might be interesting…	Information Disclosure

9.	/wp-content/plugins/akismet/readme.txt: The WordPress Akismet plugin 'Tested up to' version usually matches the WordPress version. /wp-links-opml.php: This WordPress script reveals the installed version. /wp-app.log: WordPress' wp-app.log may leak application/system details. /wordpresswp-app.log: WordPress 'wp-app.log may leak application/system details. /redis/config.json : redis config file found. It may contain sensitive information. OSVDB-3092: /license.txt: License file found may identify site software. /: A WordPress installation was found. /wordpress: A wordpress installation was found.	Information Disclosure

3.2.4 Tenable Nessus

Tenable Nessus merupakan *scanning tools* yang digunanakan untuk melakukan pemindaian pada suatu aset dan memberikan sebuah *alert* jika terdeteksi adanya kerentanan pada aset tersebut yang memiliki kemungkinan digunakan oleh peretas untuk melakukan tindak kejahatan online(Pandit & Pandit, 2021). Pada Tenable Nessus terdapat beberapa *template* yang bisa dipakai untuk mendukung kegiatan *scanning*, berikut adalah contoh dari *template* yang ada pada Tenable Nessus. Pada gambar 3.4 di bawah, penulis menggunakan dua *template*, yaitu *Basic Network Scan* dan *Web Application Tests.*



Gambar. 6. Template Nessus

3.2.4.1 *Basic Network Scan*

Basic Network Scan merupakan *template* yang hanya melakukan *scanning* pada bagian jaringan atau *network* yang dimiliki oleh situs SMA ABC. Ditemukan dua kerentanan yang berstatus *high* dan *medium.* Hasil *scanning* menggunakan Nessus *Network Scan* dapat dilihat pada tabel 3.3.

Tabel 3. Hasil temuan Nessus

Nama Kerentanan	Tingkat Resiko
DNS Server Spooled Request Amplification DDOS	High
DNS Server Recursive Query Poisoning Weakness	Medium
Common Platform Enumeration (CPE)	Informational
DNS Server Detection	Informational

Device Type	Informational
Host Fully Qualified Name (FQDN) Resolution	Informational
Nessus SYN Scanner	Informational
Nessus SYN Informational	Informational
OS Identification	Informational
OS Security Patch Assessment Available	Informational
Open Port Re-check	Informational
SSH Algorithms Language Support	Informational
SSH Password Authentication Accepted	Informational
SSH Protocol Version Supported	Informational
SSH SHA-1 HMAC Algorithms Enabled	Informational
SSH Server Type and Version Information	Informational
Service Detection	Informational
Service Detection (GET request)	Informational
Traceroute Information	Informational
Unknown Service Detection: Banner Retrieval	Informational

3.2.4.2 *Web Application Tests*

Web Application Tests adalah *template* yang digunakan untuk melakukan *scanning* pada HTTP atau HTTPS. *Web Application Tests* bisa mendeteksi kerentanan yang umum pada *Web Application* seperti *SQL Injection, Cross-site scripting (XSS), Directory Traversal, Remote File Inclusion,* dan *command execution.* Pada tahap *scanning web application test,* ditemukan dua kerentanan yang berstatus *medium*. Hasil *scanning* menggunakan Nessus *Network Scan* dapat dilihat pada tabel 3.4.

Tabel 4. Hasil temuan Nessus

Nama Kerentanan	Tingkat Resiko
HSTS Missing From HTTPS Server (RFC 6796)	Medium
WordPress User Enumeration	Medium
CGI Generic Injectable Parameter	Informational
CGI Generic Tests Timeout	Informational
HSTS Missing From HTTPS Server	Informational
HTTPS Cookie ‘scure’ Property Transport Mismatch	Informational
HTTP Methods Allowed (per directory)	Informational
HTTP Server Type and Version	Informational

HyoerText Transfer Protocol (HTTP) Information	Informational
HyperText Transfer Protocol (HTTP) Redirect Information	Informational
Misiing or Permissive Content-Security-Policy frame-ancestors HTTP Response Header	Informational
Nessus SYN Scanner	Informational
Web Application Cookies Not Marked HttpOnly	Informational
Web Application Cookies Not Marked Secure	Informational
Web Application Sitemap	Informational
Web Server Directory Enumeration	Informational
Web Server no 404 Error Code Check	Informational
Web server robots.txt Informational Disclosure	Informational
Nginx HTTP Server Detection	Informational

3.2.5 OWASP ZAP

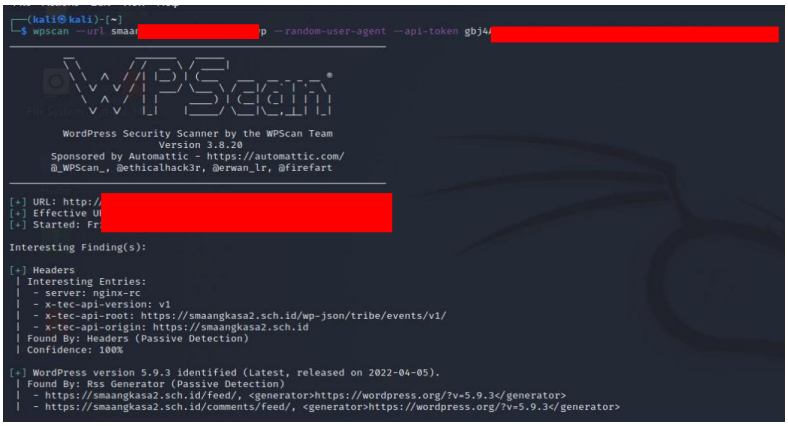
OWASP ZAP dibuat oleh organisasi bernama OWASP yang memiliki fungsi untuk melakukan *crawl* pada suatu situs dengan menggunakan *traditional spider* dan juga *AJAX spider* dan melakukan penyerangan untuk menemukan kerentanan yang terdaftar pada OWASP Top Ten(Jakobsson & Häggström, 2022). Penulis menggunakan *Automated Scanning*. Berikut ini adalah tabel yang menjelaskan mengenai temuan menggunakan OWASP ZAP. Hasil *scanning* dengan menggunakan OWASP ZAP dapat dilihat pada tabel 3.5.

Tabel 5. Hasil temuan OWASP ZAP

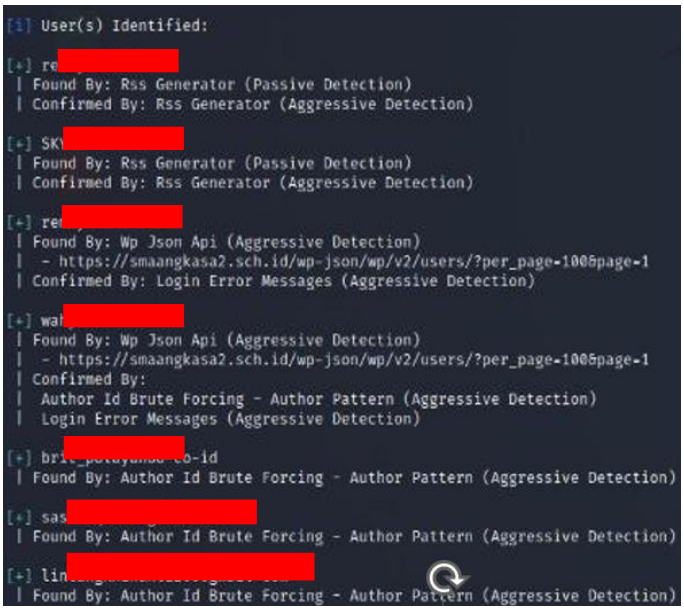
Nama Kerentanan	Tingkat Resiko
Path Traversal	High
Multiple X-Frame-Options Header Entries	Medium
Absence of Anti-CSRF Tokens	Low
Cookie No HttpOnly Flag	Low
Cookie Without Secure Flag	Low
Cookie Without SameSite Attribute	Low
Cross-Domain JavaScript Source Fule Inclusion	Low
Incomplete or No Cache-control Header Set	Informational
Timestamp Disclosure-Unix	Informational

3.2.6 WPScan

WPScan merupakan *tool* khusus untuk melakukan *scanning* pada keamanan WordPress. WPScan digunakan oleh *pentester* untuk melakukan *Vulnerability assessment* dan juga *Penetration Testing*. Hasil scanning dengan menggunakan WPScan dapat dilihat pada gambar 3.5.



Gambar. 7. Hasil WPScan



Gambar. 8. Detail WPScan

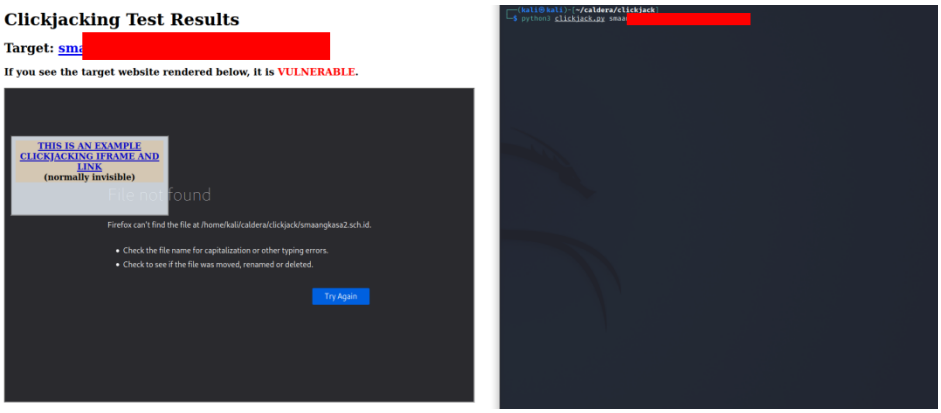
Pada gambar 3.6 terlihat *username* yang sudah pernah dibuat untuk mengakses situs SMA ABC. Temuan ini kemungkinan besar bisa dijadikan daftar untuk membuat *payload brute force*. Berikut ini adalah tabel hasil temuan *username* pada SMA ABC.

3.3 *Attack*

Pada tahap ini, penulis akan melakukan simulasi penyerangan terhadap situs SMA ABC dengan informasi yang didapat pada tahap sebelumnya yaitu tahap *discovery*. Berikut adalah daftar simulasi penyerangan yang akan dilakukan.

3.3.1 *Clickjacking*

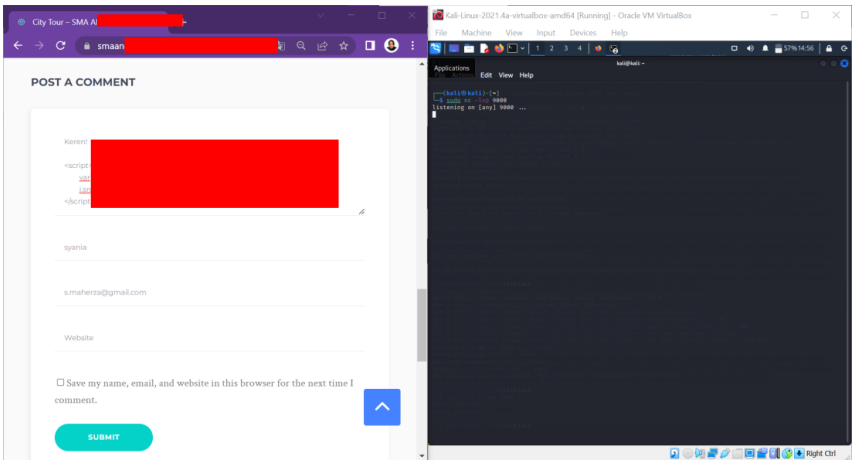
Pada tahap ini penulis melakukan simulasi penyerangan *clickjacking*. Pada gambar 3.7 terlihat hasil dari simulasi penyerangan yang dilakukan adalah situs SMA ABC tidak rentan terhadap serangan *clickjacking*.



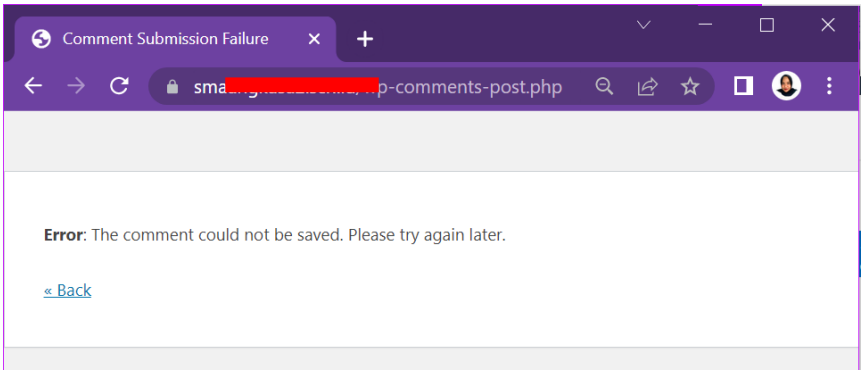
Gambar. 9. Hasil Clickjacking

3.3.2 XSS

Pada tahap ini, penulis melakukan simulasi serangan XSS atau *Cross Site Scripting*. Penulis melakukan simulasi menggunakan XSS dengan jenis stored atau disimpan. Jenis ini bisa digunakan untuk mengambil *cookie* dari *session* user. Pada simulasi ini, terlihat bahwa terjadi error pada situs SMA ABC jika berkomentar pada halaman tersebut. Pada gambar 3.8 dan 3.9 terlihat hasil dari simulasi ini adalah kerentanan ini tidak dapat diuji karena terjadi error.



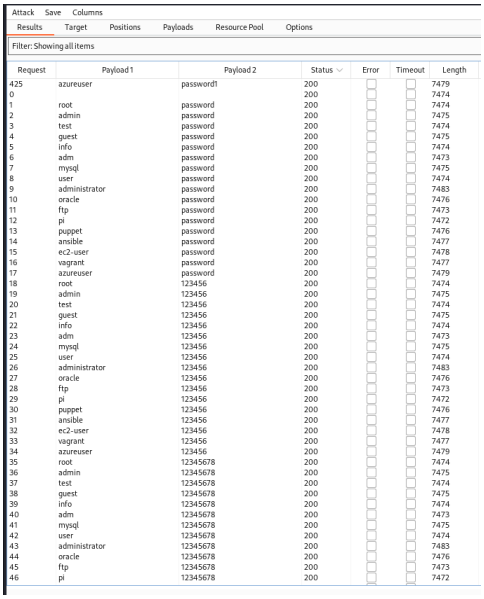
Gambar. 10. Hasil XSS



Gambar. 11. Hasil XSS

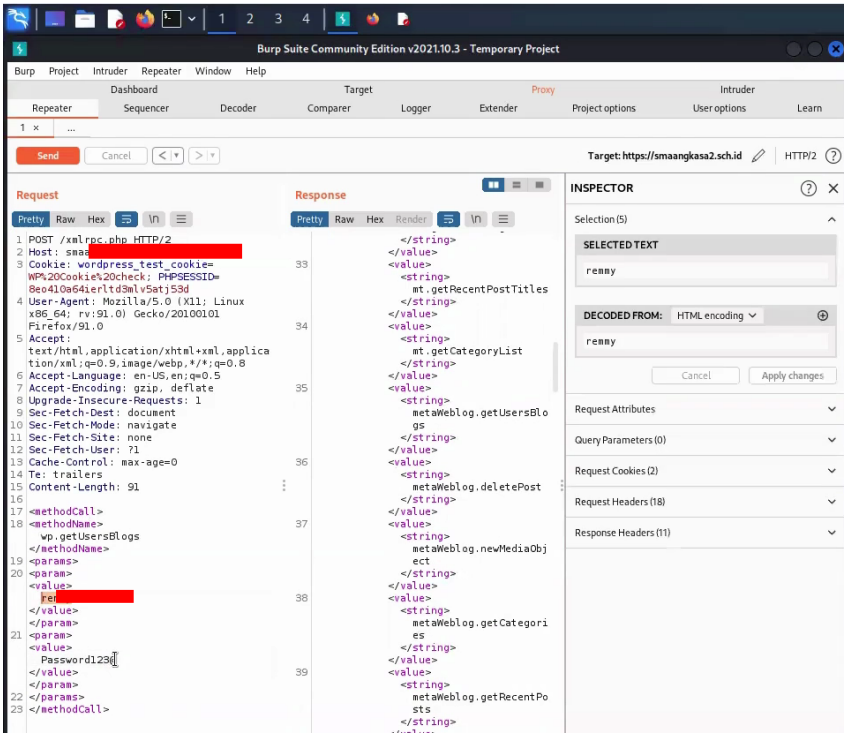
3.3.3 Brute Force

Pada tahap ini, penulis melakukan simulasi penyerangan *brute force* atau menebak *username* dan *password* secara paksa pada halaman login admin WordPress. Penulis menggunakan burp suite untuk melakukan simulasi penyerangan. Pada gambar 3.10 bisa dilihat hasil dari simulasi penyerangan *brute force* adalah tidak ditemukannya *username* dan *password* yang sesuai. Penulis menggunakan *payload* yang bisa diunduh pada github dan penulis juga menambahkan username yang ditemukan pada WPScan.



Gambar. 12. Hasil Brute force

Penulis juga menemukan bahwa situs SMA ABC mempunyai xmlrpc.php yang dapat dijadikan alat simulasi untuk mencoba Brute Force. Xmlrpc.php digunakan untuk bertukar informasi antara sistem komputer dan jaringan, yang memudahkan pengguna untuk mengakses WordPress menggunakan WordPress via mobile atau via weblog client seperti Windows Live Writer.



Gambar. 13. Hasil xmlrpc.php

ISSN: 2655-139X (ONLINE)
ISSN: 0216-4221 (PRINT)

Pada gambar 3.11 ditemukan bahwa serangan brute force bisa dilakukan melalui xmlrpc.php milik SMA ABC.

3.3.4 Information Disclosure

Pada simulasi penyerangan ini, terdapat beberapa informasi yang seharusnya tidak ditampilkan pada situs SMA ABC karena bisa menyebabkan terjadinya celah keamanan pada situs SMA ABC.

```
=== Akismet Spam Protection ===
Contributors: matt, ryan, andy, mdawaffe, tellyworth, josephscott, lessbloat, eoigal, cfinke, automattic, jgs, procifer, stephdau
Tags: comments, spam, antispam, anti-spam, contact form, anti spam, comment moderation, comment spam, contact form spam, spam comments
Requires at least: 5.0
Tested up to: 6.0
Stable tag: 4.2.4
License: GPLv2 or later

The best anti-spam protection to block spam comments and spam in a contact form. The most trusted antispam solution for WordPress and WooCommerce.

== Description ==

Akismet checks your comments and contact form submissions against our global database of spam to prevent your site from publishing malicious content. You can review the
comment spam it catches on your blog's "Comments" admin screen.

Major features in Akismet include:

* Automatically checks all comments and filters out the ones that look like spam.
* Each comment has a status history, so you can easily see which comments were caught or cleared by Akismet and which were spammed or unspammed by a moderator.
* URLs are shown in the comment body to reveal hidden or misleading links.
* Moderators can see the number of approved comments for each user.
* A discard feature that outright blocks the worst spam, saving you disk space and speeding up your site.

PS: You'll be prompted to get an Akismet.com API key to use it, once activated. Keys are free for personal blogs; paid subscriptions are available for businesses and
commercial sites.

== Installation ==

Upload the Akismet plugin to your blog, activate it, and then enter your Akismet.com API key.

1, 2, 3: You're done!

== Changelog ==

= 4.2.4 =
*Release Date - 20 May 2022*
```

Gambar. 14. readme.txt

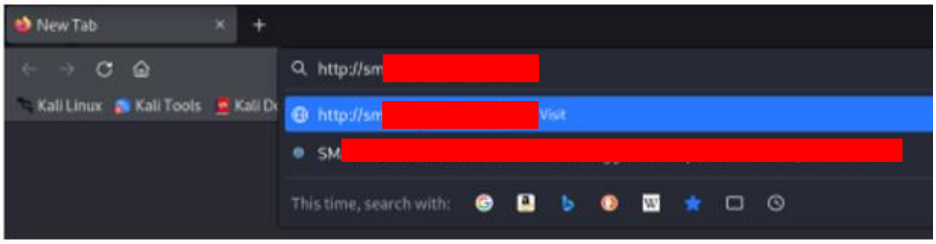
```
WordPress - Web publishing software

Copyright 2011-2022 by the contributors

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA  02110-1301  USA

This program incorporates work covered by the following copyright and
permission notices:

  b2 is (c) 2001, 2002 Michel Valdrighi - https://cafelog.com

  Wherever third party code has been used, credit has been given in the code's
  comments.

  b2 is released under the GPL

and

  WordPress - Web publishing software

  Copyright 2003-2010 by the contributors

  WordPress is released under the GPL
```

Gambar. 15. Lisence.txt

Pada gambar 3.12 dan gambar 3.13 adalah file Readme.txt dan lisence.txt yang merupakan celah *information disclosure* yang dimiliki oleh SMA ABC. Readme.txt adalah informasi mengenai instalasi situs dan sejarah perubaha-perubahan yang dilakukan pada situs SMA ABC. Lisence.txt adalah informasi mengenai lisensi SMA ABC.

3.3.5 HSTS Missing

Pertama penulis mencoba untuk mengakses situs dengan menggunakan htpp dikarenakan berdasarkan hasil scanning dengan menggunakan Nessus ditemukan bahwa situs rentan terhadap HSTS yang dapat dilihat pada gambar 3.14.

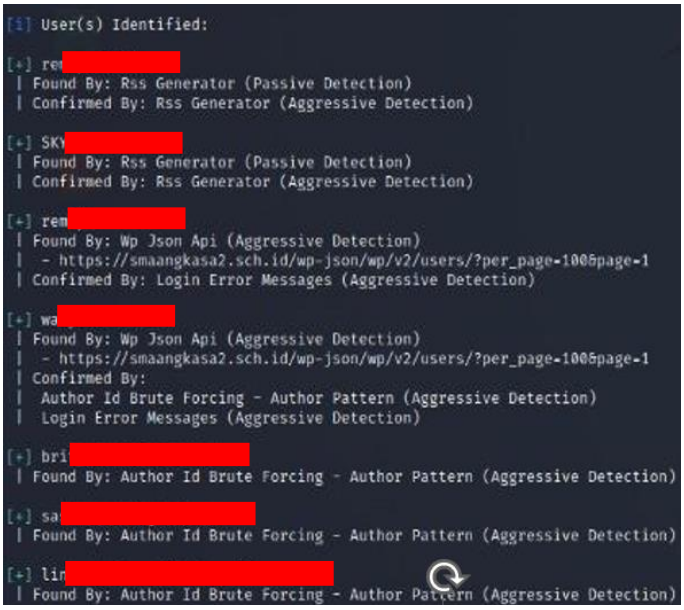


Gambar. 16. Hasil HSTS missing

Situs SMA ABC tidak dapat diakses tanpa menggunakan HTTPS dan langsung *redirect* menggunakan HTTPS.

3.3.6 WordPress User Enumeration

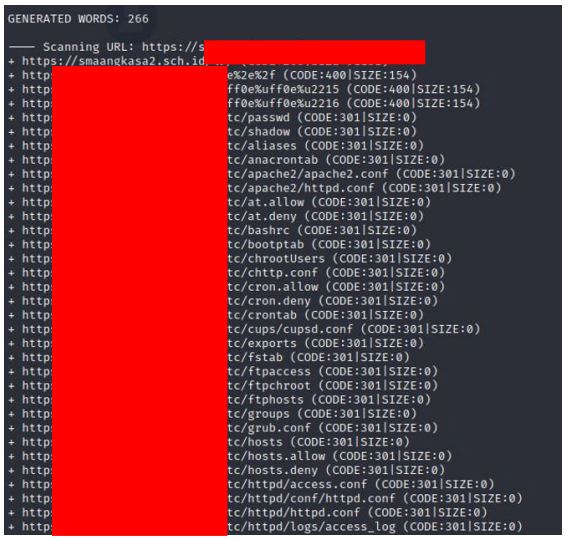
Kerentanan ini bisa ditemukan Ketika penulis menggunakan WPScan. Setiap *username* yang dibuat atau dipakai pada situs SMA ABC terlihat Ketika menggunakan WPScan dengan *plugin* '-e'. Dibawah ini pada gambar 3.15 merupakan hasil temuan Wordpress User Enumeration.



Gambar. 17. Hasil WP User Enumeration

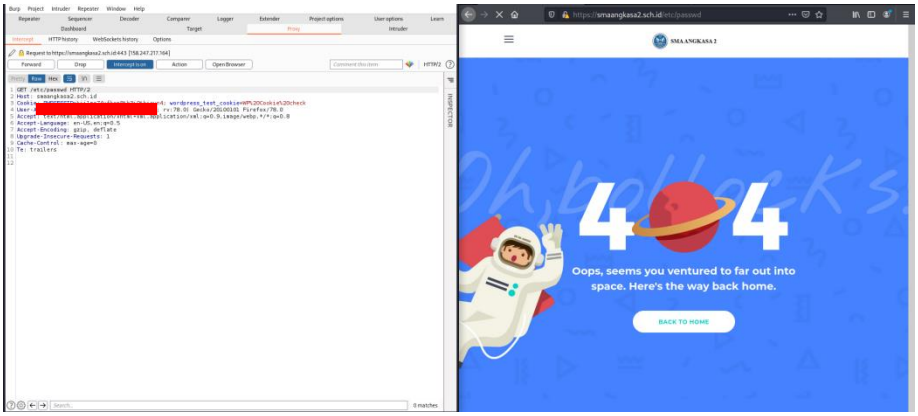
3.3.7 Path Traversal

Penulis dalam menguji kerentanan Path Traversal menggunakan *tools* dirb untuk melakukan scanning terhadap situs dengan menggunakan payload path traversal. Berikut ini merupakan hasil dari path traversal yang penulis lakukan. Pada kerentanan Path Traversal, terdapat *HTTP Status Code*, atau kode status pada HTTP. Kode status terdiri dari 3 digit nomor yang dikirim oleh web server kepada client, mengindikasikan *request* berhasil atau tidak berhasil terkirim. Pada tabel 4.8 akan menjelaskan kode status yang muncul pada saat pengujian Path Traversal.



Gambar. 18. Hasil Path traversal

Gambar 3.16 dan 3.17 merupakan hasil scanning yang mayoritasnya memiliki response 301 yang akan redirect ke page not found atau error 404 dan juga ada response 200 tetapi memiliki response error 404 yang bisa disebut bahwa kerentanan path traversal adalah false positive.



Gambar. 19. Hasil Path traversal

3.4 *Reporting*

Tahap *reporting* adalah tahap terakhir dari metode NIST SP 800-115. Pada tahap sebelumnya, penulis sudah mengumpulkan temuan kerentanan dan melakukan simulasi penyerangan terhadap kerentanan yang ditemukan. Berikut adalah tabel kerentanan yang telat ditemukan. Hasil *reporting* dapat dilihat pada tabel 3.6.

Tabel 6. Tabel kerentanan

No	Nama Kerentanan	Dampak	Status Valid	Solusi
1.	Clickjacking	Berdampak ketika *user* mengakses sebuah situs yang sudah ditanam dengan clickjacking, user adakan membuka situs palsu yang dapat mengambil data milik user.	Tidak Valid	Untuk menutup celah kerentanan ini, dapat dilakukan dengan cara menggunakan header X-Frame-Options: sameorigin atau deny. Cara selanjutnya adalah dengan menambahkan filter add_filter('wp_anti_clickjack','__return_false"); . Filter ini digunakan pada file functions.php

2.	Cross Site Scripting (XSS)	Dampak yang dapat diakibatkan oleh Cross Site Scripting adalah penyerang dapat mengambil data sensitive dari situs dan dapat menyelipkan payload kedalam situs.	Tidak Valid	Untuk menutup kerentanan ini, dapat dilakukan dengan cara: 1. Melakukan filtering karakter-karakter khusus, seperti "<" , ">" ,"&". 2. Teknik validasi yang menjamin hanya input yang tepat yang akan dipilih 3. Teknik encoding yang hamper sama dengan filtering hanya dengan menggunakan encoding data tidak akan hilang.
3.	Brute Force	Dampak dari serangan brute force adalah penyerang dapat memiliki username dan password dari user dan mengakses akun tersebut secara bebas	Valid	Untuk menutup celah kerentanan ini, dapat dilakukan dengan cara: 1. Batasi login ke situs 2. Pastikan Password sulit untuk di tebak. 3. Mengganti URL login pada situs 4. Menggunakan captcha 5. Menggunakan 2FA atau 2 *Factor Authentication*
4.	Information Disclosure	Dampak dari Information Disclosure adalah penyerang dapat mengetahui celah informasi yang tidak seharusnya ditunjukan oleh situs.	Valid	Untuk menutup celah kerentanan ini, maka dapat dilakukan dengan cara menyembunyikan semua informasi yang tidak seharusnya terlihat seperti versi pada nginx, wordpress, php.
5.	HSTS Missing	Dampak dari HSTS Missing adalah data yang dikirim bisa dilihat atau ditangkap oleh penyerang yang dapat mengakibatkan serangan Man-in-The-Middle.	Tidak Valid	Untuk menutup celah kerentanan ini, dapat dilakukan dengan cara set HTTP langsung diarahkan ke HTTPS.
6.	Wordpress User Enumeration	Dampak yang dapat diakibatkan oleh WordPress User Enumeration adalah penyerang dapat mengetahui username yang digunakan pada situs tersebut. Kerentanan ini merupakan langkah awal dari brute force.	Valid	Untuk menutup celah kerentanan ini, dapat menggunakan fail2ban untuk memblokir serangan dari firewall, bisa digunakan untuk VPS (Virtual Private Server). Fail2ban ini juga dapat memblokir serangan DDOS.
7.	Path Traversal	Dampak yang diakibatkan oleh Path Traversal adalah penyerang dapat mengakses dokumen yang tidak seharusnya diakses, contohnya seperti dokumen password.	Tidak Valid	Untuk menutup celah kerentanan ini, dapat dilakukan dengan 1. Teknik validasi yang menjamin hanya input yang tepat yang akan dipilih. 2. Situs harus menggunakan filter untuk memblokir input yang mencurigakan dari user.

5 Kesimpulan

Dari hasil temuan dan simulasi yang dilakukan, dapat disimpulkan bahwa:

1. Dalam melakukan kegiatan *penetration testing* pada situs SMA ABC, penulis menemukan tiga kerentanan yang sudah tervalidasi setelah dilakukan *penetration testing*, yaitu yang pertama adalah *Brute Force* dengan melakukan *intercerpt* menggunakan *tools burp suite* terhadap situs SMA ABC dan menggunakan XML-RPC. Kerentanan yang kedua adalah *Information Disclosure* dan yang ketiga adalah *Wordpress User Enumeration* dengan menggunakan WPScan.
2. Dampak yang ditimbulkan dari tiga kerentanan tersebut adalah yang pertama jika penyerang berhasil melakukan *Brute Force*, maka penyerang dapat mengakses akun admin dan dapat mengubah situs bahkan dapat mengambil alih situs tersebut. Dampak dari kerentanan yang kedua adalah jika penyerang mendapatkan informasi yang seharusnya tidak diperlihatkan oleh situs. Dampak dari kerentanan yang ketiga adalah jika penyerang berhasil mendapatkan username dari admin yang ada pada situs, maka ini adalah langkah awal dari penyerangan *Brute Force*.

3. Solusi untuk kerentanan *Brute Force* adalah yang pertama membuat Batasan login pada situs, jika user lima sampai sepuluh kali gagal login, maka user tersebut harus menunggu selama beberapa waktu untuk mencoba login kembali. Kedua adalah pastikan kata sandi yang digunakan admin sulit untuk ditebak. Yang ketiga adalah menyembunyikan halaman login pada situs. Yang terakhir adalah menggunakan captcha dan 2FA atau *2 Factor Authentication.* Solusi untuk *Information Disclosure* adalah menyembunyikan informasi-informasi yang tidak seharusnya ditemukan user biasa. Solusi untuk Wordpress User Enumeration adalah menggunakan fail2ban untuk memblokir serangan dari firewall.

Referensi

[1] H. M. Z. A. Shebli and B. D. Beheshti, "A study on penetration testing process and tools," *2018 IEEE Long Island Systems, Applications and Technology Conference (LISAT),* pp. 1-7, 2018.

[2] G. F. Lyon, "Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning," *Nmap Project*, 2008.

[3] A. Jakobsson and I. Häggström, "Study of the techniques used by OWASP ZAP for analysis of vulnerabilities in web applications," 2022.

[4] D. E. Nugraheny, "Kompas.com," 2021. [Online]. Available: https://nasional.kompas.com/read/2021/07/31/11283171/website-sekretariat-kabinet-diretas-polri-hingga-bin-turun-tangan.

[5] A. Luma, B. Abazi, B. Selimi and M. Hamiti, "Comparision of Maturity Model Frameworks in Information Security and Their Implementation," *in Proceedings International Conf on Engineering Technologies (ICENTE'18)*, 2018.

[6] K. Scarfone, M. Souppaya, A. Cody and A. Orebaugh, "NIST," 2008. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-115/final. [Accessed 2020].

[7] P. Pandit, "A Study of Network Reconnaissance Tools," 2021.

[8] P. Pandit, "Nessus: Study of a Tool to Assess Network Vulnerabilities," 2021.

[9] R. Primartha, "Security jaringan komputer berbasis CEH," *Bandung: Informatika Bandung*, 2018.

[10] I. Yaqoob, S. A. Hussain, S. Mamoon, N. Naseer, J. Akram and A. U. Rehman, "Penetration Testing and Vulnerability Assessment," *Journal of Network Communications and Emerging Technologies (JNCET),* vol. 7, no. 8, pp. 10-18, 2017.

ISSN: 2655-139X (ONLINE)
ISSN: 0216-4221 (PRINT)