Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)

Journal Homepage: http://journal.lembagakita.org/index.php/jtik

Analysis *Vulnerability Website Baleomolcreative* dengan Metode *Penetration Testing Execution Standard* & *Vulnerability Assessment* Pada *Http Response Header Field*

Henokh Kurniawan [1*], Erwien Christianto [2]

[1*,2] *Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Kota Salatiga, Provinsi Jawa Tengah, Indonesia.*

a r t i c l e i n f o

Article history:
Received 2 February 2024
Received in revised form
14 March 2024
Accepted 1 May 2024
Available online July 2024.

DOI:
https://doi.org/10.35870/jtik.v8i3.2202.

Keywords:
Vulnerability; Penetration; Owaspzap; Nikto; Nmap; Web.

Kata Kunci:
Vulnerability; Penetration; Owaspzap; Nikto; Nmap, Web.

a b s t r a c t

This research will analyze web security and how to find out whether there is a vulnerability or what could be called a vulnerability to enter gaps in the Baleomolcreative web, making the web unsafe. In analyzing whether there are vulnerabilities, the Penetration Testing Execution Standard and Vulnerability Assessment methods are used to determine whether there are gaps or vulnerabilities in the Baleomolcreative website that can be exploited by external parties. This method uses tools such as Owasp ZAP, Nikto, and Nmap which can be used to perform vulnerability scanning on a website. In this research, we succeeded in identifying 3 levels of vulnerability on the Baleomolcreative website, namely medium, low, and informational, with a total of 18 alerts generated from notifications on Owasp Zap. The scanning process includes vulnerability testing such as Content Security Policy, Anti-clickjacking Header, Dangerous JS Functions, Permissions Policy, and others.

a b s t r a k

Penelitian ini akan Menganalisa keamanan web serta cara untuk mengetahui apakah ada vulnerability atau bisa disebut sebagai kerentanan untuk memasuki celah pada web Baleomolcreative sehingga menjadikan web tersebut tidak aman. Dalam menganalisa apakah adanya vulnerability, Metode Penetration Testing Execution Standard dan Vulnerability Assessment digunakan untuk mengetahui apakah terdapat celah atau kerentanan terhadap situs web Baleomolcreative yang dapat dieksploitasi oleh pihak luar. Metode ini menggunakan alat-alat seperti Owasp ZAP, Nikto, dan Nmap yang dapat digunakan untuk melakukan pemindaian kerentanan pada suatu website. Pada penelitian ini berhasil mengidentifikasi 3 tingkat kerentanan pada web Baleomolcreative, yaitu medium, low, dan informational, dengan total 18 alert yang dihasilkan dari notifikasi pada Owasp Zap. Proses scanning mencakup pengujian kerentanan seperti Content Security Policy, Anti-clickjacking Header, Dangerous JS Functions, Permissions Policy, dan lainnya.

Corresponding Author. Email: 672018151@student.uksw.edu [1].

ACM Computing Classification System (CCS)
EBSCOhost
Communication and Mass Media Complete (CMMC)

© E-ISSN: 2580-1643.

1. Latar Belakang

Perkembangan teknologi saat ini telah mengubah proses pembuatan sistem penyedia layanan dalam jaringan umum atau internet. Sangat penting untuk memiliki jaringan komputer untuk menunjang operasi perusahaan. Suatu perusahaan harus mempertimbangkan aspek keamanan infrastruktur jaringan LAN dan WAN. Mereka harus membeli sistem keamanan jaringan untuk melindungi seluruh aset dari ancaman kejahatan *virtual* seperti pencuri atau *virus*. Keamanan jaringan komunikasi sangat penting untuk memberikan layanan yang aman bagi pengguna atau klien selama pengiriman data atau pesan terus menerus. Banyaknya *cyber security incident* akibat *cyber attack* mengalami peningkatan pada tahun 2021. Menurut CPR (*Check Point Research*) misalnya, pada tahun 2021, jumlah *cyber attack* pada jaringan perusahaan setiap minggunya mengalami peningkatan sebesar 50%. Begitu pula menurut BSSN (Badan Siber dan Sandi Negara), pada Januari sampai Mei 2021 terdapat cyber attack dengan jumlah sekitar 480 juta di Indonesia. Padahal, sebelumnya pada tahun 2020, jumlah *cyber attack* yang tercatat sekitar 495 juta. Dengan kata lain, tak sampai semester pertama tahun 2021 selesai, jumlah *cyber attack* yang tercatat oleh BSSN di Indonesia sudah sangat mendekati jumlah sepanjang tahun 2020 [1].

Dengan semakin berkembangnya teknologi dan Internet, lalulintas pergerakan sistem informasi mengarah pada penggunaan mereka sebagai basis. Beberapa sistem tetap menggunakan basis *web* sebagai dasar untuk sistem informasinya yang dipasang di jaringan Internet. Untuk memungkinkan semua pengguna, karyawan, dan pengguna melakukan pekerjaan secara *online*, basis *web* harus digunakan. Karena itu, kebijakan dan praktik keamanan sistem *web* yang diterapkan mempengaruhi keamanan sistem informasi yang berbasis *web* dan teknologi Internet. Agar sistem Jaringan Komputer tidak terganggu atau diserang oleh Peretas, sangat diperlukan sebuah sistem keamanan jaringan yang dapat mengamankan serta mencegah serangan dari Peretas tersebut. Mariusz Stawowski (2007) dalam jurnalnya yang berjudul "*ThePrinciples of Network Security Design*" mengatakan bahwa keamanan jaringan yang utama sebagai perlindungan bagi sumber daya sistem terhadap ancaman yang berasal dari luar jaringan [16].

Administrator jaringan harus memperhatikan keamanan jaringan dan *server* yang diakses oleh pengguna ini karena dengan memberikan akses jaringan umum ke jaringan LAN, sehingga orang luar yang tidak memiliki kepentingan membuka celah. Jika server dan jaringan lokal terhubung ke Internet dan akses web server tersedia untuk umum, keamanan jaringan harus ditingkatkan. Keamanan jaringan seringkali diabaikan, dan pengamanan sistem informasi baru hanya diketahui setelah bencana terjadi. Teknologi paling canggih pun dapat membahayakan perusahaan atau organisasi itu sendiri jika pengamanan sistem informasi dan jaringan kurang baik. *Vulnerability* adalah suatu kelemahan yang menjadi ancaman nilai *integrity, confidentiality*, dan *availability* dari suatu aset. *Penetration testing* atau yang lebih dikenal dengan sebutan *pentest* adalah salah satu metode yang dapat digunakan untuk melakukan analisa dan evaluasi terhadap suatu jaringan komputer. Selain itu, *vulnerability* juga perlu dilakukan untuk prosedur mitigasi. Sumber teori serta hasil daripada analisa dari penelitian sebelumnya menjadi hal yang sangat dibutuhkan pada penelitian ini. Penelitian sebelumnya dapat dijadikan sebagai pertimbangan serta panduan untuk melakukan *Analysis Vulnerability Website* Baleomolcreative dengan Metode *Penetration Testing Execution Standard & Vulnerability Assessment* pada *Http Response Header Field*.

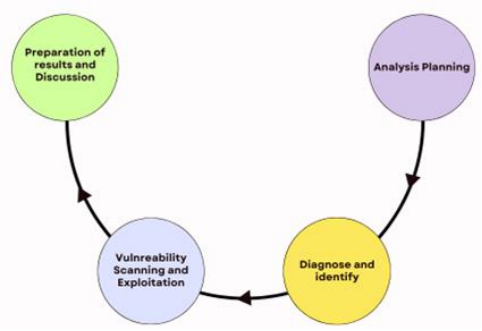
Darojat, Sediyono dan Sembiring, mengatakan dalam penelitiannya, Hasil penelitian menunjukkan ditemukan persamaan penilaian dalam hal kategori level ancaman dan jumlah kerentanan menggunakan kedua alat *web vulnerability scanner*, dan hasil yang berbeda terdapat pada durasi, kecepatan pemindaian, dan jenis temuan kerentanan. Kedua parameter *OWASP* dapat memberikan petunjuk dan penjelasan kompleks untuk membantu pengembang atau pihak pemerintah daerah melakukan pengambilan keputusan mengenai keamanan informasi pada *web e-government* yang dikelola [8]. Utoro, Nugroho, Meinawati dan Widianto, mengatakan dalam penelitiannya, Metode PTES ini dapat dijadikan sebagai standar penilaian keamanan aplikasi yang berbasis *web* pada *website e-learning* di alamat *belajar.smkn1cibatu.sch.id* yang terdiri dari tujuh tahapan atau fase yaitu dimulai dari tahap *pre-engagement interactions, intelligence gathering, threat modelling, vulnerability analysis, exploitation, post exploitation,* dan *reporting*. Berdasarkan hasil pengujian yang dilakukan

pada *website*, ditemukan celah keamanan, di antaranya adalah *Web Server Transmits Cleartext Credentials, Cross-Site Scripting* (XSS), *Cross-Site Request Forgery* (CSRF). Pada akhir penelitian dibuat rekomendasi atau usulan perbaikan untuk memperbaiki celah keamanan yang telah ditemukan. Dari hasil penelitian, dapat disimpulkan bahwa pengujian keamanan *website* milik SMKN Cibatu dengan menggunakan metode PTES mampu membantu sekolah meningkatkan keamanan sistem informasi di dalam menghadapi ancaman peretasan *website* yang berasal dari lingkungan internal maupun eksternal [2].

Fauzan dan Syukhri, mengatakan dalam penelitiannya, Hasil pengujian menggunakan *zenmap* didapatkan 6 *port* yang terbuka pada *elearning.unp2.ac.id.* Pada tahapan *Scanning* menggunakan *Acunetix* yang didapatkan ada beberapa kerentanan teratas *yaitu Cross-site Request Forgery, Development configuration file, Slow HTTP Denial of Service Attack, Weak Password, TLS 1.0 Enable ,* dan *exploitation* menggunakan teknik SQL *Injection* dengan hasil tidak dapat melihat celah keamanan, dapat disimpulkan pada tahap *scanning* didapatkan hasil celah dari keamanan *web* adalah sebanyak 96 dengan disimpulkan *Acunetix Threat Level 2* yaitu pada level *Medium* yang artinya tidak terlalu berpengaruhnya dengan serangan-serangan pada *website* tersebut, dan pada tahapan *Exploitation* menggunakan *SQLMap* dengan teknik *SQL Injection* dimana pada tahap ini dinyatakan gagal karena *e-learning2.unp* sudah menggunakan keamanan SSL/HTTPS yang menyulitkan para *hacker* masuk ke sistem *database* web tersebut. Analisis dari tahapan *Penetration Testing* dapat menjadi dasar untuk meningkatkan kualitas keamanan *website* sehingga dapat mencegah kerentanan yang akan datang [3].

Dalam penelitian ini, dilakukan sebuah analisa *vulnerability* yang ada pada *web* Baleomolcreative untuk memenuhi kebutuhan akan keamanan sistem informasi yang dijalankan oleh Baleomolcreative. *Web* Baleomolcreative adalah *website* yang menampilkan tentang penjualan jasa edit desain untuk supplier toko online di *website* mereka dan juga jasa edit konten maupun iklan dari produk *customer* untuk di iklankan nantinya, dengan harga yang murah dan berkualitas baik, dengan menggunakan *website* ini untuk kebutuhan penjualan jasa *online*, sehingga dibutuhkan pengujian terhadap *web* menggunakan teknik *Penetration Testing Execution Standard* dan *Vulnerability Assessment* untuk mengetahui apakah terdapat celah atau kerentanan terhadap situs *web* Baleomolcreative yang dapat dieksploitasi oleh pihak luar.

2. Metode Penelitian



Gambar 1. Tahapan Penelitian

Berikut penjelasan tahapan penelitian pada gambar 1:

1) *Analysis Planning*
   Pada tahapan ini, dilakukan persiapan *Software* atau *Tools* untuk melakukan penelitian dan membaca literatur-literatur yang ada untuk menunjang analisa celah pada *web* Baleomolcretive.
2) *Diagnose & Identify*
   Pada tahapan ini, dilakukan pengumpulan informasi-informasi yang berkaitan dengan target pengujian yaitu *web* Baleomolcreative. Informasi ini dapat digunakan untuk menyusun strategi serangan untuk mengeksploitasi kerawanan target.
3) *Vulnerability Scanning & Exploitation*
   Pada tahapan ini dilakukan proses pengujian serangan untuk mengetahui celah-celah atau kerawanan pada *web* Baleomolcreative dengan bantuan *Tools* seperti *Nikto, Nmap, Owasp Zap* dan *Burp Suite.*
4) *Preparation of results and Discussion*
   Pada tahap ini dilakukan penyusunan dan menyimpulkan hasil apakah *web* Baleomolcreative terdapat celah untuk diserang, apakah metode yang digunakan mampu mengetahui celah pada *web* Baleomolcretaive, dan seberapa banyak celah yang dapat diketahui pada *web* Baleomolcreative. Lalu membuat kesimpulan dan saran bagi *web* Baleomolcreative mengenai bagian mana saja yang harus dibenahi agar tidak ada celah yang dapat digunakan oleh penyerang untuk mengganti informasi yang ada di Baleomolcreative.

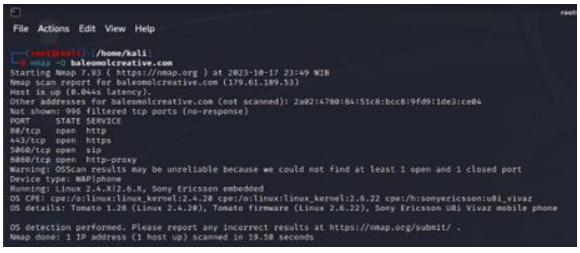
3. Hasil dan Pembahasan

Pada penelitian ini, dilakukan sebuah analisa Vulnerability yang ada pada *web* Baleomolcreative untuk memenuhi kebutuhan akan keamanan sistem informasi yang dijalankan oleh Baleomolcreative. Pengujian pada *web* Baleomolcreative menggunakan teknik *Penetration Testing Execution Standard* dan *Vulnerability Assessment* untuk mengetahui apakah terdapat celah atau kerentanan terhadap situs *web* Baleomolcreative yang dapat dieksploitasi oleh pihak luar.

Information Gathering

Gambar 2. Pengecekan informasi DNS melalui *Nslookup*

Gambar diatas merupakan sistem *Nslookup* yang digunakan untuk mengecek informasi DNS (*Domain Name System*) sebuah domain ataupun alamat IP di internet. Tujuan penggunaan *Nslookup* ini dapat membantu mengidentifikasi potensi masalah jaringan terkait dengan resolusi DNS. *Nslookup* dapat mencari informasi DNS seperti alamat IP dari suatu domain atau nama domain dari suatu alamat IP. IP Hasil dari *scanning* oleh *Nslookup* menampilkan informasi yaitu *Hostname* dan IP *address* yang bukan IP asli dari *host* tersebut karena *web* Baleomolcreative memiliki keamanan *Honeypot low Interaction, Honeypot Low Interaction* merupakan jenis *honeypot* yang memberikan interaksi rendah kepada peretas. Karena sistem yang ada di dalamnya tidak benar benar mirip dengan sistem asli.



Gambar 3. *Scanning Nmap*

Gambar diatas merupakan hasil *scanning Nmap port* dengan menggunakan kode *nmap -O*. Kode tersebut akan dapat mengidentifikasi sistem operasi yang berjalan pada *host* target. Dalam hal ini dapat membantu memahami seperti apa konfigurasi pada sistem target. Berikut beberapa port terbuka yang memungkinkan adanya celah dari *web* Baleomolcreative:

Tabel 1. *Port* yang menghasilkan celah pada *web*

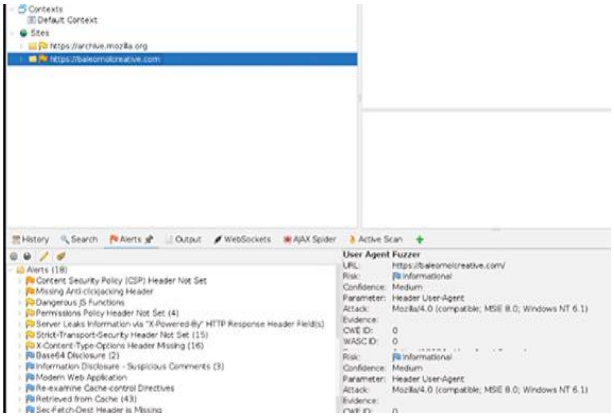
Port	*State Service*	Keterangan
80/tcp	*open HTTP*	Port ini digunakan untuk *Hypertext Transfer Protocol* (HTTP). *Port* ini dibuka agar *website* dapat diakses.
443/tcp	*open HTTPS*	*Port* ini digunakan untuk *Hypertext Transfer Protocol* dengan tambahan SSL (HTTPS). *Port* ini dibuka agar *website* dapat diakses.
5060/tcp	*open SIP*	*Port SIP* merujuk kepada nomor *port* yang digunakan oleh protokol komunikasi *SIP* (*Session Initiation Protocol*) untuk mengirim dan menerima data melalui jaringan IP (*Internet Protocol*). *SIP* adalah protokol komunikasi yang digunakan untuk menginisiasi, memodifikasi, dan mengakhiri sesi komunikasi multimedia, seperti panggilan suara dan video, serta konferensi *web* melalui jaringan IP
8080/tcp	*open HTTP-proxy*	*Port http proxy* digunakan oleh *server proxy* HTTP untuk menerima permintaan dari klien (biasanya perangkat pengguna atau aplikasi) dan meneruskannya ke server tujuan

Gambar 4. *Scanning* menggunakan *Nikto*

Dari gambar diatas dijelaskan bahwa *scanning* menggunakan *Nikto* versi 2.5.0, dengan kode *nikto -h <url>* untuk mengetahui target IP yang terkena *honeypot low interaction* menjadikan "*multiple IPs found*", *hostname*, target *port* serta *server* apa yang digunakan dalam *web* tersebut dan juga ada beberapa *vulnerable* dalam *web* tersebut seperti tidak adanya *CGI Directory* di *web* tersebut, (*Common Gateway Interface*) atau disingkat CGI adalah suatu standar untuk menghubungkan berbagai program aplikasi ke halaman *web*. CGI mirip sebuah program komputer yang menjadi perantara antara standar HTML yang menjadikan tampilan *web* dengan program lain, seperti basis data (*database*), TLS tidak didefinisikan pada website ini, TLS (*Transport Layer Security*), yang merupakan protokol keamanan yang dirancang untuk melindungi privasi dan integritas data yang ditransmisikan melalui jaringan internet. TLS adalah versi yang lebih aman dari protokol sebelumnya, yaitu SSL (*Secure Sockets Layer*).

TLS bekerja dengan mengenkripsi data yang dikirimkan antara pengguna dan *server*, sehingga membuatnya sulit diakses atau dimanipulasi oleh pihak yang tidak sah. Terdapat celah kerentanan pada *The Content-Encoding header is set to "deflate" which may mean that the server is vulnerable to the BREACH attack (The Content-Encoding Header* di setting ke "*deflate*" yang berarti server rentan terhadap serangan *BREACH attack*), *BREACH attack* merupakan singkatan dari "*Browser Reconnaissance and Exfiltration via Adaptive Compression of Hypertext*" yang merupakan jenis serangan keamanan *cyber* yang memanfaatkan kerentanan pada proses kompresi data di dalam protokol HTTPS.

Vulnerability Scanning



Gambar 5. *Scanning Vulnerability* menggunakan *Owasp Zap*

Pemindaian atau proses scanning menggunakan *tools Owasp Zap* dengan metode *standard attack* menghasilkan beberapa celah exploitasi berdasarkan *alert* pada *Owasp Zap* yang berwarna Oranye = *Medium*, Kuning = *Low*, Biru = *Informational*, dari *alert* tersebut menampilkan 3 kerentanan yaitu : 2 *Medium* dengan total *risk* = 2, 5 *low* dengan total *risk* = 37, 11 informational dengan total *risk* = 97, dengan total *alert* 18 dan menghasilkan *alert* diantaranya *Content Security Policy* (CSP) *Header Not Se*t, *Missing Anti-clickjacking Header, Dangerous JS Functions, Permissions Policy Header Not Set, Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s), Strict-Transport-Security Header Not Set, X-Content-Type-Options Header Missing, Base64 Disclosure, Information Disclosure - Suspicious Comments, Modern Web Application, Re-examine Cache-control Directives, Retrieved from Cache,Sec-Fetch-Dest Header is Missing, Sec-Fetch-Mode Header is Missing, Sec-Fetch-Site Header is Missing, Sec-Fetch-User Header is Missing, Storable and Cacheable Content*, dan *User Agent Fuzzer*. Hasil dari *scanning* menemukan beberapa celah yang memiliki tingkatan *Risk Assessment* yang berbeda – beda. Hasil tersebut dapat dilihat pada Tabel 2 dan Tabel 3 sebagai berikut:

Tabel 2. Hasil analisis *Vulnerability Assessment*

Vulnerability Scanning	Peringatan (*Alert*)	*Risk Assessment*	Keterangan
	Content Security Policy (CSP) Header Not Set	*Medium*	*Content Security Policy* (CSP) adalah lapisan keamanan tambahan yang membantu mendeteksi dan mengatasi beberapa jenis serangan, termasuk *Cross Site Scripting* (XSS) dan serangan penyisipan data. Serangan-serangan ini digunakan untuk segala hal mulai dari pencurian data hingga perusakan situs *web* atau distribusi *malware*. CSP menyediakan serangkaian *header* HTTP standar yang memungkinkan pemilik situs *web* mendeklarasikan sumber konten yang diizinkan yang seharusnya dapat dimuat oleh browser pada halaman tersebut tipe konten yang dicakup meliputi *JavaScript*, CSS, HTML *frames*, *font*, gambar, dan objek yang dapat disematkan seperti Java applets, ActiveX, file audio dan video.
	Missing Anti-clickjacking Header	*Medium*	Respon tidak mencakup baik *header Content-Security-Policy* dengan direktif *'frame-ancestors'* maupun *X-Frame-Options* untuk melindungi dari serangan *'ClickJacking'*.
	Dangerous JS Functions	*Low*	Fungsi *JavaScript* berbahaya tampaknya sedang digunakan yang dapat membuat situs rentan.
	Permissions Policy Header Not Set	*Low*	Header Kebijakan Izin (*Permissions Policy*) adalah lapisan keamanan tambahan yang membantu membatasi akses atau penggunaan yang tidak sah terhadap fitur-fitur browser/klien oleh sumber daya *web*. Kebijakan ini memastikan privasi pengguna dengan membatasi atau menentukan fitur-fitur browser yang dapat digunakan oleh sumber daya *web*. *Header* Kebijakan Izin menyediakan serangkaian *header* HTTP standar yang memungkinkan pemilik situs *web* membatasi fitur-fitur browser yang dapat digunakan oleh halaman, seperti kamera, mikrofon, lokasi, layar penuh, dll.
	Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	*Low*	Server *web*/aplikasi sedang bocor informasi melalui satu atau lebih *header* respon HTTP "*X-Powered-By*". Akses terhadap informasi semacam itu dapat memudahkan penyerang mengidentifikasi kerangka kerja/komponen lain yang dibutuhkan oleh aplikasi *web* Anda dan kerentanannya terhadap komponen-komponen tersebut.
	Strict-Transport-Security Header Not Set	*Low*	HTTP *Strict Transport Security* (HSTS) adalah mekanisme kebijakan keamanan *web* di mana *server web* menyatakan bahwa agen pengguna yang patuh (seperti browser *web*) harus berinteraksi dengan *server* tersebut hanya melalui koneksi HTTPS yang aman (artinya, HTTP yang dilapis dengan TLS/SSL). HSTS adalah protokol standar IETF dan dijelaskan dalam RFC 6797.

	X-Content-Type-Options Header Missing	*Low*	*Header Anti-MIME-Sniffing X-Content-Type-Options* tidak diatur ke *'nosniff'*. Ini memungkinkan versi lama dari *Internet Explorer* dan *Chrome* untuk melakukan MIME-*sniffing* pada tubuh respons, yang berpotensi menyebabkan tubuh respons diinterpretasikan dan ditampilkan sebagai tipe konten selain tipe konten yang dinyatakan. Versi *Firefox* yang saat ini (awal 2014) dan versi *legacy*-nya akan menggunakan tipe konten yang dinyatakan (jika ada yang diatur), daripada melakukan MIME-*sniffing*.
OWASP ZAP	*Base64 Disclosure*	*Informational*	Data yang dienkripsi dalam format *Base64* telah terbuka oleh aplikasi/server *web*. Catatan: demi kinerja, tidak semua string *base64* dalam respons dianalisis secara individual, respons secara keseluruhan sebaiknya diperiksa oleh analis/tim keamanan/pengembang.
	Information Disclosure - Suspicious Comments	*Informational*	Respons tampaknya berisi komentar yang mencurigakan yang mungkin membantu penyerang. Catatan: Kesesuaian yang dibuat dalam blok skrip atau *file* melibatkan seluruh konten, bukan hanya komentar.
	Modern Web Application	*Informational*	Aplikasi ini tampaknya adalah aplikasi *web* modern. Jika Anda perlu menjelajahinya secara otomatis, maka *Spider Ajax* mungkin lebih efektif daripada yang standar.
	Re-examine Cache-control Directives	*Informational*	*Header cache-control* tidak diatur dengan benar atau tidak ada, memungkinkan *browser* dan *proxy* menyimpan konten dalam *cache*. Untuk aset statis seperti file css, js, atau gambar, ini mungkin dimaksudkan. Namun, sumber daya tersebut sebaiknya diperiksa untuk memastikan tidak ada konten sensitif yang akan disimpan dalam *cache*.
	Retrieved from Cache	*Informational*	Konten ini diambil dari *cache* bersama. Jika data respons bersifat sensitif, pribadi, atau spesifik untuk pengguna, ini dapat mengakibatkan bocornya informasi sensitif. Dalam beberapa kasus, hal ini bahkan dapat mengakibatkan seorang pengguna mendapatkan kendali penuh atas sesi pengguna lain, tergantung pada konfigurasi komponen-komponen *caching* yang digunakan dalam lingkungan mereka. Ini terutama merupakan masalah di mana server *caching* seperti *cache* "*proxy*" diatur dalam jaringan lokal. Konfigurasi ini biasanya ditemukan dalam lingkungan perusahaan atau pendidikan, misalnya.
	Sec-Fetch-Dest Header is Missing	*Informational*	Menentukan bagaimana dan dimana data akan digunakan. Misalnya, jika nilai tersebut adalah audio, maka sumber daya yang diminta harus berupa data audio dan bukan jenis sumber daya lainnya.

	Sec-Fetch-Mode Header is Missing	*Informational*	Memungkinkan untuk membedakan antara permintaan untuk menavigasi antara halaman HTML dan permintaan untuk memuat sumber daya seperti gambar, audio, dll.
	Sec-Fetch-Site Header is Missing	*Informational*	Menentukan hubungan antara asal (*origin*) inisiator permintaan dan asal (*origin*) target.
	Sec-Fetch-User Header is Missing	*Informational*	Menyatakan apakah permintaan navigasi diinisiasi oleh pengguna.
	Storable and Cacheable Content	*Informational*	Konten respons dapat disimpan oleh komponen-komponen *caching* seperti server *proxy*, dan dapat diambil langsung dari *cache*, bukan dari *server* asal oleh *server-server caching*, sebagai tanggapan terhadap permintaan serupa dari pengguna lain. Jika data respons bersifat sensitif, pribadi, atau spesifik pengguna, ini dapat mengakibatkan bocornya informasi sensitif. Dalam beberapa kasus, ini bahkan dapat menyebabkan pengguna mendapatkan kendali penuh atas sesi pengguna lain, tergantung pada konfigurasi komponen-komponen caching yang digunakan dalam lingkungan mereka. Ini terutama merupakan masalah di mana *server caching* "bersama" seperti cache "*proxy*" diatur dalam jaringan lokal. Konfigurasi ini biasanya ditemukan dalam lingkungan perusahaan atau pendidikan, misalnya.
	User Agent Fuzzer	*Informational*	Periksa perbedaan respons berdasarkan *User Agent* yang *difuzz* (misalnya, situs *mobile*, akses sebagai Mesin Pencari *Crawler*). Bandingkan status kode respons dan kode *hash* dari tubuh respons dengan respons asli.

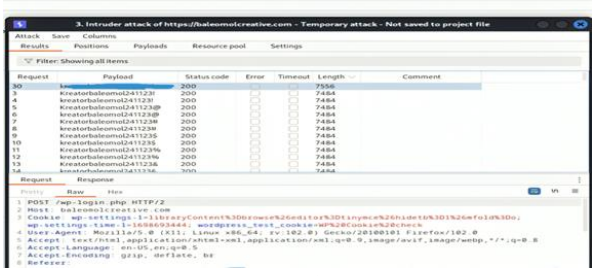
Tabel 3. Solusi perbaikan *website* Balomolcreative

Vulnerability Scanning	*Peringatan (Alert)*	*Risk Assessment*	Solusi
	Content Security Policy (CSP) Header Not Set	*Medium*	Pastikan bahwa *server web*, *server* aplikasi, penyeimbang beban, dan sebagainya diatur untuk mengatur *header Content-Security-Policy*.
	Missing Anti-clickjacking Header	*Medium*	Browser *web* modern mendukung *header* HTTP *Content-Security-Policy* dan *X-Frame-Options*. Pastikan salah satu dari keduanya diatur pada semua halaman *web* yang dihasilkan oleh situs/aplikasi. Jika mengharapkan halaman tersebut hanya dipasang oleh halaman-halaman di *web server* (misalnya, itu bagian dari *FRAMESET*), maka harus menggunakan *SAMEORIGIN*. Namun, jika tidak mengharapkan halaman tersebut akan dipasang oleh halaman lain sama sekali, maka sebaiknya menggunakan *DENY*. Sebagai alternatif, pertimbangkan untuk mengimplementasikan direktif "*frame-ancestors*" dari *Content Security Policy*.
	Dangerous JS Functions	*Low*	Lihat referensi untuk saran keamanan mengenai penggunaan fungsi-fungsi ini.

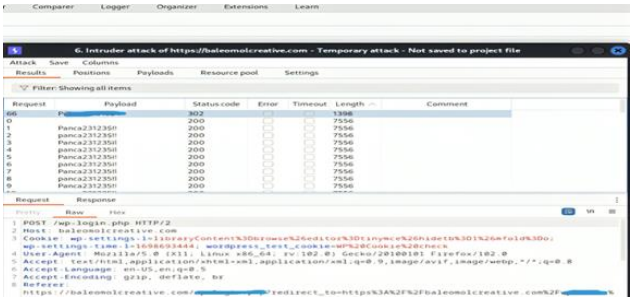
	Permissions Policy Header Not Set	*Low*	Pastikan bahwa server *web*, server aplikasi, penyeimbang beban, dan sebagainya diatur untuk mengatur *header Permissions-Policy*.
	Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	*Low*	Pastikan bahwa server *web*, server aplikasi, penyeimbang beban, dan sebagainya diatur untuk menekan *header* "*X-Powered-By*"
	Strict-Transport-Security Header Not Set	*Low*	Pastikan bahwa *server web, server* aplikasi, penyeimbang beban, dan sebagainya diatur untuk mengaktifkan *Strict-Transport-Security*.
	X-Content-Type-Options Header Missing	*Low*	*Header Anti-MIME-Sniffing X-Content-Type-Options* tidak diatur ke *'nosniff'*. Ini memungkinkan versi lama dari *Internet Explorer* dan *Chrome* untuk melakukan *MIME-sniffing* pada tubuh respons, yang berpotensi menyebabkan tubuh respons diinterpretasikan dan ditampilkan sebagai tipe konten selain tipe konten yang dinyatakan. Versi *Firefox* yang saat ini (awal 2014) dan versi *legacy*-nya akan menggunakan tipe konten yang dinyatakan (jika ada yang diatur), daripada melakukan *MIME-sniffing*.
OWASP ZAP	*Base64 Disclosure*	*Informational*	Konfirmasi secara manual bahwa data *Base64* tidak bocor informasi sensitif, dan bahwa data tersebut tidak dapat dikumpulkan/digunakan untuk mengeksploitasi kerentanannya lainnya.
	Information Disclosure - Suspicious Comments	*Informational*	Hapus semua komentar yang memberikan informasi yang dapat membantu penyerang dan perbaiki masalah yang mendasar yang mereka rujuk.
	Modern Web Application	*Informational*	Ini adalah peringatan informatif sehingga tidak ada perubahan yang diperlukan.
	Re-examine Cache-control Directives	*Informational*	Untuk konten yang aman, pastikan *header HTTP cache-control* diatur dengan "*no-cache, no-store, must-revalidate*". Jika suatu aset seharusnya disimpan dalam *cache*, pertimbangkan untuk mengatur direktif "*public, max-age, immutable*".
	Retrieved from Cache	*Informational*	Pastikan bahwa respons tidak mengandung informasi sensitif, pribadi, atau spesifik pengguna. Jika iya, pertimbangkan penggunaan *header response* HTTP berikut, untuk membatasi atau mencegah konten disimpan dan diambil dari *cache* oleh pengguna lain: *Cache-Control: no-cache, no-store, must-revalidate, private* *Pragma: no-cache* *Expires: 0* Konfigurasi ini mengarahkan *server caching* yang mematuhi HTTP 1.0 dan HTTP 1.1 untuk tidak menyimpan *respons*, dan untuk tidak mengambil *respons* (tanpa validasi) dari *cache*, sebagai tanggapan terhadap permintaan serupa.
	Sec-Fetch-Dest Header is Missing	*Informational*	Pastikan bahwa *header Sec-Fetch-Dest* disertakan dalam *header* permintaan.
	Sec-Fetch-Mode	*Informational*	Pastikan bahwa *header Sec-Fetch-Mode* disertakan dalam

Header is Missing		*header* permintaan.
Sec-Fetch-Site Header is Missing	*Informational*	Pastikan bahwa *header Sec-Fetch-Site* disertakan dalam *header* permintaan.
Sec-Fetch-User Header is Missing	*Informational*	Pastikan bahwa *header Sec-Fetch-User* disertakan dalam permintaan yang diinisiasi oleh pengguna.
Storable and Cacheable Content	*Informational*	Pastikan respons tidak mengandung informasi sensitif, pribadi, atau spesifik pengguna. Jika mengandung, pertimbangkan penggunaan *header respons* HTTP berikut, untuk membatasi atau mencegah konten disimpan dan diambil dari *cache* oleh pengguna lain: *Cache-Control: no-cache, no-store, must-revalidate, private* *Pragma: no-cache* *Expires: 0* Konfigurasi ini mengarahkan *server caching* yang patuh HTTP 1.0 dan HTTP 1.1 untuk tidak menyimpan respons, dan untuk tidak mengambil respons (tanpa validasi) dari *cache*, sebagai tanggapan terhadap permintaan serupa.
User Agent Fuzzer	*Informational*	-

Exploitation

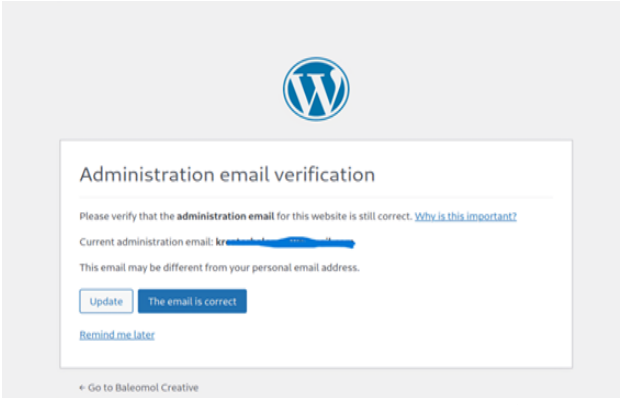


Gambar 6. Eksploitasi *Authentication Bypass Username with Burp Suite*



Gambar 7. Eksploitasi *Authentication Bypass Password with Burp Suite*

Dari kedua gambar diatas dijelaskan bahwa dengan adanya celah pada HTTP *Response Header Field* di *website* Baleomolcreative, menggunakan aplikasi *Burp Suite* ini dapat menggunakan *bruteforce* di bagian *username* dan *password* pada *website login wordpress* Baleomolcreative, dilihat pada gambar pertama untuk mencari *username* yang valid dapat melihat perbedaan di *Request* dan *Length* pada tab *Intruder*, pada gambar kedua mencari *password* yang valid dengan melihat perbedaan di *Request*, *Status Code* dan *Length* pada tab *Intruder*.



Gambar 8. Hasil eksploitasi *Bruteforce Authentication Bypass Username & Password with Burp Suite*

Dengan berhasilnya *Bruteforce*, hasil kombinasi *username* dan *password* yang dapat digunakan untuk *login* sebagai admin. Namun, untuk masuk ke database *WordPress* Baleomolcreative, sekarang perlu melakukan verifikasi *email*. Dari hasil exploitasi ini, dapat disimpulkan bahwa masih ada kelemahan pada bagian "*Server Leaks Information via "X-Powered-By*" HTTP *Response Header Field*" pada *web* Baleomolcreative.

4. Kesimpulan

Penelitian di *website* Baleomolcreative menggunakan langkah *Diagnose and Identify* dengan *tools* seperti *Nslookup*, *Nmap*, dan *Nikto*. Setelah itu, dilakukan *Vulnerability Scanning* dan *Exploitation* dengan *tools Owasp Zap* dan *Burp Suite*. Hasilnya menunjukkan adanya risiko kerentanan yang terkonfirmasi melalui *Vulnerability Scanning*. Meskipun hasil *scanning* dari beberapa *tools* berbeda, namun hasilnya bisa memberikan referensi untuk rekomendasi keamanan. Berbagai jenis kerentanan teridentifikasi pada berbagai level di *website* ini, dengan parameter utama dari *Owasp Zap*. *Tool* ini dapat memberikan petunjuk dan penjelasan yang kompleks untuk membantu penelitian dalam menentukan langkah preventif lebih lanjut dan pengambilan keputusan keamanan pada *web* Baleomolcreative. Kelebihan dari *Owasp Zap* adalah kemampuannya untuk melihat *source code* yang ditandai oleh *tool* tersebut. Dalam pengujian menggunakan Metodologi *Penetration Testing Execution Standard & Vulnerability Assessment*, *Owasp Zap* berhasil menguji dan memindai kerentanan di *web* Baleomolcreative. Pengujian ini berhasil mengidentifikasi 3 tingkat kerentanan di *web*

Baleomolcreative, yaitu *medium*, *low*, dan *informational*, dengan total 18 *alert* yang dihasilkan dari notifikasi pada *Owasp Zap*. Proses *scanning* mencakup pengujian kerentanan seperti *Content Security Policy, Anti-clickjacking Header, Dangerous JS Functions, Permissions Policy,* dan lainnya. Agar pengujian ini lebih optimal, menggunakan alat eksploitasi bernama *Burp Suite* dengan teknik *Bruteforce* pada bagian dimana "*Server Leaks Information via "X-Powered-By" HTTP Response Header Field*". Proses *scanning* yang telah dilakukan sebelumnya dengan alat *Owasp Zap* berdasarkan *Risk Assessment* dengan indikasi "*Low*". Dalam uji coba ini, berhasil ditemukan kombinasi *username* dan *password* yang dapat digunakan untuk *login* sebagai admin di *WordPress*. Meskipun *web* Baleomolcreative memiliki keamanan *Honeypot Low Interaction* seperti perubahan alamat IP *Host*, akan tetapi masih terdapat kerentanan. Disarankan agar web ini untuk menerapkan *Intrusion Prevention System* pada *Honeypot* yang digunakan agar selain mendeteksi serangan yang masuk, dapat juga dilakukan pencegahan atau melawan serangan yang diarahkan kedalam *web*.

5. Daftar Pustaka

[1] Utoro, S., Nugroho, B. A., Meinawati, M., & Widianto, S. R. (2020). Analisis Keamanan Website E-Learning SMKN 1 Cibatu Menggunakan Metode Penetration Testing Execution Standard. *MULTINETICS*, *6*(2), 169-178.

[2] Fauzan, F. Y., & Syukhri, S. (2021). Analisis Metode Web Security PTES (Penetration Testing Execution And Standart) Pada Aplikasi E-Learning Universitas Negeri Padang. *Voteteknika (Vocational Teknik Elektronika dan Informatika)*, *9*(2), 105-111. DOI : https://doi.org/10.24036/voteteknika.v9i2.111778.

[3] Zen, B. P., Gultom, R. A., & Reksoprodjo, A. H. (2020). Analisis Security Assessment Menggunakan Metode Penetration Testing dalam Menjaga Kapabilitas Keamanan Teknologi Informasi Pertahanan Negara. *Teknologi Penginderaan*, *2*(1). DOI: https://doi.org/10.33172/tp.v2i1.574.

[4] Mardianto, I., Sedyono, A., & Hafzan, A. (2015). Analisa Kerentanan Sis. trisakti. ac. id Menggunakan Teknik Vulnerability Scan. *Jetri: Jurnal Ilmiah Teknik Elektro*.

[5] Sunaringtyas, S. U., & Prayoga, D. S. (2021). Implementasi Penetration Testing Execution Standard Untuk Uji Penetrasi Pada Layanan Single Sign-On. *Edu Komputika Journal*, *8*(1), 48-56. DOI: https://doi.org/10.15294/edukomputika.v8i1.47179.

[6] Subandi, K., & Sugara, V. I. (2022). Analisis Serangan Vulnerabilities Terhadap Server Selama Work from Home saat Pandemi Covid-19 sebagai Prosedur Mitigasi. *Jurnal Asiimetrik: Jurnal Ilmiah Rekayasa dan Inovasi*, 125-132. DOI: https://doi.org/10.35814/asimetrik.v4i1.3127.

[7] Darojat, E. Z., Sediyono, E., & Sembiring, I. (2022). Vulnerability Assessment Website E-Government dengan NIST SP 800-115 dan

OWASP Menggunakan Web Vulnerability Scanner. *Jurnal Sistem Informasi Bisnis*, *12*(1), 36-44.

[8] Muhyidin, Y., Totohendarto, M. H., & Undamayanti, E. (2022). Perbandingan Tingkat Keamanan Website Menggunakan Nmap Dan Nikto Dengan Metode Ethical Hacking. *Jurnal Teknologika*, *12*(1), 80-89. DOI: https://doi.org/10.51132/teknologika.v12i1.143.

[9] Herawati, N., & Budiyanto, V. (2023). ANALISIS KEAMANAN SEBUAH DOMAIN MENGGUNAKAN OPEN WEB APPLICATION SECURITY PROJECT (OWASP) Zap. *JURNAL TEKNOLOGI TECHNOSCIENTIA*, 27-36.

[10] Tania, A. M., Setiyadi, D., & Khasanah, F. N. (2018). Keamanan website menggunakan vulnerability assessment. *INFORMATICS FOR EDUCATORS AND PROFESSIONAL: Journal of Informatics*, *2*(2), 171-180.

[11] Fathurahman, M., & Aziz, A. (2022). Vulnerability Assessment Dan Penetration Test Pada Website Ma/Mts Husnul Khatimah Kuningan. In *Prosiding Seminar Nasional Terapan Riset Inovatif (Sentrinov)* (Vol. 8, No. 3, pp. 138-145).

[12] Budiman, A., Ahdan, S., & Aziz, M. (2021). Analisis Celah Keamanan Aplikasi Web E-Learning Universitas Abc Dengan Vulnerability Assesment. *Jurnal Komputasi*, *9*(2).

[13] Aristian, A., & Cholil, W. (2022). Analisis Vulnerability Terhadap Website Lembaga Bahasa LIA Palembang Menggunakan Nessus, Netsparker dan Acunetic. *Jurnal Pendidikan dan Konseling (JPDK)*, *4*(4), 2459-2473. DOI: https://doi.org/10.31004/jpdk.v4i4.5821.

[14] Setiawan, B., Samopa, F., Akbar, I. A., Sani, N. A., Hidayanto, B. C., & Dharmawan, Y. S. (2023). Pendampingan Analisis Vulnerability dan Hardening pada Website Pemerintah Kota Surabaya. *Sewagati*, *7*(6), 897-906. DOI: https://doi.org/10.12962/j26139960.v7i6.624.

[15] Ohyver, J. T. (2022). *Simulasi Keamanan Jaringan pada DPDK OpenvSwitch Berbasis Network-Based Intrusion Detection System (NIDS)* (Doctoral dissertation).