J-INTECH (Journal of Information and Technology)
Terakreditasi Kemendikbud SK No. 204/E/KPT/2022
E-ISSN: 2580-720X || P-ISSN: 2303-1425

Perancangan *Security Information and Event Management* (SIEM) untuk Mendeteksi Insiden pada Situs Web

Stevanus Evo[1], Daniel Rudiaman Sijabat[2*]
[1,2]*Sekolah Tinggi Informatika & Komputer Indonesia, Prodi Teknik Informatika, Malang, Indonesia*

Informasi Artikel

Diterima: 21-05-2023
Direvisi: 06-06-2023
Diterbitkan: 30-06-2023

Kata Kunci
Security; SIEM; Situs Web

Email Korespondensi:
daniel223@stiki.ac.id

Abstrak

Dalam era digital yang semakin kompleks, situs *web* menjadi sasaran utama serangan oleh pihak yang tidak bertanggung jawab. Oleh karena itu, diperlukan sistem yang dapat memantau, menganalisis, dan memberikan peringatan dini terhadap aktivitas mencurigakan atau serangan yang terjadi pada situs *web*. Untuk itu pada penelitian ini dilakukan analisis dan perancangan sebuah sistem *Security Information and Event Management* (SIEM) yang dapat digunakan untuk mendeteksi insiden serangan pada situs *web*. SIEM yang dibangun pada penelitian ini menggunakan *Elastic Stack* yang akan digunakan untuk menganalisa, mengawasi, mendeteksi, dan menyimpan informasi *event-event* atau *log-log* keamanan pada tiap-tiap *agent* yang terkoneksi. Untuk mensimulasikan pengujian jenis-jenis serangan pada situs *web* seperti pada pada OWASP TOP 10 2017, digunakan beberapa aplikasi *web* yang rentan, yaitu DVWA, XVWA, dan MUTILLIDAE. Pengujian pada sistem SIEM yang dibuat menunjukkan sistem mampu mendeteksi jenis-jenis serangan terhadap situs *web* seperti yang disebutkan pada pada OWASP TOP 10 2017.

Abtract

In an increasingly complex digital era, websites are the main target for attacks by irresponsible parties. Therefore, a system is needed that can monitor, analyze, and provide early warning against suspicious activity or attacks that occur on websites. For this reason, this research analyzes and designs a Security Information and Event Management (SIEM) system that can be used to detect incidents of attacks on websites. The SIEM built in this study uses an Elastic Stack which will be used to analyze, monitor, detect, and store event information or security logs for each connected agent. To simulate testing types of attacks on websites such as the OWASP TOP 10 2017, several vulnerable web applications are used, namely DVWA, XVWA, and MUTILLIDAE. Tests on the SIEM system that was created showed that the system was able to detect the types of attacks on websites as mentioned in the OWASP TOP 10 2017.

1. Pendahuluan

Keamanan situs *web* menjadi isu yang semakin penting dengan meningkatnya serangan siber yang ditujukan pada situs *web*. Serangan siber ini mengeksploitasi banyaknya kerentanan yang ada pada situs *web*. Untuk melindungi situs *web* dari serangan, perlu adanya sistem pengelolaan informasi dan *event* keamanan yang efektif. Penelitian ini bertujuan untuk menganalisis dan merancang *Security Information and Event Management* (SIEM) yang dapat mendeteksi insiden serangan pada situs *web*. SIEM yang dibangun pada penelitian ini menggunakan *Elastic Stack* yang akan digunakan untuk menganalisis, mengawasi, mendeteksi, dan menyimpan informasi *event-event* atau *log-log* keamanan pada tiap-tiap *agent* yang terkoneksi. Untuk mensimulasikan pengujian jenis-jenis serangan pada situs *web* seperti pada pada OWASP TOP 10 2017, digunakan beberapa aplikasi *web* yang rentan, yaitu DVWA, XVWA, dan MUTILLIDAE.

Sebelum melakukan penelitian ini telah dilakukan kajian pustaka terhadap penelitian-penelitian terdahulu yang relevan, yaitu penelitian oleh Hadiansyah, Chandra (2017) dengan judul "*Pembangunan Server Security Information Management* untuk Monitoring Keamanan di *Server* Diskominfo Provinsi Jawa Barat". Pada penelitian ini penggunaan SIEM memudahkan *administrator* untuk melakukan monitoring terhadap semua *server* yang ada di Diskominfo Provinsi Jawa Barat. Pada penelitian oleh Syani, Ropi (2018) yang berjudul "Analisis dan Perancangan *Network Security System* Menggunakan Teknik *Host-Based Intrusion Detection System* (HIDS) Berbasis *Cloud Computing*" sistem HIDS digunakan untuk mendeteksi serangan-serangan yang menyerang jaringan komputer di Politeknik TEDC. Pada penelitian oleh Arass (2019) dengan judul "*Smart* SIEM: *From Big Data Logs and Events To Smart Data Alerts*" dikembangkan sebuah prototipe *Smart* SIEM dengan mengintegrasikan *Big Data Platform ELK* dengan *Smart* SIEM sehingga SIEM yang dikembangkan dapat menangani isu-isu keamanan yang terkait dengan *Big Data*. Pada penelitian dengan judul Perancangan *Security Information* and *Event Management* (SIEM) untuk Mendeteksi Insiden pada Situs *Web* ini dikembangkan sebuah SIEM yang dapat mendeteksi, memonitor dan menganalisa serangan terhadap situs *web* yang meliputi serangan-serangan utama menurut OWASP Top 10 2017.

2. Metode Penelitian

Bahan dan Alat Penelitian

Bahan dan alat pada penelitian ini mencakup *Virtual Private Server* (VPS) dengan spesifikasi sebagai berikut: 12GB RAM, 8vCPU, 60 GB Disk, OS - Ubuntu 20.04LTS. Untuk *Software menggunakan* Elastic, Logstash, Kibana, Filebeat, ZEEK, *Mozilla* Firefox (*Browser*). data menggunakan Jenis Kerentanan OWASP TOP 10 2017. Dan *hardware* : MacBook Air (Retina, 13-inch, 2018 , 1,6 GHz Intel Core i5, 8 GB 2133 MHz LPDDR3).

Pengumpulan Data dan Informasi

Pada penelitian ini diperlukan data sebagai bahan analisis. Teknik yang digunakan dalam mengumpulkan data atau informasi yaitu dengan mengunjungi situs web yang menyediakan data-data dan dokumentasi tentang *ZEEK, OWASP* dan *Elastic Stack (Elastic,logstash & Kibana)*. Data-data yang dikumpulkan meliputi jenis-jenis kerentanan pada situs *web* dan 10 jenis serangan pada situs *web* berdasarkan OWASP Top 10.

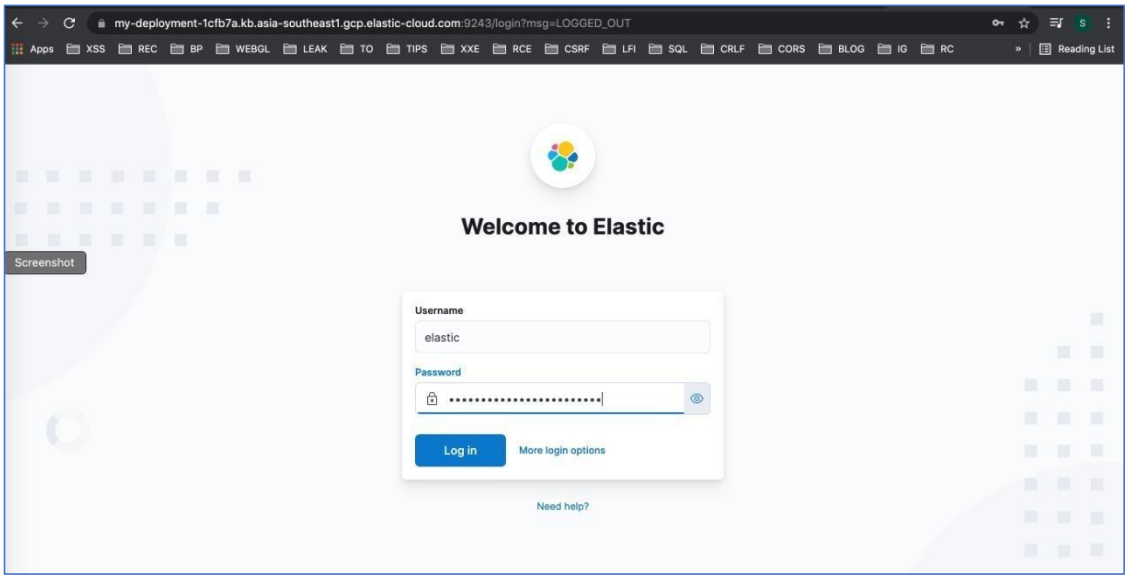
Prosedur Penelitian

Prosedur penelitian ini adalah sebagai berikut: (1) Melakukan pengumpulan data-data dengan mengunjungi beberapa situs yang dibutuhkan dan membahas tentang beberapa *tools* pada penelitian ini seperti situs milik OWASP, ZEEK, dan Elastic. (2) Mengkaji beberapa penelitian serupa atau penelitian yang membahas tentang kerentanan suatu situs web terhadap pola serangan yang dimuat di *Open Web Application Security Project* (OWASP) TOP 10 2017. (3) Merancang *flowchart*, *use case diagram*, *diagram activity*, dan *sequence diagram* dari *Security Information* and *Event Management* (SIEM) sehingga dapat mendeteksi insiden serangan pada situs *web* yang dimuat di *Open Web Application Security Project* (OWASP) TOP 10 2017 . (4) Mengimplementasikan *Security Information and Event Management* (SIEM) menggunakan *software* Elasticsearch, Logstash & Kibana (ELK) pada Elastic Stack Siem Server, memasang *software Filebeat*, dan ZEEK pada *Agent* lalu melakukan simulasi kerentanan pada *web* dan membuat manajemen *rule* pada dashboard Kibana menggunakan KQL.

3. Hasil dan Pembahasan

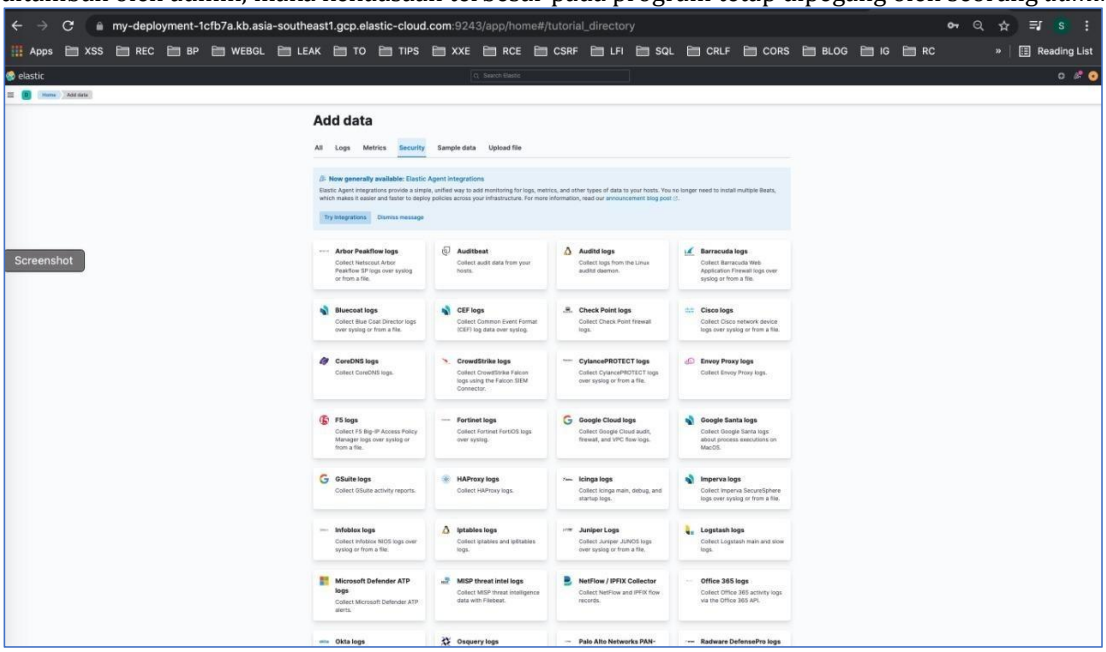
Implementasi SIEM

Untuk menggunakan sistem, administrator terlebih dahulu melakukan *login* melalui jendela *login* seperti pada gambar 1.



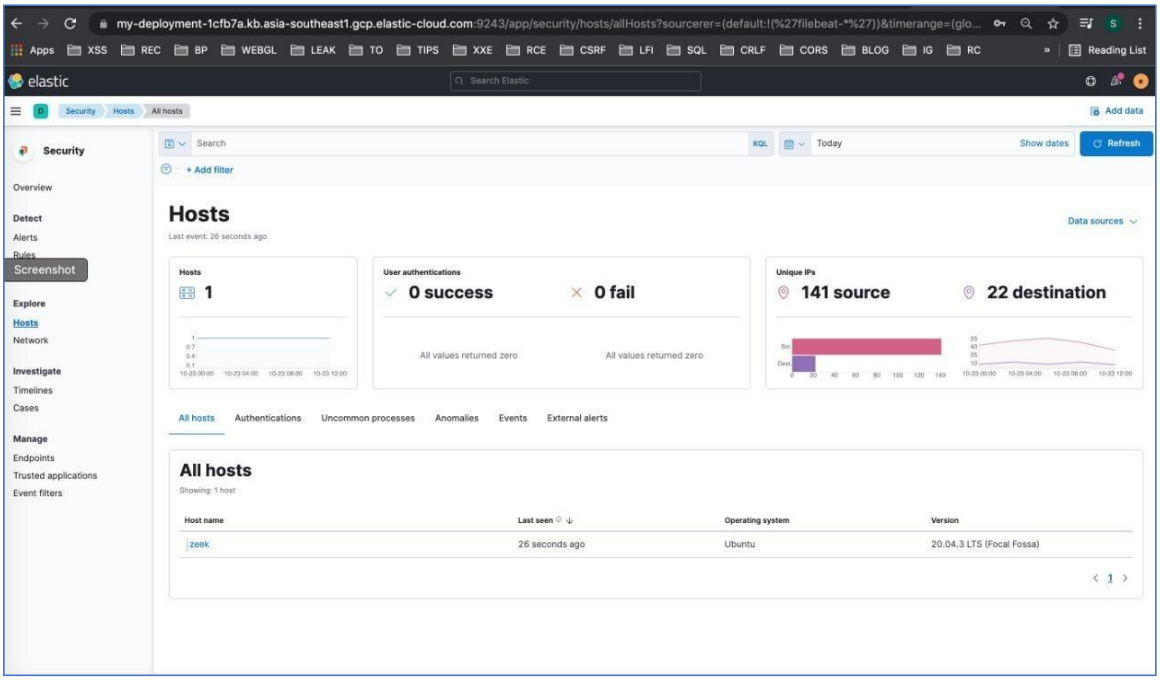
Gambar 1. Jendela Login

Sistem administrator dapat menambahkan agent guna memberikan bantuan untuk menjalankan program melalui jendela Penambahan Agen Zeek seperti pada gambar 2. Tugas agent dibatasi berdasarkan *rule* yang telah diberikan oleh sistem administrator. *Rule* tersebut dimasukkan ke dalam *ruleset* agar *agent* mengetahui langkah mana yang bisa dan tidak bisa dilakukan. Penambahan *agent* juga bertujuan untuk memberikan kemudahan kepada sistem administrator saat menambahkan *rule* dan juga mengurangi *rule*. Karena *agent* dapat ditambah oleh admin, maka kekuasaan terbesar pada program tetap dipegang oleh seorang *admin*.



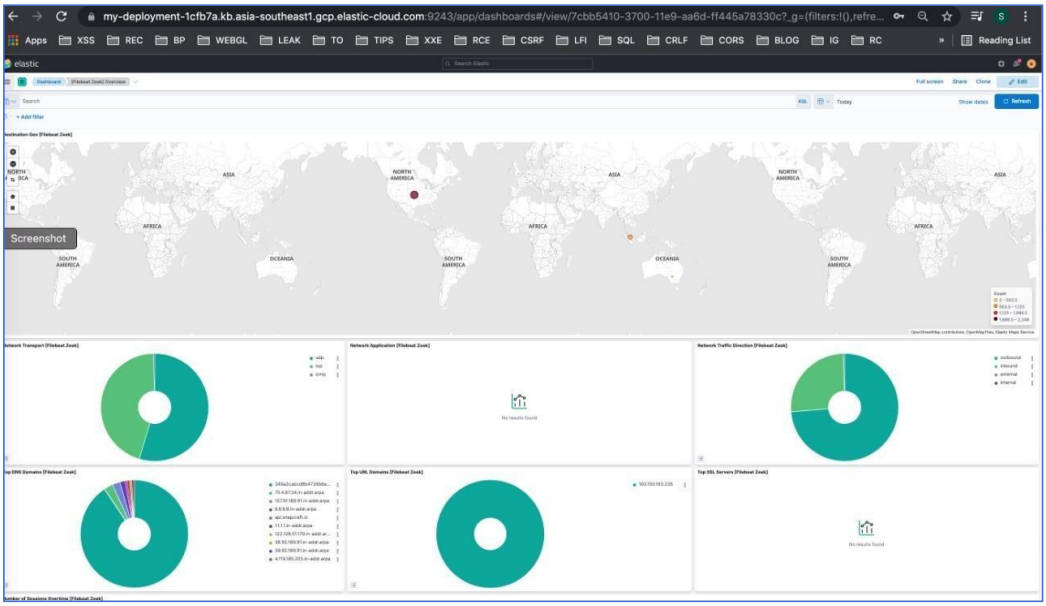
Gambar 2. Penambahan Agen Zeek

Monitoring yang dilakukan oleh *agent* akan ditampilkan dalam bentuk report kepada *System Administrator* seperti pada gambar 3. Pada tahap ini *System Administrator* juga bisa melihat berapa jumlah *agent* yang terhubung.



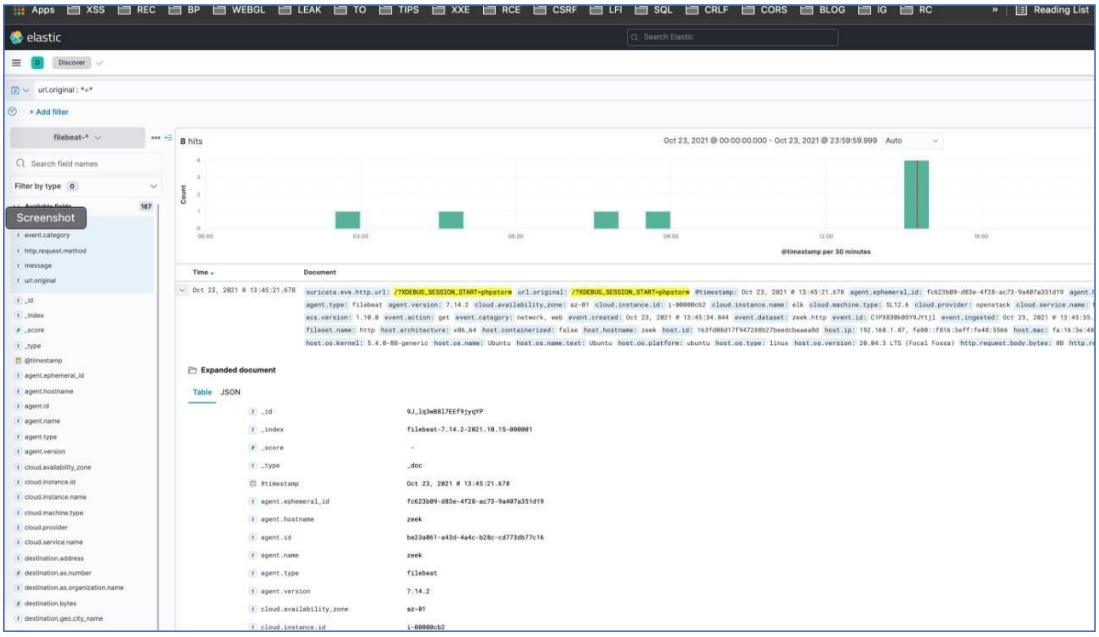
Gambar 3. Tampilan Monitoring Agent

Setelah melakukan monitoring, *agent* akan memberikan *report* berupa hasil riset yang telah dilakukan oleh agent. Sistem administrator dapat memberikan keputusan setelah mendapatkan report dari *agent*. Selain dapat memberitahukan informasi serangan yang diterima, sistem juga dapat mengetahui lokasi dari penyerang menggunakan *log-log* yang diterima kemudian menampilkannya pada sebuah peta geografis seperti ditampilkan pada gambar 4.



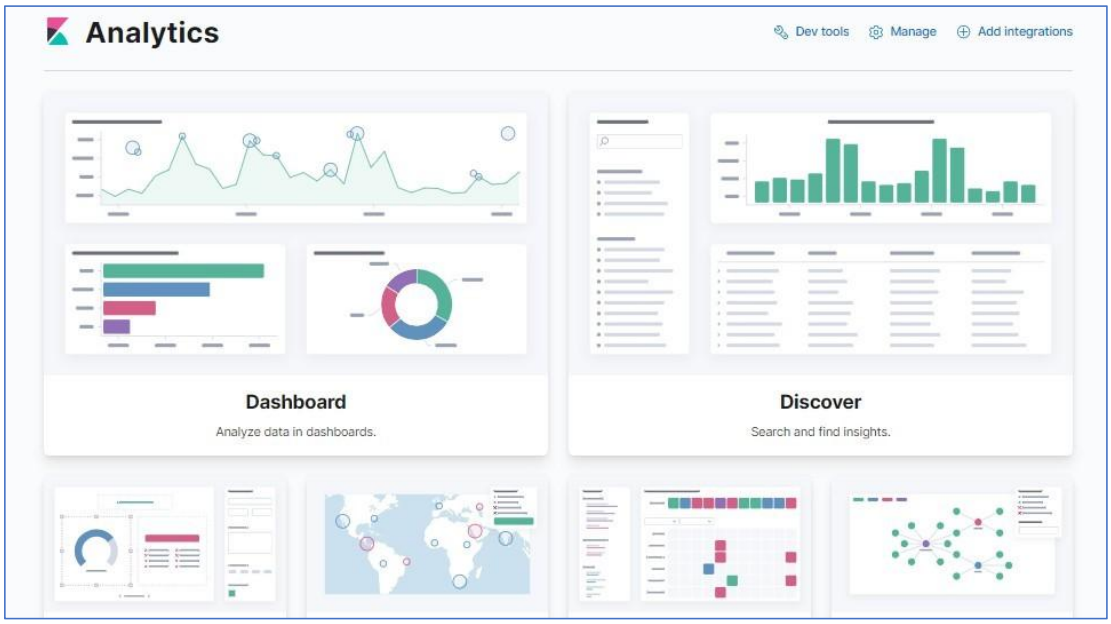
Gambar 4. Tampilan Report dari Agent

Ketika sistem administrator memperoleh hasil *report* dari agent seperti pada gambar 5, maka sistem administrator dapat melakukan analisa guna menentukan langkah selanjutnya yang akan diambil. Pada tahap ini, sistem administrator akan menganalisis alur dari situs *web* untuk mengetahui permasalahan yang sedang terjadi. Apabila permasalahan tidak terindikasi setelah admin melakukan analisa, maka *admin* dapat melanjutkan kepada pemberian aksi.



Gambar 5. Tampilan Analisis Event Log

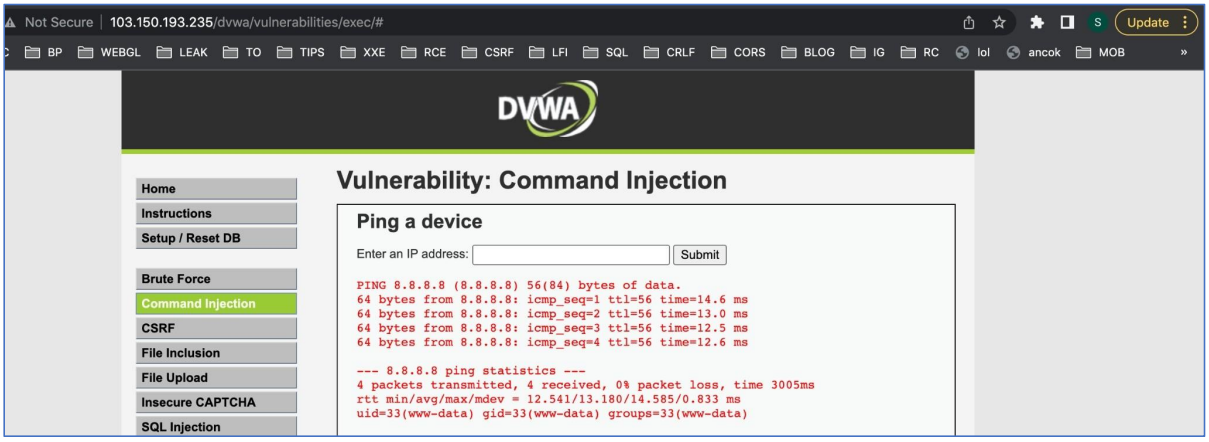
Menu activity yang ditampilkan pada gambar 6 merupakan sub menu yang menampung segala jenis aktivitas, dan menjadikannya satu *log* sehingga apabila pengguna membutuhkan data mengenai aktivitas dalam kurun waktu tertentu, pengguna bisa mengambilnya secara langsung. Selain itu, menu *activity* juga mengatur tentang bagaimana *ruleset*, *agent*, serta *watcher* berjalan pada sisi yang sama untuk mengawasi situs *web* dari kerentanan yang mungkin terjadi. Pengguna juga dapat melihat report dari aktivitas harian melalui *menu activity*. Cara untuk melihat aktivitas pun sangat mudah, pengguna hanya tinggal memilih pada tanggal dan jam berapa aktivitas ingin dilihat. Setelah itu, pengguna tinggal memilih berapa lama waktu aktivitas dimulai. Artinya, pengguna dapat melihat aktivitas dengan sangat detail mulai dari 1 menit, hingga pada waktu yang pengguna inginkan.



Gambar 6. Menu Activity

Penguji Sistem dan Pembahasan

Tujuan dari pengujian sistem adalah untuk memastikan bahwa sistem dapat berjalan seperti yang diinginkan. Pada pengujian ini dilakukan simulasi serangan yang mengeksploitasi jenis kerentanan A1 - *Injection:Remote Code Execution*, di mana attacker memasukan input berupa perintah Linux yang berfungsi untuk mencetak informasi tentang pengguna, inputan yang disisipkan adalah : `8.8.8.8;id` seperti pada gambar 7 . Hasilnya ditunjukkan pada gambar 8, di mana Slack menampilkan notifikasi insiden serangan *remote code injection*.



Gambar 7. Attacker Melakukan Penyerangan Remote Code Injection Pada Server Agent

Gambar 8. Slack Menampilkan Notifikasi Insiden Serangan Remote Code Injection

Selain itu pada penelitian ini juga dilakukan pengujian-pengujian lainnya yang bertujuan untuk mengeksploitasi kerentanan-kerentanan web seperti diberikan pada OWASP TOP 10 2017.
Berikut adalah tabel hasil pengujian sistem dalam mendeteksi 10 insiden serangan yang merujuk pada OWASP TOP 10 2017.

Tabel 1. Uji Coba Deteksi Insiden Serangan OWASP TOP 10 2017

No	*Aksi*	*Malicious Input*	*Reaksi Sistem*	*Hasil Pengujian*
1	Melakukan Serangan A1:2017 - Injection	Remote CommandInjection/DVWA/ (;id)	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil
		SQL Command Injection/DVWA/(1' order by 10–)	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil
2	Melakukan Serangan A2:2017 -Broken Authentication	Bruteforce/DVWA/ (admin:admin) 10x REQUEST	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil
3	Melakukan Serangan A3:2017 -Sensitive Data Exposure	PHPINFO/DVWA/ (/dvwa/phpinfo.php)	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil
4	Melakukan Serangan A4:2017 -XML EXTERNAL ENTITIES (XXE)	XXE/MUTILLIDAE/ (<!ENTITY xxe SYSTEM "file:///dev/random" >]>)	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil
5	Melakukan Serangan A5:2017 -Broken Access Control (BAC)	Local File Inclusion/DVWA/ (../../etc/passwd)	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil
6	Melakukan Serangan A6:2017 -Security Misconfiguration	Default Login Credential/DVWA/ (admin:password)	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil
7	Melakukan Serangan A7;2017 -Cross Site Scripting(XSS)	Reflected XSS/DVWA/ (<script>alert(1)</script>)	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil
8	Melakukan Serangan A8:2017 -Insecure Deserialization	PHP Insecure Deserialization/XVWA/ (/php_object_injection/?r=O:18:"PHPObjectInjection":1:{s:6:"inject";s:13:"system('id');";})	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil
9	Melakukan Serangan A9:2017 -Using Components with Known Vulnerabilities Example	CVE-2017-16894 (LARAVEL)/LARAVEL/ (/laravel/.env)	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil
10	Melakukan Serangan A10:2017 - Insufficient Logging & Monitoring	Insufficient Logging &Monitoring/-/ (Agent DISABLED)	• Menampilkan notifikasi insiden serangan • Menampilkan detail informasi	Berhasil

4. Kesimpulan

Dari penelitian ini dapat disimpulkan bahwa manajemen terhadap insiden serangan pada situs web sangat dibutuhkan untuk mempercepat proses identifikasi serangan dan membantu administrator melakukan tindak lanjut terhadap serangan tersebut. Penggunaan manajemen keamanan khususnya SIEM sangat membutuhkan sumber daya, koneksi jaringan dan biaya yang tidak sedikit. Penerapan SIEM akan sangat berguna khususnya di lingkungan perangkat-perangkat terdistribusi di mana ada banyak *server* yang perlu dimonitor.

5. Referensi

Arass (2019). Smart SIEM: From Big Data Logs and Events To Smart Data Alerts. *International Journal of Innovative Technology and Exploring Engineering (IJITEE), 3186-3191.* https://www.researchgate.net/publication/333752299_Smart_SIEM_From_Big_Data_logs_and_events_to_Smart_Data_alerts

Chandra, Edward (2018). Sensitive Data Exposure. https://mti.binus.ac.id/2018/02/09/sensitive-data-exposure/

Dizdar, Admir (2022). Security Misconfiguration: Impact, Examples, and Prevention. https://brightsec.com/blog/security-misconfiguration/

Febrianto, Andi (2018). Using Component With Known Vulnerabilities. https://mti.binus.ac.id/2018/07/11/using-component-with-known-vulnerabilities/

Hadiansyah, Chandra (2017). Pembangunan server security information management untuk monitoring keamanan di server Diskominfo Provinsi Jawa Barat. https://elib.unikom.ac.id/gdl.php?mod=browse&op=read&id=jbptunikompp-gdl-chandrahad-36808

Kiprin, Borislav (2021). Logging & Monitoring and How to Prevent It. https://crashtest-security.com/insufficient-logging-monitoring-guide/

Muscat, Ian (2019). What Are XML External Entity (XXE) Attacks. https://www.acunetix.com/blog/articles/xml-external-entity-xxe-vulnerabilities/

Nugraha, Ayub (2022). Broken Access Control. https://www.academia.edu/35776231/Broken_Access_Control_OWASP

Situmeang, Yoel (2018). Broken Authentication and Session Management. https://mti.binus.ac.id/2018/07/11/broken-authentication-and-session-management/

Syaikhoni, Ahmad (2018). Cross Site Scripting. https://mti.binus.ac.id/2018/07/11/cross-site-scripting/

Syani, Ropi (2018). Analisis dan Perancangan Network Security System Menggunakan Teknik Host-Based Intrusion Detection System (HIDS) Berbasis Cloud Computing. *Seminar Nasional Telekomunikasi dan Informatika(SELISIK 2018),* 158-163. https://www.researchgate.net/publication/327791386_analisis_dan_implementasi_network_security_system_menggunakan_teknik_host-based_intrusion_detection_system_hids_berbasis_cloud_computing