JKT Jurnal Komputer Terapan
ISSN : 2443-4159 (Print) ISSN: 2460-5255 (Online)
Politeknik Caltex Riau
Empower You to Global Competition

ANALISA FORENSIK CYBER ATTACK TERHADAP WEB SECURITY

Jenri Pujianto Saragih[1], Yuli Fitrisia*[2]
Teknologi Rekayasa Komputer, Politeknik Caltex Riau, Jl. Umban Sari, Pekanbaru, 28265, Indonesia[1,2]
jenri20tk@mahasiswa.pcr.ac.id[1], uli@pcr.ac.id*[2]

**Penulis Koresponden*

ABSTRAK

Teknologi menjadi kebutuhan sehari-hari bagi banyak orang. Penggunaan teknologi yang tidak tepat dapat menimbulkan dampak negatif baik bagi individu maupun organisasi seperti mendapatkan informasi-informasi yang bersifat illegal. Hal ini dapat dilakukan dengan memanfaatkan teknologi komputer sehingga menimbulkan kejahatan siber seperti eksploitasi website dengan serangan cross-site scripting. Untuk mengungkap kasus tersebut dibutuhkan bukti digital untuk membantu dalam mengungkap kasus tindak kejahatan siber melalui digital forensic seperti disk image yang berisikan file system, file temp (log), dan partisi disk lain didalamnya. Adapun tujuan penelitian ini yaitu untuk memperoleh dan menganalisis barang bukti digital tersebut yang dilakukan dengan teknik static forensic dengan menerapkan metode NIST. Teknik static forensic, memperoleh bukti digital dengan melakukan ekstraksi dan analisis setelah terjadi insiden ataupun setelah sistem tidak beroperasi. Penelitian ini menghasilkan artifak-artifak barang bukti digital (evidence) berupa disk image, file access.log yang mencatat seluruh permintaan yang dilakukan pelaku terkait serangan reflected cross-site scripting dan stored cross-site scripting yang tercatat pada log server, dan IP Address pelaku. Sedangkan pada file error.log tidak ada indikasi terjadinya serangan cross-site scripting dan tidak menghasilkan bukti apapun indikasi serangan. Barang bukti tersebut, diolah dan disajikan sebagai laporan hasil investigasi digital forensic untuk memperkuat kasus hukum terhadap pelaku serangan.

Kata kunci: *Cross-site scripting, Digital Forensic, Kejahatan Siber, NIST, Static Forensic.*

ABSTRACT

Technology has become a daily necessity for many people. Improper use of technology can have negative impacts on both individuals and organizations such as obtaining illegal information. This can be done by utilizing computer technology, resulting in cybercrime such as website exploitation with cross-site scripting attacks. To uncover the case, digital evidence is needed to help uncover cybercrime cases through digital forensics such as disk images containing system files, temp files (logs), and other disk partitions. The purpose of this study is to obtain and analyze digital evidence which is carried out using static forensic techniques by applying the NIST method. Static forensic techniques obtain digital evidence by extracting and analyzing it after an incident occurs or after the system is not operating. The results of this study are digital evidence artifacts namely, disk images, access.log files that record all requests made by the perpetrator related to reflected cross-site scripting attacks and stored cross-site scripting recorded on the server log, and the perpetrator's IP Address. While in the error.log file there is no indication of a cross-site scripting attack and does not find any evidence of an attack. The evidence was processed and presented as a digital forensic investigation report to strengthen the legal case against the perpetrators of the attack.

Keywords: *Cross-site scripting, Cybercrime, Digital Evidence, Digital Forensic, Static Forensic.*

Histori Artikel
Diserahkan: 03 Sept 2024 Diterima setelah Revisi: 01 Nov 2024 Diterbitkan: 28 Nov 2024

1. PENDAHULUAN

Peranan teknologi menjadi aspek penting dalam kehidupan banyak orang. Salah satu manfaat terbesar dari hasil pemanfaatan teknologi adalah pencarian informasi dan pertukaran informasi dari berbagai

sumber yang diakses banyak orang. Website adalah halaman informasi yang disediakan jalur internet sehingga bisa diakses dimana saja, selama terkoneksi dengan jaringan internet[1]. Namun, dibalik banyaknya pengguna yang memanfaatkan teknologi dalam pencarian informasi dan pertukaran informasi, kerap sekali adanya penyalahgunaan teknologi yang tidak tepat untuk kepentingan pribadi yang kerap dilakukan kejahatan siber atau *cybercrime* mengacu pada berbagai aktivitas kriminal yang dilakukan melalui jaringan komputer dan internet. Bentuk-bentuk kejahatan ini meliputi penipuan online, pencurian identitas, serangan malware, peretasan, dan eksploitasi data pribadi. Dengan pesatnya perkembangan teknologi informasi dan komunikasi, kejahatan siber menjadi semakin kompleks dan tersebar luas, mempengaruhi individu, organisasi, dan negara[2].

Oleh karena itu, untuk mengungkap sebuah kasus kejahatan diperlukan barang bukti yang valid dan sah. Jika ada bukti yang sah, maka pelaku dianggap melakukan tindak pidana yang sebenarnya, dan pelaku harus bertanggung jawab atas tindak pidana tersebut[3]. Barang bukti kasus *cybercrime* terbagi menjadi dua, yaitu barang bukti digital dan barang bukti elektronik. Salah satu barang bukti digital yang dapat dijadikan sebagai barang bukti dalam kasus kejahatan tersebut adalah *disk image*. Hal ini tertulis pada pasal 5 ayat 1 Undang-Undang Nomor 11 Tahun 2008 tentang Informasi dan Transaksi Elektronik "*informasi elektronik dan/atau dokumen elektronik dan/atau hasil cetaknya merupakan alat bukti hukum yang sah*"[4].

Dalam dunia teknologi informasi, ilmu forensik sering disebut dengan forensik digital. Tujuan dari forensik digital adalah untuk memahami proses investigasi dan menemukan data yang dapat digunakan untuk mendukung bukti digital suatu kejahatan, seperti penipuan, penyalahgunaan data, atau modifikasi data[5]. *Static Forensic* merupakan salah satu cabang ilmu forensik yang melibatkan analisis dan ekstraksi data pada penyimpanan perangkat dengan keadaan perangkat ataupun *system* tidak bekerja. *Static Forensic* difokuskan pada pemeriksaan hasil untuk menganalisis isi dari bukti digital, seperti file yang dihapus, *history* dari proses data transaksi, koneksi jaringan, *history login user* guna membuat ringkasan tentang kegiatan yang dilakukan pada bukti digital sewaktu digunakan[6]. Dalam melakukan proses tersebut harus memenuhi Standarts Operating Procedure (SOP) dari *National Institute of Standards and Technology* (NIST), untuk menjadi barang bukti yang sah di persidangan hukum.

Pada proses ini fokus utama pengakuisian barang bukti digital dilakukan pada *disk image*, *log access server*. Oleh karena itu, tujuan penelitian ini adalah melakukan *static forensic* untuk menganalisis barang bukti digital yang diperoleh dimana kondisi perangkat atau *system* dalam keadaan tidak beroperasi. Adapun ruang lingkup penelitian ini yaitu menelukan bukti digital berupa *disk image*, *file access.log* yang mencatat seluruh permintaan yang dilakukan pelaku terkait serangan *reflected cross-site scripting* dan *stored cross-site scripting* yang tercatat pada *log server*, dan IP Address pelaku. Hal ini sangat penting untuk menjaga integritas data dan pemeriksaan mendalam terhadap seluruh isi *disk*, termasuk *file* yang dihapus, *metadata*, dan struktur *file* yang mungkin tidak terlihat pada sistem yang beroperasi.

2. TINJAUAN PUSTAKA

2.1 PENELITIAN TERDAHULU

Penelitian [7] mengusulkan Analisa Serangan *SQL Injection* pada *Server* pengisian Kartu Rencana Studi (KRS) Online dengan menggunakan pengujian *white box*. Pada penelitian tersebut, dapat mensimulasikan serangan pada sistem pengisian Kartu Rencana Studi (KRS) yang didalamnya terdapat sistem informasi guna mencapai *Digital Forensic Readiness Index* (DiFRI). *Digital forensic Readiness Index* (DiFTERI) merupakan suatu cara untuk mengukur kesiapan suatu institusi/organisasi dalam mencegah dan menangani kejahatan dunia maya yang nantinya dapat diukur dengan melihat berbagai faktor dan indikator yang setelahnya dihitung akan menghasilkan suatu nilai [7]. Adapun kelemahan pada penelitian ini yaitu tidak menjelasan metode forensik digital yang digunakan. Selain itu juga tidak dijelaskan jenis barang bukti apa saja yang ditemukan dan bagaimana proses pengolahan barang buktinya.

Penelitian [6] mengusulkan menggunakan metode *static forensic* menunjukan bahwa telah terjadinya manipulasi atau perubahan data pagu anggaran kegiatan yang melebihi pagu anggaran program (mark-up dalam perencanaan anggaran) yang dilakukan oleh peretas yang masuk melalui *database* Sistem Informasi Manajemen Daerah (SIMDA). Adapun jenis serangan yang dilakukan yaitu *SQL Injection*. Hasil dari proses forensik dapat membuktikan terjadinya manipulasi data yang dilakukan oleh peretas

yang dapat terdeteksi menggunakan *tools* SQL Profiler dan SQL Log Analyzer sehingga hasil foreksik dapat dijadikan barang bukti digital untuk membantu penegak hukum dalam mengungkapkan kasus kejahatan *cybercrime* dan dapat dipertanggungjawabkan pada proses hukum dipengadilan [6]. Pada penelitian ini tidak menjelaskan teknik *imaging* barang bukti yang digunakan untuk menjamin integritas barang bukti tersebut. Selain itu, penelitian ini hanya menjelaskan data apa saja yang telah dirubah dan waktu kejadian, tanpa menemukan IP Address pelaku.

Penelitian [8] mengusulkan Analisis Forensik Serangan *SQL Injection* dan *DoS* menggunakan *Instrution Detection System* pada *Server* berbasis Lokal. Penelitian ini menghasilkan snort mendeteksi hasil percobaan *attacker* menggunakan software sqlmap. Tetapi hasil *scan* tidak menemukan *database* maka serangan terdeteksi oleh snort sebagai serangan *DDoS* dengan tingkat priority 2 (medium). Pada pengujian teknik serangan DoS yang dilakukan dengan berfokus pada port 22 ssh, hasil yang didapat teknik ini mampu menyerang protocol TCP/IP pada eth0 server. Hal ini juga membuat snort harus mendeteksi kembali adanya serangan dari *DoS* dengan hasil *flooding attack*, yang sebelumnya terdeteksi *Attempted Information Leak*, kemudian jalur ICMP pun menjadi *uncreachable* dikarenakan serangan *DoS* [8]. Pada penelitian ini tidak menjelaskan teknik forensik yang digunakan. Selain itu, penelitian ini hanya menjelaskan seperti apa proses serangan yang terjadi dan perubahan apa yang dilakukan oleh *attacker.* Sehingga analisis forensik yang dilakukan tidka dapat dijamin integritasnya.

Penelitian [9] mengusulkan Analisis Forensik Pada Web Phising menggunakan Metode National Institute of Standarts and Technology. Penelitian ini terkait tahapan-tahapan National Institute of Standards and Technology (NIST) untuk menghasilkan barang bukti. berdasarkan implementasi tahapan-tahapan seperti file capture Barangbuktiphising.pcapng, menggunakan hashcalc untuk pemeriksaan nilai hash MD5 pada barang bukti digital analisis. Sedangkan pada barang bukti file capture Barangbuktiphising.pcapng yang didapatkan berupa URL phishing, DNS yang digunakan oleh pelaku, IP Address server, IP Address destination, identitas penyerang dan e-mail yang menghasilkan informasi tindak kejahatan yang dilakukan phisher [9]. Pada penelitian ini tidak menjelaskan teknik yang digunakan untuk tahap collection apakah perangkat yang dijadikan sebagai barangbukti dalam kondisi hidup atau mati. Selain itu juga tidak menjelaskan teknik yang digunakan apakah static forensic atau live forensic.

Penelitian [10] terkait metode *Network Forensic* (*live forensic*) bertujuan untuk melakukan akuisisi, analisis dan *recovery* dengan menggunakan skenario penyerangan *remote exploit* terhadap mesin target yaitu menggunakan sistem operasi Windows XP SP2 dengan memanfaatkan celah dari protokol *Server Message Block* (SMB) pada port 445 pada sistem operasi Windows XP SP2 sehingga memungkinkan *attacker* untuk masuk kedalam sistem operasi. Hasil penelitian berupa informasi serangan *Remote Exploit* dari lalu lintas jaringan komputer serta bukti digital dari komputer target terkait aktivitas dan IP Penyerang [10]. Pada penelitian ini tidak dijelaskan metode forensik yang digunakan. Selaint itu, juga tidak dijelaskan proses yang dilakukan untuk memastikan integritas barang bukti yang diperoleh. Tetapi untuk pengumpulan barang bukti dilakukan dengan teknik *live forensic.* Penelitian ini memberikan informasi yang jelas tentang artifak barang bukti yang ditemukan.

Secara umum beberapa penelitian sebelumnya, serangan yang dilakukan menggunakan Teknik *SQL Injection*. Sedangkan serangan pada web menggunakan web *Phishing* dan *DoS*. Selain itu pada penelitian sebelumnya tidak menjelaskan teknik forensik digital dan proses yang digunakan untuk menjamin integritas barang bukti yang ditemukan. Sedangkan pada penelitian ini menerapkan teknik *static forensic* untuk menemukan bukti digital dan menerapkan metode NIST untuk proses forensik digital. Selain itu, skenario serangan berupa *cross-site scripting* pada suatu halaman web. Selain itu setiap tahapan NIST menggunakan *tools* yang berbeda yang umum digunakan untuk analisis forensik digital.

2.2 NATIONAL INSTITUTE STANDART OF TECHNOLOGY (NIST)

National Institue Standart of Technology merupakan salah satu lembaga resmi yang mengembangkan dan mengeluarkan standarisasi dan panduan mengenai keamanan. *U.S. Departement of Commerce* melalui *National Institute of Standar and Technology* (NIST) memberikan rekomendasi dalam proses penanganan barang bukti elektronik dan/atau digital yang dapat disajikan di pengadilan atau kebutuhan sendiri padainstitusi investigasi[11].

Dalam melakukan forensic digital terdapat 4 tahapan standar operasional prosedur (SOP), yang telah ditetapkan. Langkah kerja forensik ini digunakan untuk menjabarkan tahapan-tahapan forensics yang akan dilakukan dan dapat diketahui alur-alur penelitian secara terstruktur, dan dapat menjadi acuan dalam menyelesaikan masalah-masalah yang ada [12]. Adapun 4 tahapan tersebut adalah *collection* (pengumpulan), *examination* (pemeriksaan), *analysis* (analisa), dan *repoting* (laporan).

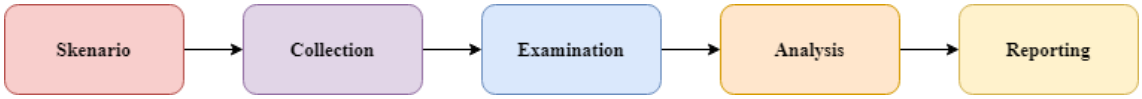
2.3 STATIC FORENSIC

Static Forensic merupakan suatu penerapan metode forensik yang digunakan untuk mengumpulkan bukti digital pada kejahatan siber dengan mengumpulkan bukti dari media penyimpanan. Adapun barang bukti yang ditemukan dalam kondisi mati. Tahapan yang dilakukan biasanya merujuk kepada *Standarts Operating Procedure* (SOP) yang diterapkan dari *National Institute of Standarts and Technology* (NIST) dengan melakukan *collection, examination, analysis* dan *reporting*. Prosedur dan pendekatan konvesional yang digunakan pada metode *forensic static* dimana barang bukti eletronik diproses secara bit-by-bit image dalam melakukan proses forensik[13].

Metode *static forensic* kerap sekali dianggap sebuah metode tradisional, dikarenkan metode ini menjadi salah satu jenis metode forensic digital yang memperoleh hasil ekstraksi barang bukti ketika suatu sistem komputer dimatikan (*post-incident*). Namun, metode ini menawarkan perbandingan yang sangat signifikan dibandingkan dengan *live forensik* mulai dari aspek ekstraksi bukti digital, analisis, penilaian hingga tahap kesesuaian terhadap prosedur hukum [14].

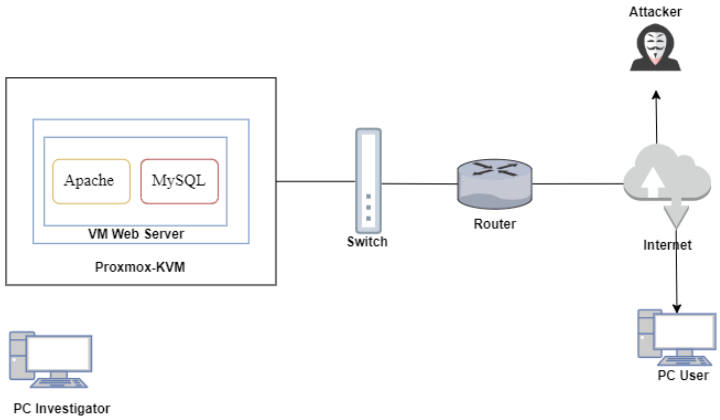
3. METODE

Metode penelitian ini terdiri dari beberapa tahapan berdasarkan metode NIST, mulai dari melakukan skenario serangan, melakukan pengumpulan barang bukti (*collection*), pemeriksaan barang bukti (*examination*), analisis (*analysis*), dan tahap pelaporan (*reporting*). NIST dipilih karena merupakan *cybersecurity framework* yang sesuai *Standarts Operating Procedure* (SOP) internasional untuk untuk melakukan Analisa forensik digital. Pada gambar 1 menunjukkan tahapan penelitian.



Gambar 1. Alur Penelitian

Tahapan pertama pada penelitian ini adalah skenario serangan *cross-site scripting* menggunakan Topologi Jaringan seperti pada Gambar 2.



Gambar 2. Topologi Jaringan

Pada penelitian ini, skenario penyerangan menggunakan jaringan publik yang dilakukan oleh *attacker*. *Attacker* menyerang *web server*, dan mengeksploitasi *website* dengan teknik serangan *reflected* dan *stored cross-site scripting*. Ketika serangan telah selesai dilakukan oleh *attacker* dan menyebabkan perubahan pada tampilan web server, kemudian *investigator* melakukan penyelidikan forensik digital. Pengujian yang dilakukan menggunakan 4 PC (1 PC *server*, 1 PC *attacker*,1 PC investigator, dan 1 PC *user*).

Setelah skenario serangan terjadi, dilanjutkan dengan tahapan *collection*. Pada tahapan ini dibantu dengan *tools* QEMU untuk mengkonversi sebuah *disk image* barang bukti dimana format dari *disk*

tersebut adalah *qcow2* kemudian dikonversi menjadi *format disk* .img (RAW). Selanjutnya melakukan verifikasi terhadap *disk image* yang telah dikonversi dengan menggunakan *tools* HashCalc. Tujuan dari proses tersebut yaitu, untuk mengakuisisi *server*, kemudian pada proses konversi image bertujuan untuk memungkinkan penggunaan *image disk* dengan berbagai *hypervisor* atau *platform* yang mendukung *format .img* dan memudahkan proses *backup* dan *restore* dengan representasi *disk* yang lebih sederhana, membuat salinan cadangan (*backup*) dari *server* tanpa merubah metadata fisik dari *disk image server* sebelumnya.

Berikutnya tahapan *examination*, hasil dari *disk server* yang telah diakuisisi akan diperiksa secara menyeluruh. Proses pencarian bukti digital ini akan dilakukan dari file *log server* hingga ditemukan bukti serangan yang telah terjadi. Kemudian hasil penemuan barang bukti digital tersebut dilanjutkan ketahapan *analysis.*

Tahapan *analysis* merupakan tahapan untuk melakukan analisis temuan barang bukti digital dari proses sebelumnya. Analisis ini difokuskan pada *access logs* dan *error logs.* Tujuan utama analisis ini difokuskan pada *access logs* dan *error logs*, untuk mengetahui identitas dari penyerang dan ciri-ciri serangan dan ciri-ciri *error* yang disebabkan oleh serangan.

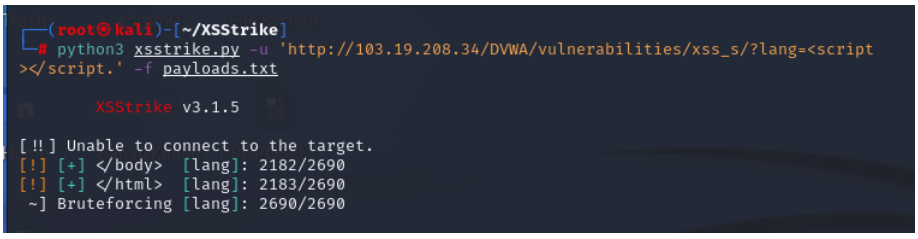
Tahapan terakhir adalah tahapan *reporting*, tahapan ini akan membuat report dari hasil analisis yang telah dilakukan pada tahap sebelumya. Dalam konteks penelitian ini akan membandingkan aktivitas log *server* sebelum dan sesudah serangan terjadi. Hal ini dilakukan untuk membaca dan mencatat seluruh akses pengguna web. Perbandingan tersebut akan dituliskan dalam laporan hasil penemuan barang bukti berupa informasi kasus, tim pemeriksa, barang bukti (*evidence*), dan hasil pemeriksaan, kesimpulan dan penutup.

4. HASIL DAN PEMBAHASAN

4.1 HASIL

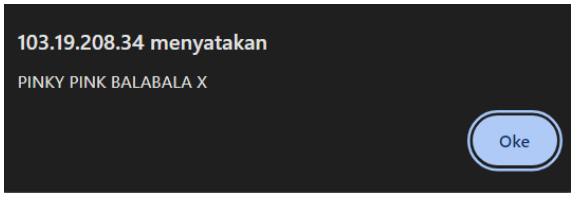
4.1.1 Skenario Serangan Cross-site Scripting

Skenario serangan *cross-site scripting* ini dilakukan dengan menggunakan jenis serangan *reflected cross-site scripting* dan *stored cross-site scripting*. Salah satu *tools* yang dapat digunakan untuk melakukan serangan *cross-site scripting* pada *web* adalah dengan menggunakan XSStrike. *Tool* ini merupakan sebuah *tool free* dan *open-source* yang tersedia di GitHub. XSStrike digunakan untuk mengeksploitasi halaman web dengan menyisipkan *script* berbahaya.



Gambar 3. Serangan Cross-Site Scripting

Pada Gambar 3 dapat dijelaskan bahwa penyerang melakukan serangan dengan menggunakan *tools xsstrike*. Perintah tersebut merupakan upaya dari penyerang untuk menemukan titik kerentanan dari *website* dengan menguji dengan mengirimkan banyak permintaan untuk menemukan celah yang dapat dieksploitasi pada *webiste payloads xss* yang sudah dibuat. Sehingga hasil yang didapatkan dari percobaan serangan tersebut yaitu pada *payloads xss* ke 2182 dan 2183 dapat digunakan untuk disisipkan pada masukan form di halaman website. *Script* yang dapat disisipkan dengan unsur tag <body> dan tag <html> dari 2690 percobaan *payload xss*.



Gambar 4. Pop Up Alert

Skenario selanjutnya menyuntikkan serangan melalui masukan form yang didapatkan dengan

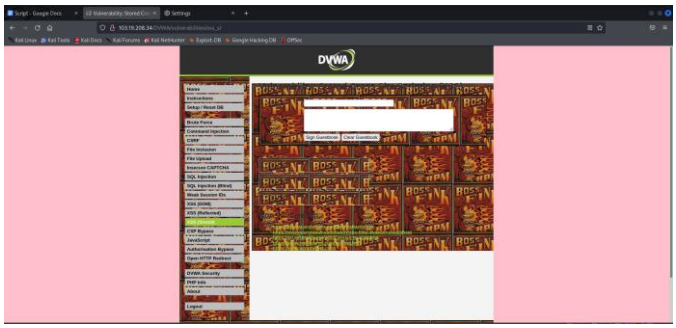
menggunakan unsur tag <body> dan tag <html> pada halaman website. Penyisipan *script* pertama yang dilakukan yaitu membuat pop up alert pada halaman xss_r dengan *script* yang disisipkan

`<script>alert (“PINKY PINK BALABALA X”)</script>` pada masukan *form name*. Sehingga dari penyisipan *script* tersebut memperoleh sebuah pop up alert yang berisikan pesan “PINKY PINK BALABALA X”. Pop up alert tersebut dapat dilihat pada Gambar 4.

Penyerangan selanjutnya pada masukan form halaman xss_s yang bertujuan merubah tampilan halaman tersebut dengan menyisipkan *script* untuk merubah interface website halaman tersebut seperti pada Gambar 5. Adapun *script* yang disisipkan ke dalam *text box* yaitu:

```
<style>
div {
 background-image: url('http://www.deepeddy.net/img/deepeddyfish.gif');
}
</style>
```

Dari hasil serangan yang dilakukan, pada Gambar 5 merupakan hasil menyisipkan *script* yang merubah *background image* pada halaman tampilan website. Ketika *user* memuat ulang halaman tersebut maka *user* akan melihat tampilan halaman berubah menjadi Gambar 5.



Gambar 5. Hasil Serangan pada Form Text Box

4.1.2 Collection

Pada tahapan *collection*, segala informasi mengenai barang bukti dikumpulkan atau didokumentasikan dengan mengunakan *metode static forensic*. Tahap ini pengakuisisian dengan kondisi barang bukti dalam keadaan tidak beroperasi. Investigator bekerja melakukan pengakuisisian dengan cara melakukan konversi *disk image* dikarenakan *server* sebelumnya memiliki *disk image* yang terletak pada *proxmox virtual environtment* (ProxmoxVE). Sebelum melakukan pengkonversian *disk image*, investigator membuat sebuah folder direktori dengan nama *DiskBarangBukti*. Kemudian investigator mengakses *root* ke folder direktori letak barang bukti berada. Pada tahapan ini dilakukan dengan perintah *cd /data/images/110*.

Setelah masuk pada folder direktori dimana file *disk image* berada, kemudian dilakukan pengkonversian barang bukti yang diakuisisi yang awalnya berbentuk *.qcow2* menjadi *file disk image* berbentuk *.img*. Adapun letak dari hasil konversi disk image tersebut berada pada folder direktori *DiskBarangBukti*. Setelah proses selesai selanjutnya investigator melakukan *verification disk image* untuk mengetahui nilai *hash* dari file *.img* yang diakuisisi seperti pada Gambar 6.

Gambar 6. Nilai Hash Disk Image

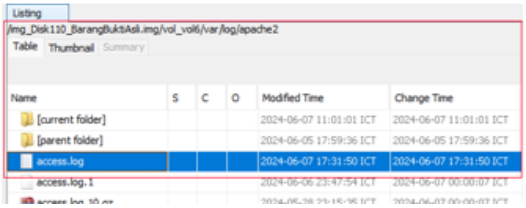
Setelah tahapan proses *backup* barang bukti asli dari *server* utama ke komputer penyidik selesai, Selanjutnya, penyidik mengolah barang bukti dengan menggunakan *verification hash* untuk mendapatkan *nilai hash* dari *disk image* yang dicadangkan apakah terjadi perubahan *metadata* dari *disk*

image tersebut. Hasil yang diperoleh kemudian digunakan untuk analisis lebih lanjut. Adapun hasil dari proses ini akan menampilkan proses *hashing* yang diterapkan pada *file image* untuk melihat bahwa *file* yang telah dicadangkan identik atau sama tanpa ada yang diubah.

4.1.3 Examination

4.1.3.1 Pemeriksaan pada Autopsy

Pada tahap ini merupakan pemeriksaan pada *log access server* yang didapatkan dari pemerikasan dengan mengakses */var/log/apache2/* pada software *Autopsy*. Dari Gambar 7 dapat dijelaskan bahwa hasil pemeriksaan yang dilakukan pada *software Autopsy* yang difokuskan pada pemeriksaan *log server*, didapatkan beberapa file *log access* dari *server*.



Gambar 7. Pemeriksaan log salinan disk image di Autopsy

Adapun penemuan *log access* yang ditemukan seperti data dari *access.log, error.log*. Selanjutnya, investigator melakukan ekstraksi pada *file log access* tersebut. File yang diekstrak berada pada folder direktori *D:\01-Salinan1 Barang Bukti\001-BarangBukti Salinan\Export* komputer investigator. Hal ini bertujuan agar *file log* yang dijadikan fokus utama barang bukti dijamin keasliannya agar tidak terjadi kesalahan dalam pemeriksaan.

Selanjutnya investigator melakukan perhitungan nilai *hash* dari masing-masing file yang diekstraksi dengan menggunkan *tools HashCalc*. Pada Tabel 1, didapatkan hasil perhitungan nilai *hash* dari masing-masing *file Log*.

Tabel 1. Nilai hash file dari setiap file log apache

File Log	Nilai Hash
access.log	446d6da60be86f5841dd65d22688f793
error.log	21507e01397389bba567d55d22e006fd

4.1.3.2 Pemeriksaan pada Kali Linux

Pemeriksaan dengan memanfaatkan sistem operasi Kali Linux bertujuan untuk menyelidiki potensi pelanggaran keamanan atau serangan, menganalisis *disk image* yang bersifat *reflected* dan *stored cross-site scripting*.

Adapun hasil yang didapatkan yaitu berupa *log access* yang menampilkan data-data permintaan yang tercatat pada *file access.log.* Adapun perintah untuk mencari log terkait serangan tersebut yaitu "*grep "POST /DVWA/vulnerabilities/xss_s/" access.log | grep " 200 [2-9][0-9][0-9][0-9]* "" perintah ini berfungsi untuk menemukan *entri log* data post dengan ukuran respons yang lebih besar dari ukuran normal. Ukuran respons dari data post yang besar dapat dincurigai bahwa endpoint /DVWA/vulnerabilities/xss_s/ adalah target penyerangan.

4.1.4 Analysis

4.1.4.1 Analisis Bukti Digital

Berdasarkan hasil akuisisi barang bukti dan pemeriksaan barang bukti digital yang sah, didapatkan sebuah barang bukti dengan nama *file Disk110_BarangBuktiAsli.img* yang telah dilakukan proses akuisisi, penggandaan file, serta telah melewati proses *imaging* dan *hashing* menggunakan *tools FTK Imager.* Hasil preservasi dalam pengakuisisian, penggandaan dan proses *hashing* barang bukti yaitu:

i) Nama *file*: Disk110_BarangBuktiAsli.img
ii) Nilai *hash file*: 84b72460272da46054f48cb0b418c4c0
iii) Ukuran (*bytes*): 53,687,091,200 *bytes*
iv) Waktu dan tanggal akuisisi: 18:15:05, Jumat 7 Juni 2024

4.1.4.2 Analisis Log Apache

Berdasarkan dari hasil pemeriksaan *log apache*, investigator menganalisa dan berhasil menemukan data

log web server yang tersimpan pada direktori */var/log/apache2*. Investigator memfokuskan pencarian barang bukti digital pada direktori file tersebut. Pada file direktori */var/log/apache2* menjadi fokus utama investigator, dikarenakan *log web server* menyimpan seluruh data *log apache* pada *server*.

Hasil analisis *log apache* didapatkan 2 (dua) file yang menjadi fokus utama pemeriksaan yaitu *file access.log* dan *error.log*. *File access.log* berisikan data-data dari seluruh permintaan (*requests*) yang diterima oleh *web server*. Data file access.log biasanya berisi informasi mengenai IP address, tanggal dan waktu permintaan, metode HTTP yang digunakan, URL permintaan, status kode HTTP, ukuran respon yang dihasilkan, dan *user-agent* atau perangkat yang digunakan *client.*

Sedangkan pada *file error.log* berisikan seluruh data kesalahan atau pesan *error* yang terjadi di *server*. Data ini termasuk pesan kesalahan yang terjadi saat memproses permintaan *client* serta kesalahan internal pada *server*. Adapun data yang dicatat berisikan informasi mengenai tanggal dan waktu saat kesalahan terjadi, level kesalahan seperti *warning*, *error*, dan *critical*, kemudian mencatat pesan kesalahan yang menjelaskan terdapat *error* pada bagian tertentu, dan informasi tambahan yang mungkin berguna untuk mendiagnosis masalah pada sisi internal *server* dan *client* seperti *script* atau modul yang menyebabkan kesalahan.

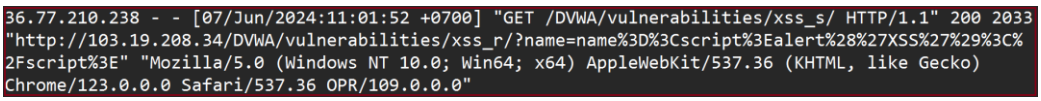
4.1.4.3 Analisis File access.log

Berdasarkan bukti digital pada *file access.log* yang diakses pada direktori */var/log/apache2/* dan pemeriksaan *disk image* pada komputer investigator, Ditemukan beberapa serangan dan IP Address pelaku dengan **IP Address 36.77.210.238**. Hal tersebut dapat dilihat pada waktu dan tanggal 07/Jun/2024:10:52:07, pada waktu dan tanggal tersebut pelaku pertama sekali mengakses website. Hasil temuan tersebut dapat dilihat pada Gambar 8.

Gambar 8. Pelaku Pertama Sekali Mengakses Website

Kemudian investigator melakukan analisis dimana terlihat **IP Address 36.77.210.238** melakukan aktifitas serangan pertama sekali dengan teknik *reflected cross-site scripting* pada website. Hal tersebut dapat dilihat pada waktu dan tanggal 07/Jun/2024:11:01:52 dengan melakukan *method* permintaan ***GET*** dan *url* permintaan yang dikirimkan ke ***/DVWA/vulnerabilities/xss_s/*** dengan parameter ***name*** yang bernilai ***?name=name%3D%3Cscript%3Ealert%28%27XSS%27%29%3C%2Fscript%3E*** parameter tersebut berisikan *JavaScript* yang mencoba menampilkan alert **XSS**.

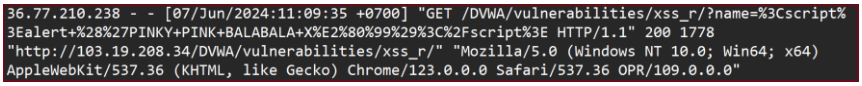
Adapun kode status yang diberikan oleh *server* adalah 302, yang menunjukkan bahwa permintaan tersebut diteruskan ke lokasi lain. Ukuran respons yang diberikan adalah 672 *bytes*. Referer dari permintaan ini tidak disediakan, dan *user agent* yang digunakan adalah Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0. Dari data *log* tersebut, terlihat bahwa terdapat potensi serangan *reflected cross-site scripting* karena nilai parameter name berisi skrip *JavaScript* yang dieksekusi secara langsung oleh browser ketika halaman tersebut dimuat. Analisa log serangan tersebut merujuk pada Gambar 9.



Gambar 9. Data Entri log Indikasi Serangan *Cross-site Scripting*

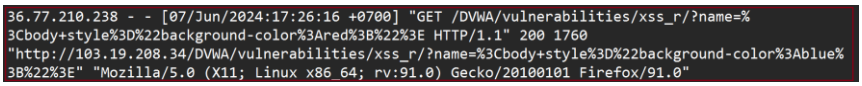
Kemudian pada *entri log* pada tanggal 07 Juni 2024, jam 11:09:35 penyerang mencoba melakukan serangan kembali dengan menyisipkan *JavaScript* dengan *method* permintaan ***GET*** dan *url* permintaan yang dikirimkan ***/DVWA/vulnerabilities/xss_r/*** dengan parameter ***name*** dengan nilai parameter yang digunakan adalah ***%3Cscript%3Ealert+%28%27PINKY+PINK+BALABALA+X%E2%80%99%29%3C%2Fscript%***. Parameter tersebut berisikan *payloads xss* dengan *alert* "PINKY PINK BALABALA X".

Adapun kode status yang diberikan oleh *server* adalah 200, yang menunjukkan respon berhasil dengan ukuran respons yang diberikan 1778 byte. *User agent* yang digunakan adalah Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, seperti Gecko) Chrome/123.0.0.0 Safari/537.36 OPR/109.0.0.0. Log serangan tersebut dapat dilihat pada Gambar 10.



Gambar 10. Log *payloads Cross-site Scripting*

Kemudian investigator melakukan pencarian pada *file access.log*, dimana terdapat indikasi serangan *reflected cross-site scripting*. Adapun log indikasi serangan tersebut terdapat pada Gambar 11 dengan mengubah *background color* pada halaman website.

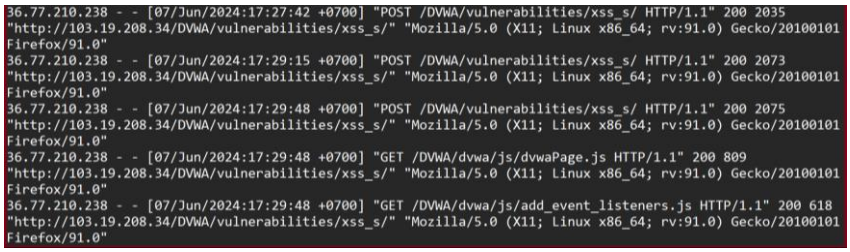


Gambar 11. Log Serangan Merubah *Background* Color

Pada Gambar 11, log tersebut mengindikasikan terjadinya serangan *reflected cross-site scripting.* Penyerang dengan **IP 36.77.210.238** melakukan serangan tersebut menyisipkan *script reflected cross-site scripting* pada tanggal 07 juni 2024, pada jam 17:26:16 adapun method permintaan yang dilakukan yaitu ***GET*** dengan *url* permintaan kehalaman ***/DVWA/vulnerabilities/xss_r/*** dengan parameter ***name*** yang berisi kode ***HTML/JavaScript*** yang dapat menyebabkan serangan *reflected cross-site scripting.* Adapun kode status yang diberikan oleh *server* adalah 200, yang menunjukkan respon berhasil dengan ukuran *respons* yang diberikan 1760 byte. Adapun halaman yang diakses dan disisipkan *script* yaitu ***http://103.19.208.34/DVWA/vulnerabilities/xss_r/?name=<body style="background-color:blue;">.*** *Script* yang disisipkan bertujuan untuk merubah tampilan *background* halaman website menjadi warna biru. Tampilan halaman *website* yang disisipkan *script* ini hanya menampilkan halaman *website* dengan *background* biru yang hanya diakses dan ditampilkan sementara.

Adapun log indikasi serangan *stored cross-site scripting* tersebut terdapat pada Gambar 12. Pada Gambar 12, terlihat **IP 36.77.210.238** melakukan beberapa method permintaan ***GET*** dengan ukuran *respons* sebesar 2035 *bytes*, Pada jam 17:27:42, kemudian pada jam 17:29:48 penyerang kembali melakukan permintaan dengan ukuran *respons* 2075 *bytes*. Pada *method* permintan ***POST*** ke halaman ***/DVWA/vulnerabilities/xss_s/*** yang menunjukkan adanya interaksi dengan formulir yang rentan terhadap serangan *cross-site scripting* pada jam 17:29:15 sampai 17:29:48 dari masing-masing ukuran respons yang diberikan berkisar 2035 *bytes* dan 2075 *bytes*.

Pada baris terakhir, terlihat ada permintaan ***GET*** untuk mengambil file *JavaScript* ***/DVWA/dvwa/js/dvwaPage.js*** dan ***/DVWA/dvwa/js/add_event_listeners.js***. Dari log tersebut terjadi indikasi skenario serangan *stored cross-site scripting*, di mana *script-script JavaScript* dari website *Damn Vulnerable Web Application* (DVWA) dapat disisipkan serangan *stored cross-site scripting*. Adapun rentan waktu penyerang melakukan serangan tersebut terjadi dari jam 17:27:02 sampai 17:29:48. Respons *server* yang diberikan dari script yang disisipkan bernilai 200 yang artinya berhasil di *respons* oleh *server*.



Gambar 12. Log indikasi *Stored Cross-site Scripting*

Investigator melakukan analisa dan menemukan bahwa **IP Address 36.77.210.238** terakhir kali mengakses dan melakukan penyerangan di jam **17:31:50** pada tanggal **07/Jun/2024** dengan mengirimkan serangan *reflected cross-site scripting* ke website yang diserang. Log serangan tersebut dapat dilihat pada Gambar 13. Dengan menggunakan *tools whois*, maka dapat diketahui ISP yang digunakan oleh penyerang dan untuk mengetahui dimana pelaku melakukan serangan.

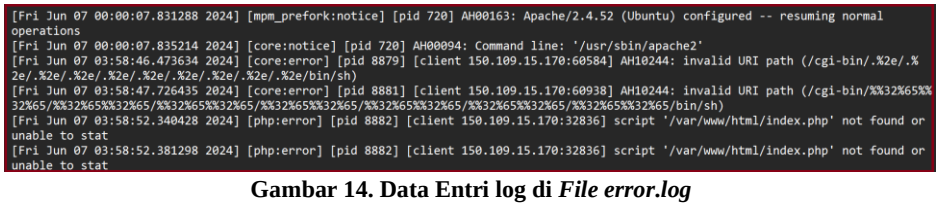
Gambar 13. Serangan Terakhir Kali yang Dilakukan Penyerang

Dari analisa file access.log diatas dapat disimpulkan bahwa IP Address 36.77.210.238 melakukan penyerangan terhadap website Damn Vulnerable Web Application (DVWA) dengan 2 (dua) jenis serangan *cross-site scripting.* Adapun 2 (dua) jenis serangan *cross-site scripting* tersebut berjenis reflected dan stored *cross-site scripting*. Ciri-ciri serangan *reflected cross-site scripting* yang disisipkan pada website yaitu *script* yang disisipkan bersifat sementara atau tidak disimpan di server dan terbatas pada satu sesi yang berarti ketika sesi berakhir atau pengguna memuat ulang halaman, serangan tidak akan berulang. Adapun ciri-ciri khusus dari serangan ini yaitu pembuatan alert, karakter *script* berbahaya dalam *query string* atau *URL seperti =, <, >, '', '*, javascript, dan *tag HTML* dan *JavaScript* seperti karakter *<script>, <img>, </script>, onclick=, onload=.*

Kemudian ciri-ciri serangan *stored cross-site scripting* yang didapatkan pada website yaitu penyisipan kode *HTML/JavaScript* berbahaya ke dalam parameter permintaan, tidak adanya validasi atau sanitasi yang memadai pada masukan yang diterima oleh *server*, dan penyimpanan, serta penampilan kembali kode *HTML/JavaScript* yang berbahaya di halaman web yang membuat *interface website* berubah. Adapun ciri-ciri khusus dari serangan ini yaitu permintaan data *post* ke *endpoint cross-site scripting*, Ukuran respons yang lebih besar dari biasanya.

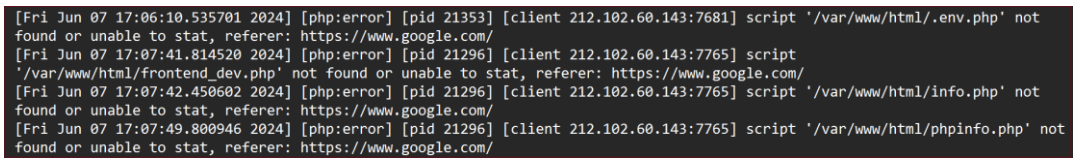
4.1.4.4 Analisis File error.log

Berdasarkan dari hasil pemeriksaan *log apache*, investigator kemudian melakukan analisis pada *file error.log*. File tersebut berada di direktori */var/log/apache2.* File ini menjadi fokus utama investigator, dikarenakan file *error.log* menyimpan informasi tentang berbagai kesalahan dan pesan diagnostik yang terjadi selama operasi *web server*. Adapun data *error.log apache* yang ditemukan berupa pesan *error*, yang ditampilkan pada Gambar 14.



Gambar 14. Data Entri log di *File error.log*

Investigator melakukan analisis pada data *entri* di file *error.log*. Investigator tidak menemukan adanya *entri log* yang menunjukkan adanya permintaan yang mengandung parameter atau data yang berisi kode *JavaScript* yang mencoba menyuntikkan *script* ke halaman web, yang merupakan indikasi serangan *cross-site scripting*. Sebagian besar pesan *error* terkait dengan permintaan ke *URI* tidak valid dan upaya mengakses *script* PHP yang tidak ada di *server* seperti Gambar 15.



Gambar 15. Data Entri log pada *File error.log*

Dari hasil analisa disimpulkan bahwa, data *entri log* pada *file error.log* tidak ada indikasi terjadinya serangan *cross-site scripting*. Pada file error.log tidak menghasilkan bukti apapun terkait terjadinya indikasi serangan. Pada *Entri log* tersebut lebih banyak menunjukkan upaya mengakses jalur *URL* yang tidak valid dan skrip PHP yang tidak ada di *server* seperti yang ditampilkan pada Gambar 15.

4.1.5 Reporting

Berdasarkan hasil pengumpulan, pemeriksaan dan analisis yang telah dilakukan, maka berikut adalah temuan utama yang dihasilkan dari investigasi serangan *cross-site scripting* yang ditemukan:

i) Nama dan spesifikasi perangkat:
   a) Nama: server DVWA
   b) Disk size: 50 GB
   c) Memory: 8 GB
   d) Processor: 2 sockets, 1 core
   e) Software: Linux Ubuntu 22.04.3 LTS
ii) Tersangka penyerangan menggunakan IP Address 36.77.210.238, IP Address tersebut didapatkan dari permeriksaan *timeline* pada *file access.log*.
iii) Tersangka melakukan penyerangan pertama kali pada 07 Juni 2024, jam 11:01:37. Waktu dan tanggal penyerang melakukan serangan pertama kali didapatkan dari permeriksaan *timeline* pada *file access.log*.
iv) Tersangka terakhir kali melakukan penyerangan pada tanggal 07 Juni 2024, jam 17:31:50. Waktu dan tanggal penyerang melakukan penyerangan terakhir kali didapatkan dari permeriksaan *timeline* pada *file access.log*.
v) Tersangka menggunakan teknik *reflected* dan *stored cross-site scripting* untuk melakukan penyerangan ke Web Server.
vi) Tersangka menggunakan *payloads cross-site scripting* untuk melihat kerentanan yang terjadi pada website dan merubah tampilan *interface* website.
vii) ISP yang digunakan tersangka.

4.2 PEMBAHASAN

Berdasarkan hasil analisis forensik digital yang telah dilakukan sesuai dengan metode NIST diperoleh bahwa setiap tahapan NIST dilakukan sesuai dengan prosedur untuk memastikan barang bukti yang ditemukan dapat terjaga integritasnya dan dapat digunakan sebagai barang bukti yang sah dipengadilan. Selain itu, pada penelitian ini juga menggunakan *tools* yang berbeda dengan penelitian sebelumnya. Adapun *tools* yang digunakan merupakan *tools* yang umum untuk proses forensik digital. Penelitian ini juga menghasilkan artifak-artifak yang jelas yang dapat dijadikan barang bukti seperti yang terdapat pada sub-bab 4.1.6. Selain itu, ditemukan juga teknik serangan yang dilakukan yaitu *cross-site scripting* seperti *reflected cross-site scripting* dan *stored cross-site scripting.*

5. KESIMPULAN

Adapun kesimpulan yang dapat diambil mengenai penelitian ini yaitu untuk melakukan forensik digital menggunakan static forensic pada media yang terdampak kasus serangan cross-site scripting dengan memanfaatkan tools QEMU, Autopsy, HashCalc. Static forensic mengacu pada metode NIST untuk melakukan pengumpulan, pemeriksaan, analisis, dan pelaporan barang bukti. Sehingga pada penelitian didapatkan sebuah barang bukti digital yaitu Disk110_BarangBuktiAsli.img. Hasil dari pengumpulan, pemeriksaan, dan analisis penelitian ini menghasilkan rangkuman investigasi yang telah dilakukan dan hasil tersebut dirangkum tersebut menjadi sebuah laporan investigasi forensik digital terkait dengan evidence serangan dari kasus cross-site scripting. Adapun barang bukti yang diperoleh berupa spesifikasi perangkat, IP Address penyerang, waktu penyerangan, teknik serangan serta ISP yang digunakan. Hasil bukti digital ini kemudian dijadikan sebagai barang bukti yang dapat digunakan dipengadilan. Adapun rekomendasi untuk penelitian berikutnya yaitu menggunakan teknik penyerangan yang berberda dan teknik pengumpulan barang bukti berupa live forensic.

DAFTAR PUSTAKA

[1] R. Salamah, I., Lindawati, L., Fadhli, M., & Kusumanto, "Evaluasi Pengukuran Website Learning Management System Polsri Dengan Metode Webqual 4.0," *J. Digit Digit. Inf. Technol.*, vol. Vol. 10, N, pp. 1–10, 2020, doi: 10.51920/jd.v10i1.151.
[2] F. Awaluddin, Amsori, and M. Mulyana, "Tantangan dan Peran Digital Forensik dalam Penegakan Hukum terhadap Kejahatan di Ranah Digital," *Humaniorum*, vol. 2, no. 1, pp. 14–19, 2024, doi: 10.37010/hmr.v2i1.35.
[3] A. E. Saragih, N. Christian, and P. Khoirunisa, "Analisis Penggunaan Barang Bukti Digital di

Dalam Sistem Hukum di Indonesia (Studi Kasus Putusan Nomor 3 K/PID.SUS/2019)," vol. Vol.2, No., no. 2, pp. 504–510, 2024, doi: https://doi.org/10.5281/zenodo.12082755.

[4] R. Indonesia, "Pasal 25 Undang-Undang Nomor 1 Tahun 2024 tentang Perubahan Kedua Atas Undang-Undang Nomor 11 Tahun 2008 tentang Informasi dan Transaksi Elektronik," no. 190185, p. 39, 2024.

[5] Rahmat Inggi and Heri Pebrianto Alam, "Analisis Forensik Web Browser Pada Perangkat Android," *Simtek J. Sist. Inf. dan Tek. Komput.*, vol. 8, no. 1, pp. 215–220, 2023, doi: 10.51876/simtek.v8i1.249.

[6] I. Riadi, R. Umar, and D. Bernadisman, "Analisis Forensik Database Menggunakan Metode Forensik Statis," *J. Sist. Inf. Bisnis*, vol. Vol 9, No, pp. 9–17, 2019, doi: https://doi.org/10.21456/.

[7] A. D. Djayali, "Analisa Serangan SQL Injection pada Server pengisian Kartu Rencana Studi (KRS) Online," *JAMINFOKOM J. Manaj. Inform. dan Komput.*, vol. Vol 1 No 1, pp. 16–24, 2020, [Online]. Available: https://garuda.kemdikbud.go.id/documents/detail/1973246

[8] D. Kurnia, "Analisis Forensik Serangan SQL Injection dan DoS Menggunakan Instrution Detection System Pada Server Berbasis Lokal," *InfoTekJar J. Nas. Inform. dan Teknol. Jar.*, vol. Vol 4 No 2, pp. 0–4, 2020.

[9] M. Mushlihudin, "Analisis Forensik pada Web Phishing Menggunakan Metode National Institute Of Standards And Technology (NIST)," vol. 8, no. 2, pp. 11–23, 2020.

[10] I. W. Ardiyasa, "Analisa Serangan Remote Exploit pada Jaringan Komputer dengan menggunakan Metode Network Forensic," *Explore*, vol. 11, no. 2, p. 46, 2021, doi: 10.35200/explore.v11i2.451.

[11] D. Hariyadi, M. W. Indriyanto, and M. Habibi, "Investigasi dan Analisis Forensik Digital Pada Percakapan Grup WhatsApp Menggunakan NIST SP 800-86 dan Support Vector Machine," *Cyber Secur. dan Forensik Digit.*, vol. 3, no. 2, pp. 34–38, 2020, doi: 10.14421/csecurity.2020.3.2.2193.

[12] Imam Riadi, Abdul Fadlil, and Muhammad Immawan Aulia, "Investigasi Bukti Digital Optical Drive Menggunakan Metode National Institute of Standard and Technology (NIST)," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 5, pp. 820–828, 2020, doi: 10.29207/resti.v4i5.2224.

[13] M. F. Hasa, A. Yudhana, and A. Fadlil, "Analisis Bukti Digital pada Storage Secure Digital Card Menggunakan Metode Static Forensic," *Mob. Forensics*, vol. 1, no. 2, pp. 76–84, 2019, doi: 10.12928/mf.v1i2.1217.

[14] B. S. Santoso and P. M. Sulaksono, "Static Forensic Pada USB Mass Storage Menggunakan Forensics Toolkit Imager," *J. Komput. Terap.*, vol. 8, no. 1, pp. 132–142, 2022, doi: 10.35143/jkt.v8i1.5334.