Penerapan *Framework* OWASP dan *Network Forensics* untuk Analisis, Deteksi, dan Pencegahan Serangan Injeksi di Sisi *Host-Based*

Ade Kurniawan

Teknik Informatika, Universitas Universal Batam, Kepri, Indonesia
adekurniawanrusdy@gmail.com

Abstract— ***The Internet has changed the world. The penetration of internet users in 1995 is only 1 percent of the world population, while in 2018, the figure reached 70 percent or 4.5 billion users. Simultaneously, it was reported that eight of the top ten web sites in the world were at a critical point of vulnerability to attacks by injection methods, such as Cross-Site Scripting (XSS) and Structured Query Language Injection (SQLi). Furthermore, XSS and SQLi attacks can be used by certain parties to steal information or specific purposes. In this paper, we research by conducting attack simulations, analyzing packet data, and finally conducting prevention at host-based. Initial simulations of attacks using social engineering attack techniques by sending a phishing email. At this stage of attack simulation, the attack includes information gathering, webcam screenshots, keyloggers, and spoofers. Furthermore, at the stage of analysis, we do with the approach of network forensics with evidence collection techniques using live forensics acquisition. The final stage is prevent (patching) by creating an application or add-on on the browser side by name, XSSFilterAde. This research contribution offers a broad and in-depth study of how to do a simulation, analysis, and finally prevent. Furthermore, the method of protecting the user or host-based solution in the browser application functions to filter, disable plugins, notify, block, and reduce injection attacks.***

Keywords*—*Network forensics, live forensics, cross-site scripting, OWASP, xenotix.

Abstrak*— Internet telah mengubah dunia. Internet telah mengubah wajah dunia. Penetrasi pengguna internet di tahun 1995 hanya 1 persen dari populasi dunia, sedangkan di tahun 2018 angkanya telah mencapai 60 persen atau 4,5 miliar pengguna. Secara bersamaan, dilaporkan delapan dari sepuluh situs *web* teratas di dunia berada pada titik kritis kerentanan terhadap serangan dengan metode injeksi, seperti: *Cross-Site Scripting* (XSS) dan *Structured Query Language Injection* (SQLi). Selanjutnya, serangan XSS dan SQLi dapat digunakan oleh pihak tertentu untuk mencuri informasi atau untuk tujuan tertentu. Dalam makalah ini, penelitian dilakukan dengan mensimulasikan serangan, analisis paket data, dan terakhir melakukan pencegahan di host-based atau di sisi pengguna. Simulasi awal serangan menggunakan *social engineering attack* dengan cara mengirim sebuah *phishing email*. Pada tahapan simulasi serangan ini, serangan meliputi pengumpulan informasi, *screenshot webcam*, *keyloggers*, dan *spoofer*. Selanjutnya, di tahapan analisis, kami melakukan pendekatan *network forensics* dengan teknik pengambilan barang bukti menggunakan metode *live forensics acquisition*. Tahapan terakhir adalah mencegah (menambal) dengan membuat sebuah aplikasi atau *add-on* di sisi *browser* dengan nama XSSFilterAde. Kontribusi penelitian ini menawarkan sebuah studi secara luas dan mendalam tentang bagaimana melakukan sebuah simulasi, analisis, dan, terahir, melakukan pencegahan (*prevent*). Lebih jauh, metode solusi perlindungan kepada pengguna atau *host-based* dalam aplikasi *browser* berfungsi untuk memfilter, menonaktifkan *plugin*, memberi tahu, memblokir, dan mengurangi serangan injeksi.

Kata Kunci— ***Network forensics, live forensics, Cross-Site Scripting, OWASP, xenotix.***

I. PENDAHULUAN

Saat ini, aplikasi *web* terus berevolusi dengan cepat yang menjadikan mereka menjadi lebih kaya fitur dan kompleks. Kompleksitas teknologi ini terjadi dari meningkatnya permintaan konsumen akan *web* yang semakin menarik. Menanggapi hal ini, tim pengembang perangkat lunak harus meningkatkan frekuensi rilis versi baru aplikasi *web* mereka. Sementara itu, siklus rilis yang lebih cepat kepada publik juga berarti keamanan *web* menjadi lebih sulit untuk diukur [1].

Jumlah serangan siber bertambah setiap tahun yang membuat kuatir setiap individu, pemerintah, institusi swasta, dan bisnis [2]. Laporan tahunan dari Symantec Reports [3] mengatakan serangan rata-rata pada tahun 2015 mencapai 568.734 dan 496.697 per hari pada tahun 2016. Akhir-akhir ini, serangan dengan menggunakan metode injeksi meningkat dan mengakibatkan kerugian di sektor bisnis, pemerintah, komunitas, dan target individu. Masalah ini menjadi perhatian serius dari pemerintah, perusahaan, dan komunitas penelitian [4]. Hasil penelitian *OWASP Top 10 Most Critical Web Application Security Risks* [5], menempatkan *Cross-Site Scripting* (XSS) berada di posisi ketiga. *OWASP Top Ten 2019* [6] melaporkan 77 persen perangkat *browser* di sisi individu dikategorikan rentan dan 16 persen diklasifikasikan pada titik kritis.

Serangan *Cross-Site Scripting* adalah jenis serangan injeksi yang sangat spesifik terhadap aplikasi *web* [7]. Penyerang menyuntikkan skrip berbahaya ke halaman *web* yang dipercaya karena aplikasi *web* tidak memeriksa dan memfilter *input* pengguna secara efisien. Saat halaman *web* tepercaya ditampilkan di *browser* pengguna, skrip berbahaya dijalankan dan informasi pengguna yang sensitif dicuri [8]. Dampaknya, korban akan kehilangan privasi dan informasi sensitif.

OWASP adalah kategori komunitas riset lembaga nirlaba 501c. Keanggotaan OWASP berasal dari para ilmuwan, peneliti, dan sektor swasta yang menerbitkan laporan artikel,

alat/peralatan, dan dokumen yang bersifat *open source* [9]. Penggunaan *OWASP Framework* dalam penelitian ini didasarkan kepada alasan yang telah dikemukakan oleh para peneliti keamanan, didorong oleh perangkat lunak *open source*, dan dibandingkan dengan alat di pasar masih relatif mahal [10]. Selain itu, *framework/tool* lain memiliki masalah dalam penggunaannya, seperti jaminan sistem keamanan dengan teknologi terbaru, dan harus menentukan XSS *payload* secara manual untuk mencari kerentanan [11].

Serangan di sisi pengguna atau *host-based* dapat terjadi pada siapa saja; termasuk Presiden, selebriti, CEO perusahaan [12]. Dalam hal ini, akan dilakukan sebuah simulasi serangan, analisis, dan pencegahan serangan di sisi pengguna. *Host-based attack* dalam penelitian ini akan meluncurkan beberapa serangan, antara lain yang sering atau mungkin terjadi di sisi pengguna, seperti: pengumpulan informasi, *webcam screenshot*, *keylogger*, dan *spoofer*.

Dalam simulasi ini, kami mengasumsikan korban memakai *Firefox Mozzilla* dengan metode serangan awal menggunakan *social engineering attack* dengan mengirim sebuah *phishing email*. Alat yang digunakan adalah *OWASP Xenotix XSS Exploit Framework* v6.2. Untuk menangkap, mengekstraksi, dan menganalisis lalu lintas paket menggunakan teknik *network forensics* dengan pendekatan akuisisi barang-barang bukti mengggunakan teknik *live forensics*. *Network forensics* (forensik jaringan), umumnya mengacu pada studi ilmiah dengan bukti berbasis jaringan. *Network forensics* adalah bidang studi untuk menemukan petunjuk kejahatan di internet [13].

Alat yang digunakan untuk analisis paket data menggunakan *Wireshark* dan *Live HTTP Header*. Selanjutnya, pada tahap *prevent solution*, pendekatan dilakukan dengan membuat *patching stages. Prevent solution* dilakukan dengan membuat *browser patching* pada *Mozilla Firefox* dalam bentuk *add-on extension* dengan nama *"XSSFilterAde*" yang mampu menyediakan fungsi: peringatan dini, mematikan *plugin*, dan membatasi *payload/script* kepada korban ketika akan membuka alamat situs *web*.

II. METODOLOGI

A. Forensik

Arti dari kata forensik adalah *"menghadirkan ke pengadilan"* sedangkan istilah "forensik" berasal dari bahasa Latin yang berkaitan dengan hukum atau menerapkan analisis ilmiah dalam konteks hukum. *Digital forensics* adalah proses ilmiah atau upaya ilmiah yang didasarkan pada ilmu mengumpulkan, menganalisis, dan menyajikan bukti di pengadilan untuk membantu pengungkapan kejahatan melalui pengungkapan bukti yang disahkan oleh hukum dan peraturan [14].

Forensik jaringan adalah proses ilmiah untuk mengidentifikasi, menganalisis, dan merekonstruksi peristiwa berdasarkan bukti digital dari jaringan dalam bentuk bukti *log* [15]. Saat ini, alat, teknik, dan keterampilan yang berkualitas para penyidik forensik jaringan diperlukan karena *cybercrime* semakin meningkat dan semakin canggih [16]. Alat *network analyzer* yang digunakan oleh peneliti dalam mencari, mengidentifikasi, dan menganalisis bukti log adalah *Wireshark*, *Tcpdump* dan *Network Miner*.

A. Cross-Site Scripting

Cross-Site Scripting, seperti terlihat pada Gambar 1, adalah salah satu jenis serangan kode injeksi *(code injection attack*) dengan memasukkan kode klien HTML atau kode skrip untuk dimuat ke situs [17].

Serangan ini seolah berasal dari situs resmi, sebagai akibat dari serangan ini, antara lain, penyerang dapat memintas keamanan di sisi klien, mendapatkan informasi sensitif, atau menyimpan aplikasi jahat [18]. *Cross-Site Scripting* dapat diklasifikasikan ke dalam tiga kategori utama:

1) *Stored Cross-Site Scripting (Persistent).*

Serangan *Cross-Site Scripting* melibatkan penyerang untuk menyuntikkan skrip yang disimpan (disebut *payload*) secara permanen (tetap) dalam aplikasi target (misalnya dalam *database* atau *web browser*). Contoh klasik yang disimpan XSS adalah skrip berbahaya yang dimasukkan oleh penyerang pada kolom komentar di *blog* atau pada kiriman forum [19].

2) *Reflected Cross-Site Scripting*.

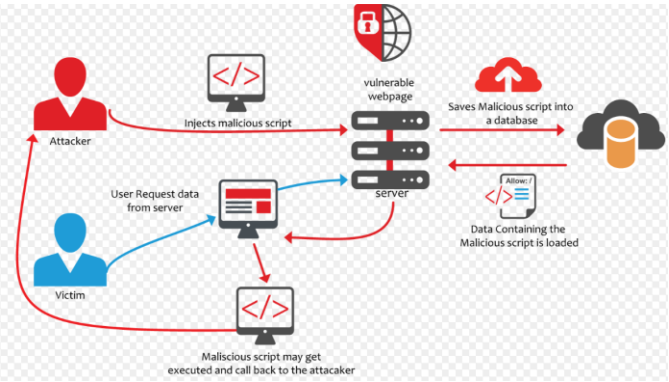
Serangan oleh penyerang untuk mengirim skrip *payload* merupakan bagian dari permintaan yang dikirim ke *web server* kemudian direfleksikan sedemikian rupa sehingga respon HTTP juga termasuk muatan permintaan HTTP [19].

3) *DOM-based Cross-Site Scripting.*

Cross-Site Scripting berbasis *Document Object Model* (DOM) adalah jenis serangan *Cross-Site Scripting* lanjutan yang dimungkinkan ketika aplikasi web dari skrip sisi klien untuk pengguna memberikan data ke (DOM) [20].

B. Serangan di Sisi Pengguna/Host-Based

Serangan di sisi pengguna adalah serangan yang biasanya diawali menggunakan metode rekayasa sosial yang menargetkan sistem, perangkat, atau korban dengan mengeksploitasi kelemahan dan kerentanan sistem atau perangkat pengguna. Berikut ini adalah empat serangan dalam korban serangan sisi pengguna.



Gambar 1 Serangan skrip situs lintas

1) Pengumpulan Informasi

Pengumpulan informasi adalah mencari, menemukan, mengumpulkan, dan memanfaatkan layanan protokol dan informasi penting yang digunakan oleh korban, salah satu dari *firewall*, IP, ISP, dan lokasi yang terdapat pada perangkat target.

2) Keylogger

Keylogger adalah salah satu jenis *spywares*. *Keylogger* berfungsi mencuri informasi pengguna, melacak setiap pengguna *keyboard keystroke*, dan menyimpannya dalam bentuk *log file* [21].

3) Spoofer

Unduh spoofer adalah serangan di mana penyerang berpura-pura menjadi layanan resmi dengan mengirim alamat untuk mengunduh *file* ke korban yang sebelumnya telah disusupi oleh *malware* [22].

4) Screenshot webcam

Screenshot webcam atau cuplikan layar *webcam* adalah serangan di mana penyerang mengambil gambar atau video dengan merekam aktivitas kamera tanpa disadari oleh korban.

C. Open Web Application Security Project (OWASP) Framework

Open Web Application Security Project (OWASP) Framework adalah organisasi nirlaba yang bertujuan membantu organisasi mengembangkan, membeli, dan memelihara aplikasi perangkat lunak yang dapat dipercaya [19]. OWASP adalah *open source*, telah diakui di berbagai forum. Profesional teknologi informasi dan jaringan dapat membangun keahlian *Xenotix OWASP XSS Framework Exploit*, yaitu aplikasi yang dibuat oleh Ajin Abraham untuk pengujian penetrasi, mendeteksi, dan mengeksploitasi kerentanan *Cross-Site Scripting* dalam aplikasi *web* [11]. Ajin Abraham membangun *database* lebih dari 5000 *payload Cross-Site Scripting* (XSS) di VB.net.

Tahap persiapan dimulai dengan pengaturan perangkat keras dan perangkat lunak yang akan digunakan dalam penelitian ini. Perangkat keras yang digunakan dalam penelitian ini adalah *notebook* dengan spesifikasi: prosesor Intel (R) Core (TM) i7-6500U CPU @ 2.30GHz, 8GB RAM, 250GB SSD, Intel 530 *Graphics Card*, dan aplikasi mesin virtual (*VMware Workstation 12*) untuk menjalankan penyerang perangkat dan server.

Sistem operasi yang digunakan oleh penyerang adalah Windows 10 dan aplikasi untuk melakukan simulasi serangan adalah *Xenotix OWASP XSS Exploit Framework v6.2*. Sistem operasi yang digunakan oleh korban adalah Windows 10 dan aplikasi untuk *browser* adalah *Mozilla Firefox* v49. Di sisi server menggunakan sistem operasi Windows 8.1 dan menjalankan *localhost* tempat menyimpan situs *web*. *Backdoor file* menggunakan perangkat lunak XAMPP v5.6. Ilustrasi metodologi ditunjukkan pada Gambar 2 dengan penjelasan sebagai berikut.

A. Attacking (Seranggan)

Serangan sisi pengguna adalah serangan terhadap berbagai sasaran secara terencana dan terstruktur. Dalam penelitian ini, *host-based*/korban yang disimulasikan akan dieksploitasi pada berbagai metode serangan, termasuk pengumpulan informasi, *screenshot webcam*, *keylogger*, dan *spoofer*. Tahap serangan dimulai dengan tahap persiapan. Tahapan persiapan dimulai dengan pemindai kerentanan, *phishing email*, dan korban serangan sisi pengguna .

1) Vulnerability Scanner

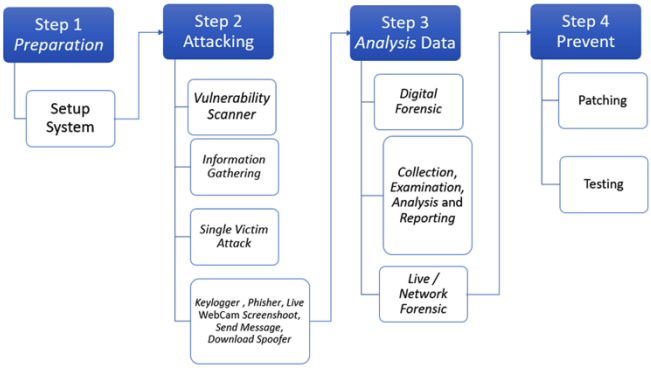
Vulnerability adalah kelemahan yang dapat menyebabkan mesin dapat dihentikan, dimodifikasi, atau diambil alih oleh pihak ketiga. Pemindai kerentanan adalah teknik yang menggunakan komputer untuk mencari kelemahan dalam keamanan komputer lainnya yang memungkinkan penyerang mengurangi informasi sistem [23]. Proses pemindaian kerentanan adalah langkah-langkah untuk mencari, menemukan, dan mengeksploitasi kelemahan aplikasi *web* dengan menggunakan teknik atau alat tertentu. Alat yang digunakan dalam penelitian ini menggunakan OWASP Xenotix v6. Proses tahapannya adalah untuk menyuntikkan *Cross-Site Scripting payload* ke dalam kotak pencarian di *web*.

2) Mengirim Email Phishing

Email phishing adalah bentuk pencurian identitas secara *online* yang bertujuan untuk mencuri informasi sensitif, seperti kata sandi, informasi kartu kredit, dan sistem operasi pengguna dengan cara menyamar sebagai entitas yang sah [24]. Proses pengiriman *email phishing* kepada korban dari penyerang menggunakan teknik *social engineering*. Proses ini untuk mengambil keuntungan dari kecerobohan korban yang diberikan penawaran gratis dengan meng-klik tautan tertentu.

B. Analisis

Integritas bukti digital memainkan peran penting dalam proses investigasi forensik [25]. Dimulai dengan prosedur yang sesuai dari tahapan cara bukti digital dikumpulkan, lalu ekstraksi data dari bukti elektronik menjadi bukti informasi digital dengan metode ilmiah, dalam bentuk biner, baik yang disimpan atau dikirim [26].



Gambar 2 Metodologi

Sumber bukti digital dalam penelitian ini diperoleh dengan menggunakan *Wireshark*, *HashMyFile*, dan *Mozilla extension Live HTTP Headers*. Proses analisis dibagi menjadi tiga tahap berbeda sesuai dengan sumber bukti digital. Dua tahapan tersebut adalah: analisis lalu lintas dan analisis *file hashing*.

1) Analisis Lalu Lintas.

Analisis lalu lintas jaringan adalah proses pencatatan, peninjauan, dan analisis lalu lintas jaringan untuk menilai kinerja jaringan secara umum dan keamanan. Sumber dalam analisis bukti digital diperoleh dari trafik lalu lintas dengan menggunakan *Wireshark* dan ekstensi tambahan *Live HTTP Header*. Untuk komunikasi data ke permintaan penyerang, korban dan server menggunakan *Wireshark* untuk menangkap, mengekstraksi, dan menganalisis.

2) Analisis File Hashing

Sumber bukti digital diperoleh dari *file* backdoor-master.zip yang diunduh oleh korban dalam serangan. *Spoofer file* unduh backdoor.exe di mesin server.

C. Solusi Pencegahan

Pencegahan dalam serangan di sisi pengguna pada *Cross-Site Scripting* ditambal untuk membuat *add-on* atau ekstensi tambahan di Mozilla Firefox. Penambalan dengan menggunakan metode *add-on* dimaksudkan untuk memberikan peringatan dini, menonaktifkan *plugin*, membatasi, membolehkan *payload*/skrip kepada korban ketika akan membuka alamat situs *web* ekstensi Mozilla Firefox bernama XSSFilterAde. Beberapa fitur yang terdapat dalam *add-on* tersebut adalah fitur yang diizinkan saat semua halaman ini, fitur yang memungkinkan semua halaman ini, dan fitur yang memungkinkan skrip ketiga secara global.

III. HASIL DAN PEMBAHASAN

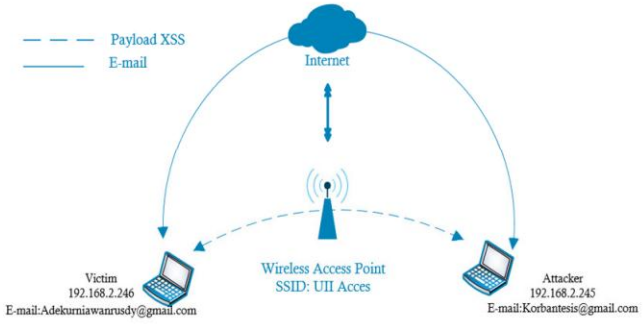
A. Simulasi Serangan Di Sisi Pengguna

Dalam penelitian ini, diasumsikan penyerang sudah mengetahui surel korban dan berada di jaringan yang sama. Dalam hal ini, korban dan penyerang berada di *local area network* yang sama. Gambaran bagaimana proses simulasi kasus ini ditunjukkan pada Gambar 3. Terlihat pada gambar tersebut penyerang mengidentifikasi IP 192.168.2.245 melalui email di korbantesis@gmail.com yang terhubung di jaringan tempat umum. Setelah itu, penyerang mengirim email phishing yang telah disuntikkan muatan XSS menggunakan email OWASP Xenotix ke email korban di adekurniawanrusdy@gmail.com dengan IP 192.168.2.246. Setelah korban menerima *email* tersebut dan meng-klik tautan yang telah disuntikkan dengan muatan, penyerang melakukan serangan, seperti yang ditunjukkan pada Gambar 4.

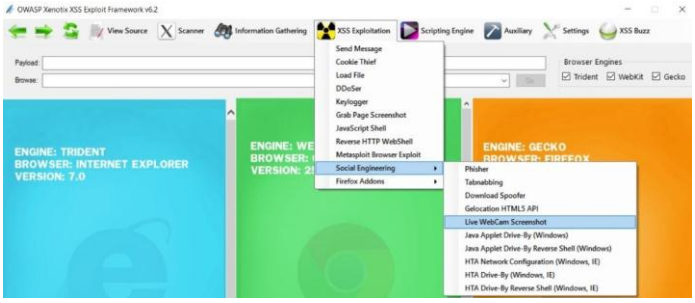
Langkah selanjutnya penyerang melakukan serangan di sisi *host-based* dengan menggunakan teknik *social engineering*. Pertama, melakukan serangan *information gathering* dengan maksud mencuri informasi penting yang terkait pada diri korban. Selanjutnya, melakukan *keylogger* dengan merekam setiap ketukan *keyboard* dari setiap *input*. Serangan *spoofer download* diluncurkan kepada korban untuk keperluan mengunduh *malware* dan serangan. Terakhir, *webcam screenshoot* untuk merekam aktivitas *live streaming* korban.

Selama proses serangan, data lalu lintas dan aktivitas penelusuran korban diakuisisi dan direkam dengan menggunakan *Wireshark* dan *Live HTTP Header* dalam bentuk bukti *digital file* yang berekstensi buktidigital.pcap. Terlihat di Gambar 5, hasil serangan di sisi pengguna dalam serangan pengumpulan informasi. Hasil yang didapatkan dari penyerang yaitu kode negara, IP negara, IP publik, *Internet Service Provider* (ISP), dan lain-lain yang terkait dengan kondisi dan posisi korban.

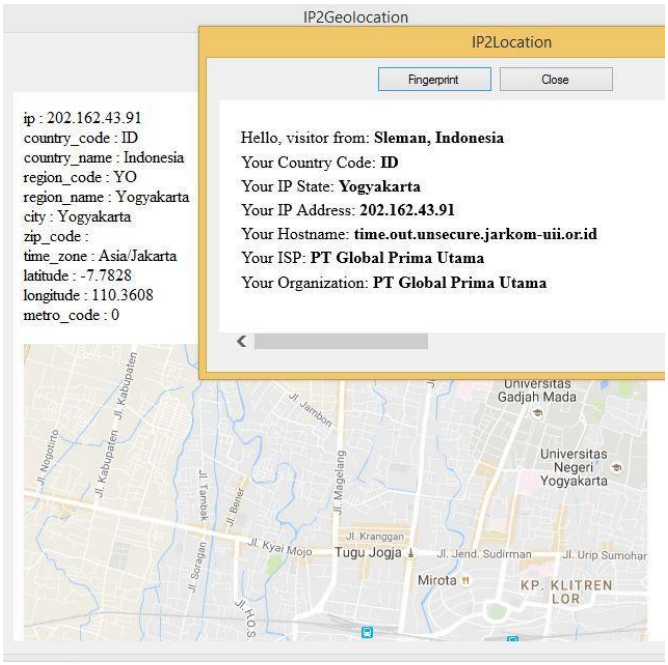
Gambar 6 menunjukkan hasil serangan di sisi pengguna pada serangan *keylogger*. Terlihat pada Gambar 6 tersebut, korban menuliskan *username* dengan kata adekurniawanrusdy@gmail.com, untuk masuk ke halaman media sosialnya di https://www.facebook.com/.



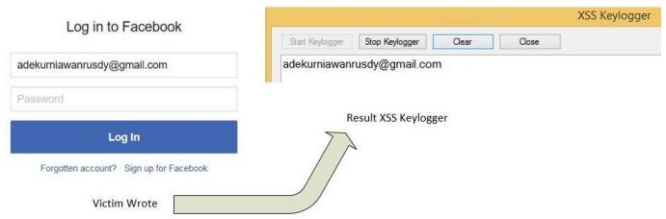
Gambar 3 Alur diagram serangan di sisi pengguna



Gambar 4 *OWASP Xenotix XSS Eksploitation*



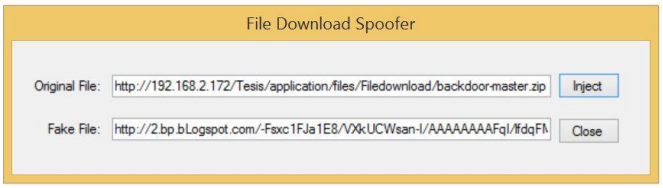
Gambar 5 Pengumpulan informasi



Gambar 6 Hasil *keylogger*

Hasil serangan di sisi pengguna pada serangan *spoofer* ditampilkan pada Gambar 7. Serangan *spoofer* adalah sebuah serangan yang memanipulasi korban dengan sebuah unduhan palsu atau tautan palsu untuk menggunduh sebuah *file* atau aplikasi jahat (*malware*). Terlihat pada gambar tersebut, penyerang berhasil mengelabuhi korban dengan mengunduh sebuah *file* jahat dengan nama file backdoormaster.zip.

Simulasi terakhir, seperti terlihat di Gambar 8, menunjukkan hasil serangan di sisi pengguna pada tangkapan layar *webcam*. Serangan ini bermaksud mengaktifikan *webcam* korban dan melakukan *streaming* atau menunjukkan video korban.



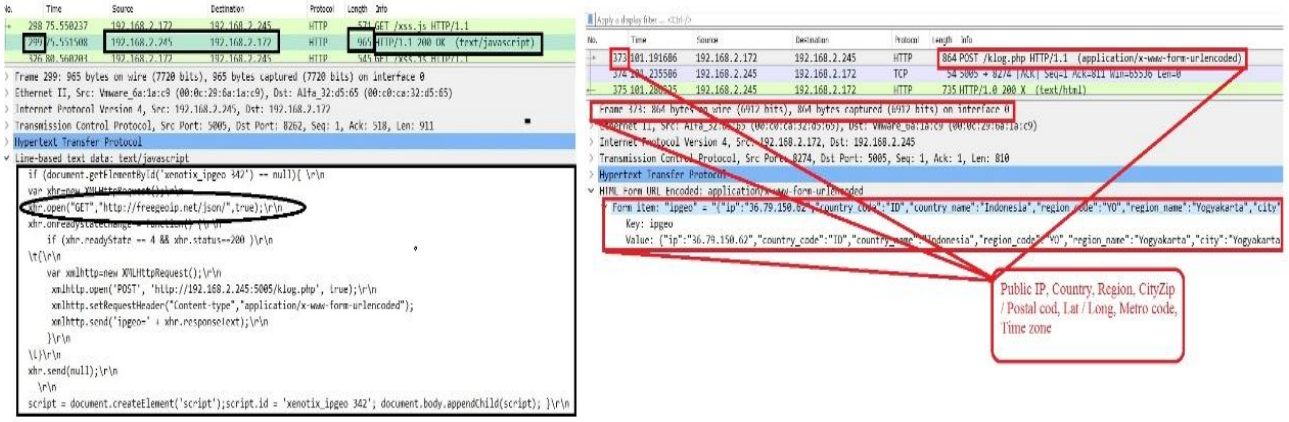
Gambar 7 Unduh *spoofer*

Gambar 8 *Screenshot webcam* langsung

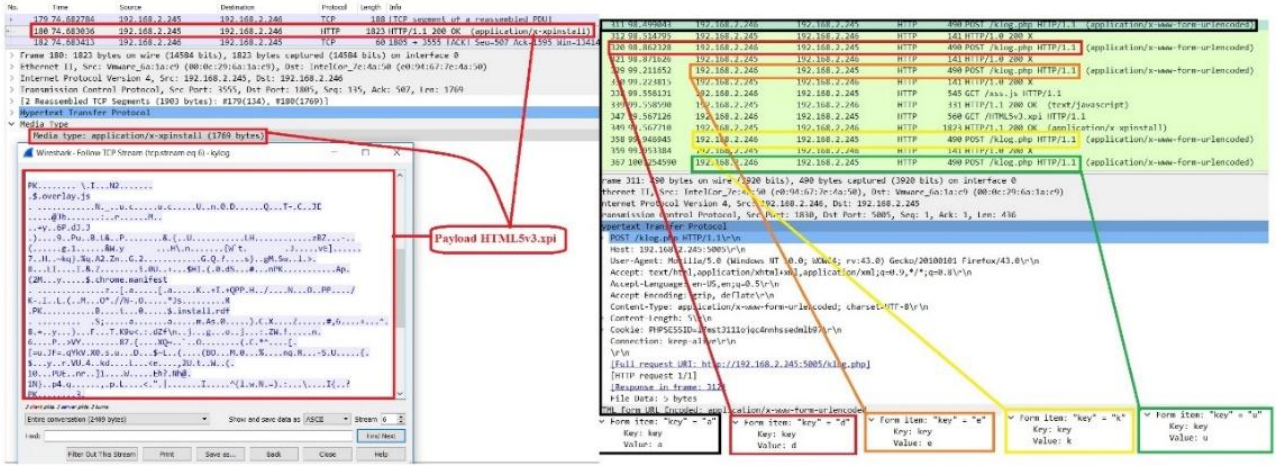
B. Analisis Jaringan Korban Serangan Sisi Pengguna

Pengumpulan informasi dan analisis serangan menggunakan Wireshark. Pada paket *frame* 299 dan 373, terlihat lalu lintas data paket yang dikirim dari korban ke penyerang. Hasil pengumpulan informasi pada Gambar 9 menunjukkan hasil skrip *payload* yang digunakan oleh penyerang yang berhasil diketahui.

Skrip *payload* tersebut berfungsi untuk memaksa, mengalihkan, dan dengan tanpa diketahui oleh korban, membuka situs *web* http://freegeoip.net/. Selanjutnya, tujuan dari penyerang adalah dapat mengetahui semua informasi penting dari korban: IP publik, negara, wilayah, kode pos, *latitude*/*longitude*, kode metro, dan zona waktu.



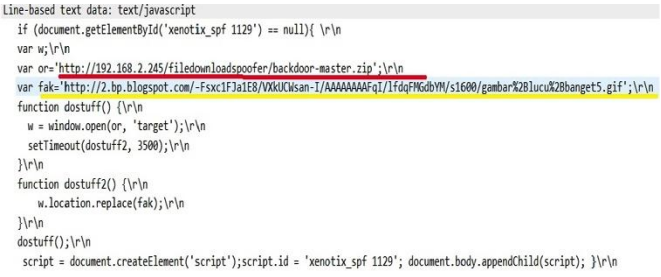
Gambar 9 Pengumpulan informasi skrip muatan dan IP publik, negara, wilayah, kodepos, *latitude/longitude*, kode metro, zona waktu



Gambar 10 Muatan HTML5v3.xpi

Analisis hasil menunjukkan *keylogger payload* dikirim oleh penyerang, terlihat pada Gambar 10. Selanjutnya, dengan metode *Live Forensics* oleh *Wireshark* berhasil menangkap, mengekstraksi, dan menganalisis. Analisis paket 580 menunjukkan muatan yang dikirim oleh penyerang menggunakan ekstensi *file* HTML5v3.xpi. Pada *frame* paket 311 hingga *frame* 641, lalu lintas data menggunakan protokol TCP dan panjang paket 60. Analisis forensik jaringan secara mendalam berhasil menangkap kunci *input* dari korban yang menulis adekurniawanrusdy@gmail.com (Gambar 10). Analisis hasil menunjukkan bahwa skrip *spoofer payload* yang dikirim oleh penyerang berhasil diekstrak dan dianalisis oleh *Wireshark* dengan metode *Live Forensic*.

Analisis menggunakan *Wireshark* ditunjukkan pada Gambar 11. Hasil tersebut menjelaskan cara penyerangan yang digunakan untuk menipu korban, dengan dua alamat tautan palsu dan asli, untuk mengunduh *file* backdoor-master.zip. Hasil ekstraksi dan analisis terhadap uji integri-



Gambar 11 Dua alamat tautan untuk menipu korban

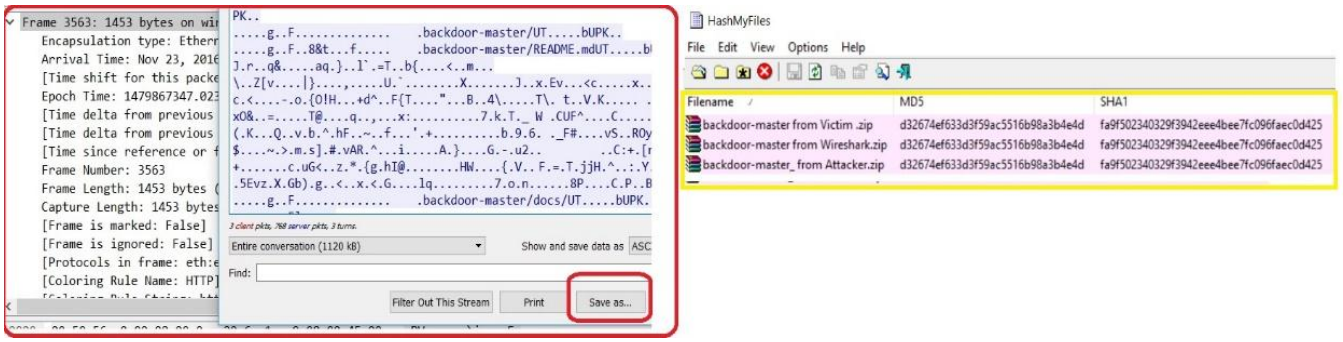
tas data dapat dilihat pada Gambar 12. Lingkaran dalam kotak merah pada gambar tersebut adalah tahap ekstraksi *file* backdoormaster.zip. Lingkaran kuning adalah hasil perbandingan nilai kedua *hash file*, antara *file* yang telah diunduh oleh korban dengan ekstraksi *file Wireshark*. Pada ekstraksi *Wireshark*, korban mengunduh *file* yang tersimpan di *server*.

Uji integritas data *file* yang telah diunduh oleh korban dengan *file* yang disimpan di *server* sangat penting dilakukan oleh penyelidik digital untuk menguji apakah *file* tersebut otentik atau tidak. Aplikasi yang digunakan untuk melakukan *file hashing* adalah *HashMyfile* dengan algoritma MD5 dan SHA1.

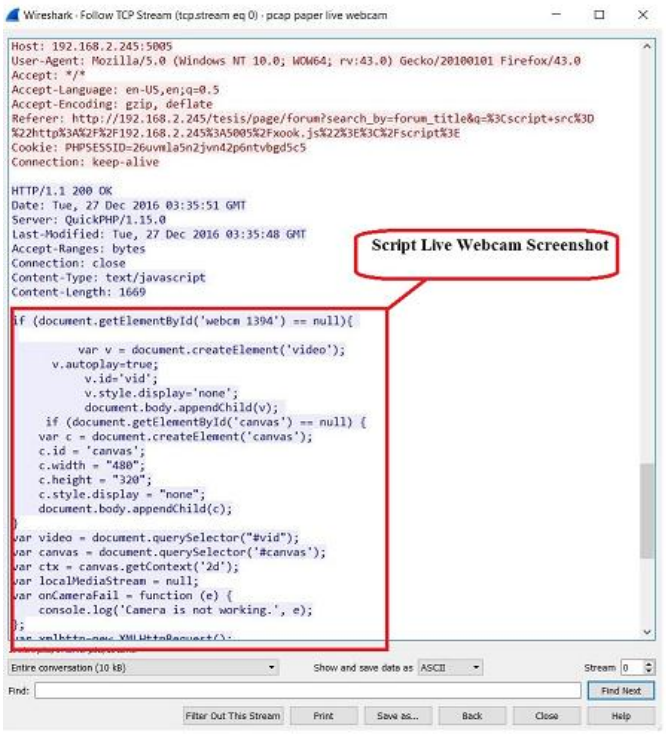
File yang digunakan terdiri atas tiga *file* dari *file Analysis on Live Webcam Screenshot* yang menunjukkan hasil *payload* yang dikirim oleh penyerang berhasil ditangkap, diekstraksi, dan dianalisis oleh Wireshark dengan metode Live Forensik. Ekstraksi dan analisisnya menggunakan file *PCAP Wireshark*. Gambar 13 menunjukkan bahwa korban menerima paket *payload* 350 dan *script* untuk melakukan serangan *Screenshoot Live Webcam*. Gambar 13 juga memperlihatkan hasil analisis paket 903 yang menggambarkan muatan yang digunakan oleh penyerang untuk menyerang *Screenshot Webcam Live*. Hal ini dimaksudkan untuk mengambil keuntungan di sisi kelemahan korban untuk meng-klik tautan yang dapat mengaktifkan kamera korban. Gambar 14 menggambarkan aliran data hasil *webcam streaming* langsung dari korban ke penyerang.

C. Mencegah dan Menambal

Gambar 15 menunjukkan tampilan rentang XSSFilterAde dan Pengaturan XSSFilter pada Mozilla Firefox. Pada menu Pengaturan XSSFilter dibagi menjadi submenu *General Setting, Whitelist, Embeddings, Appearance, dan Advanced Notification*.



Gambar12. Ekstraksi dan *hashing file*

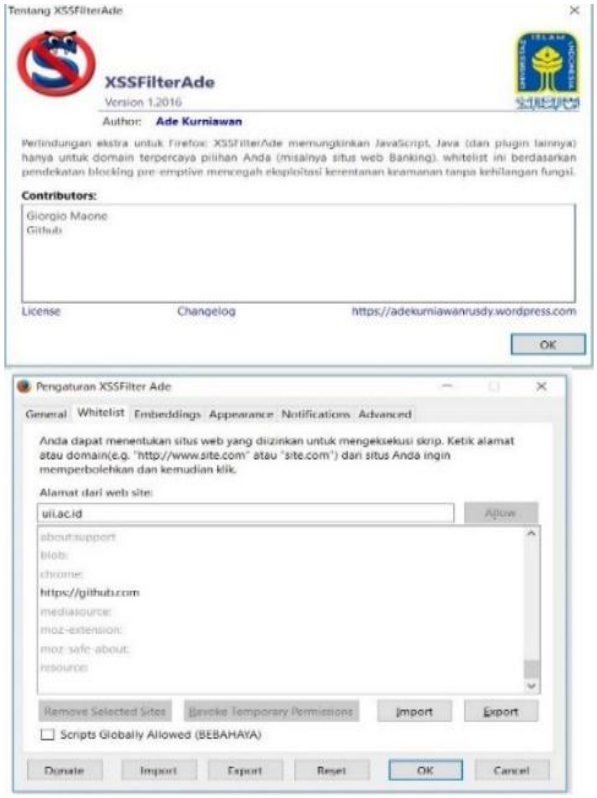


Gambar 13 *Script live webcam screenshot*

Gambar14 *Screenshot webcam* langsung ekstraksi

Menu XSSFilterAde membuat pengaturan secara umum untuk *add-on*. Submenu *Whitelist* membuat pengaturan di situs terpercaya. Submenu *Embeddings* melakukan pengaturan *plugin*, seperti Java, Adobe Flash dan Microsoft Silverlight, yang mungkin pada gilirannya, tampilan submenu ini membuat penyesuaian ke notifikasi di menu *sidebar*. Submenu Notifikasi membuat notifikasi pengaturan. Submenu ini akan ditampilkan jika tidak ada *payload* atau skrip yang menemukan skrip *Cross-Site Scripting*. Selanjutnya, submenu *Advanced* melakukan pengaturan tambahan untuk *plugins* dan lainnya.

XSSFilterAde pada pengaturan memiliki tiga fungsi utama, seperti yang ditunjukkan pada Gambar 16. Tiga fungsi utama memberikan hak kepada pengguna untuk melakukan pengaturan setiap kunjungan situs *web* (ditunjukkan dalam kotak merah).



Gambar15 Tentang dan Pengaturan

Gambar16 Fitur utama XSSFilterAde

1) *Pengaturan 1*:

   Memberikan hak istimewa ke *browser* Biarkan Sementara Semua Halaman Ini,

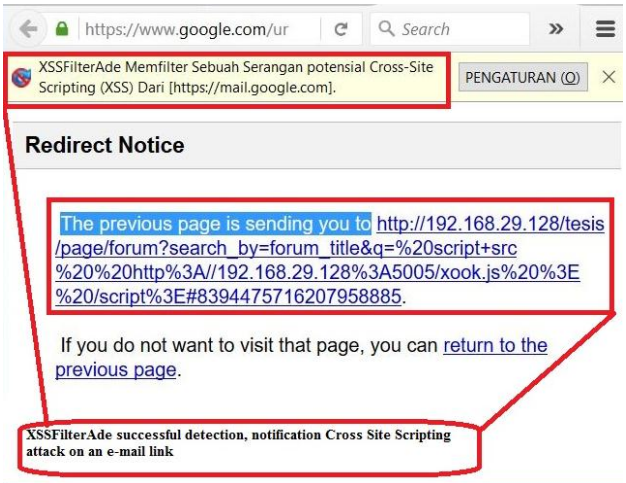
2) *Pengaturan 2*:

   Hak istimewa ke *browser* Izinkan Semua Halaman.

3) Pengaturan 3:

   Izinkan Skrip Secara Global (Berbahaya).

Gambar 17(a) menunjukkan XSSFilterAde berhasil mendeteksi, memfilter, memblokir, dan memberi tahu serangan *Script* Lintas Situs. Kemudian, skrip *payload* disuntikkan ke *email* oleh penyerang dengan skrip ke *browser* pengguna. Selanjutnya, pada Gambar 17(b) XSSFilterAde berhasil mendeteksi, memfilter, memblokir, dan memberi tahu pengguna. Korban akan menjadi lebih waspada bahwa ada muatan yang telah disuntikkan oleh penyerang ke situs *web* dengan sebuah bentuk skrip kait dari *Cross-Site Scripting*.



(a)

(b)

Gambar17 Hasil sukses *email phishing* XSSFilterAde

IV. KESIMPULAN

Analisis forensik dan pencegahan serangan XSS menggunakan OWASP dan *network forensics* mencakup tiga tahap penting, yaitu: tahap serangan, analisis, dan penambalan. Tahapan serangan adalah melakukan kegiatan simulasi dengan metode di sisi pengguna menggunakan OWASP Xenotix XSS Attack Exploit Framework v6.2. Serangan dimaksudkan untuk pengumpulan informasi, *keylogger*, unduh *spoofer*, dan *webcam screenshot* langsung kepada korban melalui Mozilla Firefox.

Tahapan analisis dilakukan menggunakan pendekatan *live forensics* dengan mengunakan Wireshark, HTTP *header*, dan Tcpdump. Penggunaan metode *live forensics* memungkinkan untuk menangkap semua jenis kegiatan yang terjadi, seperti permintaan, muatan, dan skrip. Hasil tahapan analisis ini berhasil merekontruksi serangan dengan mengungkapkan cara dan file skrip yang digunakan oleh penyerang. Beberapa *file* atau bukti digital, dalam nilai *hash test*, dengan aplikasi untuk menggunakannya untuk membandingkan nilai integritas *file* yang telah diunduh oleh *file* korban yang disimpan di *server*.

Tahap terakhir adalah pencegahan, di mana proses dengan membuat tambalan (*paching*) di sisi pengguna dengan memasang ekstensi *add-on* di Mozilla Firefox, dengan nama `XSSFilterAde`. XSSFilterAde adalah aplikasi *add-on* di sisi *browser* untuk peringatan dini, penonaktifan *plugin*, pembatasan, dan perijinan *payload*/skrip kepada korban ketika akan mengunjungi dan/atau membuka alamat situs *web*. Tiga pengaturan utama di XSSFilerAde adalah Sementara Mengizinkan Semua Halaman Ini, Izinkan Semua Halaman Ini, dan Izinkan *Script* Secara Global (berbahaya).

UCAPAN TERIMA-KASIH

Terima kasih kepada Kementerian Riset, Teknologi, dan Pendidikan Tinggi Republik Indonesia atas skema Penelitian Dosen Pemula tahun anggaran 2019 dengan Nomor Kontrak: 020/LPPM/UVERS/IV/19.

DAFTAR REFERENSI

[1] J. Williams, J. Manico, dan N. Mattatall, "XSS (Cross Site Scripting) Prevention Cheat Sheet," *OWASP*, 2018.

[2] Microsoft, "Microsoft Security Intelligence Report," vol. 21, hlm. 7–8, 2016.

[3] Symantec, "015 Internet Security Threat Report," *Internet Secur. Threat Rep.*, vol. 20, no. April, hlm. 119, 2017.

[4] J. Fonseca, N. Seixas, M. Vieira, dan H. Madeira, "Analysis of field data on web security vulnerabilities," *IEEE Trans. Dependable Secur. Comput.*, vol. 11, no. 2, hlm. 89–100, 2014.

[5] OWASP, "OWASP Top 10-2017 - The Ten Most Critical Web Application Security Risks," *OWASP*, 2017.

[6] The OWASP Foundation, "OWASP API Security Top 10-2019 - The Ten Most Critical API Security Risks," hlm. 1–31, 2019.

[7] M. Parvez, P. Zavarsky, dan N. Khoury, "Analysis of effectiveness of black-box web application scanners in detection of stored SQL injection and stored XSS vulnerabilities," dalam *10th Int. Conf. Internet Technol. Secur. Trans. ICITST 2015*, hlm. 186–191, 2016.

[8] G. Dong, Y. Zhang, X. Wang, P. Wang, dan L. Liu, "Detecting cross site scripting vulnerabilities introduced by HTML5," dalam *2014 - 11th Int. Jt. Conf. Comput. Sci. Softw. Eng. "Human Factors Comput. Sci. Softw. Eng. - e-Science High Perform. Comput. eHPC, JCSSE 2014*, hlm. 319–323, 2014.

[9] B. Appiah, E. Opoku-Mensah, and Z. Qin, "SQL injection attack detection using fingerprints and pattern matching technique," *Proceedings of the IEEE International Conference on Software Engineering and Service Sciences, ICSESS*, 2018.

[10] OWASP, "4.0 Testing Guide," *OWASP Found.*, 2014.

[11] A. Abraham, "Detecting and Exploiting XSS With OWASP Xenotix XSS Exploit Framework v3," 2013.

[12] R. Vibhandik, "Vulnerability assessment of web applications – a testing approach," *Proc. IEEE 2015 Forth International Conference on e-Technologies and Networks for Development (ICeND)*, hlm. 1–6, Sept. 21-23, 2015.

[13] R. C. Joshi dan E. S. Pilli, *Fundamentals of Network Forensics*. London: Springer London, 2016.

[14] M. N. Al-Azhar, *Digital Forensic: A Practical Guide Computer Investigation*. Salemba: Infotek, hlm. 236, 2012.

[15] A. Kurniawan dan I. Riadi, "Detection and analysis cerber ransomware using network forensics behavior based," *Int. J. Netw. Secur.*, vol. 20, no. 5, hlm. 1–8, 2018.

[16] J. He, C. Chang, P. He, dan M. S. Pathan, "Network Forensics Method Based on Evidence Graph and Vulnerability Reasoning," 2016.

[17] P. A. Sonewar dan N. A. Mhetre, "A novel approach for detection of SQL injection and cross site scripting attacks," *Pervasive Comput. (ICPC), 2015 Int. Conf.*, vol. 00, hlm. 1–4, 2015.

[18] S. Fogie, J. Grossman, R. Hansen, A. Rager, dan P. D. Petkov, *XSS Attacks: Cross Site Scripting Exploits and Defense*. 2007.

[19] A. Shrivastava, S. Choudhary, dan A. Kumar, "XSS vulnerability assessment and prevention in web application," *Proceedings on 2016 - 2nd International Conference on Next Generation Computing Technologies, NGCT 2016*, 2017.

[20] W. Melicher, A. Das, M. Sharif, L. Bauer, dan L. Jia, "Riding out DOMsday: Towards Detecting and Preventing DOM Cross-Site Scripting," 2018.

[21] R. Rahim, H. Nurdiyanto, A. Ansari Saleh, D. Abdullah, D. Hartama, dan D. Napitupulu, "Keylogger application to monitoring users activity with exact string matching algorithm," *Journal of Physics: Conference Series*, 2018.

[22] D. P. Shepard dan T. E. Humphreys, "Characterization of receiver response to spoofing attacks," dalam *24th Int. Tech. Meet. Satell. Div. Inst. Navig. 2011, ION GNSS 2011*, 2011.

[23] B. Wikipedians dan R. Creutzburg, "Handbook of Computer Security and Digital Forensics 2016 Part I – Computer Security," April, 2016.

[24] L. Wu, X. Du, dan J. Wu, "Effective defense schemes for phishing attacks on mobile computing platforms," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, hlm. 6678–6691, 2016.

[25] M. Mulazzani, M. Huber, dan E. Weippl, "Social Network Forensics: Tapping the Data Pool of Social Networks," *Eighth Annu. IFIP WG 11.9 Int. Conf. Digit. Forensics*, 2012.
[26] E. Casey, *Digital Evidence and Computer Crime*, Third Ed. Maryland: Elsevier Academic Press, 2011.

Ade Kurniawan is a Doctoral Student at the Graduate School of Information Science and Technology Osaka University. He received a Master's degree in Digital forensics in 2017 from Universitas Islam Indonesia. Currently, he has the Scopus H-Index 2 and acts as a lecturer at the Department of Informatics Engineering of Universitas Universal. His research interests include anomaly detection, deep learning, digital forensics, IoT, machine learning, and network security.