ANALISIS KEAMANAN JARINGAN UNIVERSITAS KRISTEN DUTA WACANA DENGAN SERANGAN SSL/TLS

Nathanael Dharmawan[1], Gani Indriyanta[2], I Kadek Dendy Senapartha [3]
[1,3]*Informatika, Universitas Kristen Duta Wacana*
Jl. Dr. Wahidin Sudirohusodo 5-25, Yogyakarta
nathanel.dharmawan@ti.ukdw.ac.id
ganind@staff.ukdw.ac.id
dendy.prtha@staff.ukdw.ac.id

***Abstract*— The security of data communication over the network has become an obligation that needs to be considered in a technology ecosystem. Data security has various layers, one layer that needs to be protected is the presentation layer where SSL/TLS is located. If at this layer there are vulnerabilities where sensitive data such as cookies, usernames, and passwords are present, then data leakage will have a major impact on all stakeholders in the technology sector using SSL/TLS technology.**

In order to research and improve data security in Duta Wacana Christian University (DWCU) campus network, the researchers conducted SSL/TLS vulnerability testing on the SSAT and E-Class websites using the SSL Test from Qualys and a script from testssl.sh, the author also conducted Checking Mixed Content with GeekFlare and checking HSTS Preload using the HSTS Preload website provided by Google. Researchers also conducted SSL Strip penetration tests at 12 points of the DWCU building and also in Lab D.

Based on the results of the study, there were several results found. The results on the SSL Test using Qualys found that the SSAT and E-Class websites already use HTTP Strict Transport Security (HSTS) rules with Max-Age 31536000 (1 year) but HSTS Preload has not been implemented, Mixed Content testing with GeekFlare shows that all transactions on SSAT and E-Class already uses HTTPS paths, then in tests using the testssl.sh script there are vulnerabilities that are read, and SSL Strip attacks are possible in Duta Wacana Christian University network under several conditions.

Intisari— Keamanan komunikasi data melalui jaringan sudah menjadi kewajiban yang perlu di pertimbangkan dalam sebuah ekosistem teknologi. Keamanan data memiliki berbagai layer, salah satu layer yang perlu dilindungi adalah layer presentasi dimana SSL/TLS berada. Jika pada layer ini terdapat kerentanan dimana data sensitif seperti *cookie*, *username*, dan *password*, maka kebocoran data akan berdampak besar bagi semua pelaku kepentingan di bidang teknologi yang menggunakan teknologi SSL/TLS.

Dalam rangka penelitian dan peningkatan keamanan data di jaringan kampus Universitas Kristen Duta Wacana (UKDW), maka peneliti melakukan pengujian kerentanan SSL/TLS pada situs web SSAT UKDW dan E-Class UKDW menggunakan *Test SSL* dari Qualys dan *script* dari testssl.sh, penulis juga melakukan pengecekan *Mixed Content* dengan GeekFlare serta pengecekan HSTS Preload meenggunakan situs web HSTS Preload yang disediakan Google. Peneliti juga melakukan uji penetrasi *SSL Strip* di 12 titik gedung Universitas Kristen Duta Wacana dan juga di Lab D.

Berdasarkan hasil penelitian, ada beberapa hasil yang ditemukan. Hasil pada *SSL Test* menggunakan Qualys menemukan situs web SSAT dan E-Class sudah menggunakan aturan *HTTP Strict Transport Security* (*HSTS*) dengan *Max-Age* 31536000 (1 tahun) namun HSTS Preload belum di terapkan, pengujian *Mixed Content* dengan GeekFlare menunjukkan bahwa seluruh transaksi pada SSAT dan E-Class sudah menggunakan jalur HTTPS, lalu pada uji menggunakan *script* testssl.sh terdapat kerentanan yang terbaca, serta serangan *SSL Strip* dimungkinkan terjadi di jaringan Universitas Kristen Duta Wacana dengan beberapa kondisi.

***Kata Kunci*— SSL Test, SSL Strip.**

I. PENDAHULUAN

Keamanan komunikasi data melalui jaringan sudah menjadi kewajiban yang perlu di pertimbangkan dalam sebuah ekosistem teknologi. Keamanan data memiliki berbagai layer, salah satu layer yang perlu dilindungi adalah layer presentasi dimana SSL/TLS berada. Jika pada layer ini terdapat kerentanan dimana data sensitif seperti *cookie*, *username*, dan *password*, maka kebocoran data akan berdampak besar bagi semua pelaku kepentingan di bidang teknologi yang menggunakan teknologi SSL/TLS.

Dalam rangka penelitian dan peningkatan keamanan data di jaringan kampus Universitas Kristen Duta Wacana, maka peneliti akan melakukan pengujian kerentanan pada SSL/TLS baik secara pemindaian maupun pengujian penetrasi serangan *SSL Strip*.

Pembatasan pengujian penetrasi serangan *SSL Strip* dilakukan di bawah kepengurusan Pusat Pelayanan Informasi dan Intranet Kampus (PUSPINDIKA) dan Pusat Pelatihan dan Layanan Komputer (PPLK). Lokasi uji penetrasi serangan *SSL Strip* di bawah kepengurusan PUSPINDIKA adalah di 12 titik gedung, sedangkan uji penetrasi serangan *SSL Strip* di bawah kepengurusan PPLK adalah di Lab D. Pada akhirnya penelitian ini bertujuan untuk melihat seberapa rentan jaringan UKDW terhadap kerentanan maupun serangan pada SSL/TLS.

Tujuan dari penelitian ini adalah mendapat informasi dan memastikan apakah jaringan UKDW memiliki kerentanan terhadap pengujian penetrasi *SSL Strip* dan serangan SSL/TLS yang lain dengan *SSL Test*.

Manfaat yang ingin dicapai adalah mengetahui sejauh mana pertahanan jaringan UKDW terhadap pengujian penetrasi *SSL Strip*, dan kerentanan SSL/TLS yang lain, sehingga terjadi peningkatan kesadaran yang akan membawa kepada peningkatan keamanan jaringan UKDW baik secara konfigurasi pada infrastruktur jaringan maupun peningkatan secara kesadaran keamanan dari sumber daya manusia. Serta

sebagai pengetahuan bagi mahasiswa yang akan melakukan penelitian serupa.

Ada 6 metode yang dilakukan pada penelitian ini, diantaranya: Pertama, Persiapan Kebutuhan Sistem: Pada tahap persiapan kebutuhan sistem, peneliti mengumpulkan perangkat keras dan perangkat lunak yang dibutuhkan untuk penelitian. Kedua, Pengujian Sistem Menggunakan Qualys: Pada tahap ini sistem konfigurasi situs web E-Class dan SSAT UKDW di uji menggunakan *SSL Test* dari Qualys. Ketiga, Pengujian Sistem menggunakan testssl.sh: Dalam rangka menemukan kerentanan yang berbeda maka peneliti menguji dengan menggunakan *SSL Test* yang berbeda yaitu testssl.sh. Keempat, Pengujian *Mixed Content* menggunakan GeekFlare: Hal ini dilakukan untuk memastikan bahwa situs web E-Class dan SSAT telah melalui jalur yang aman (https). Kelima, Pengujian HSTS Preload menggunakan HSTS Preload: Untuk mendapatkan informasi tambahan mengenai HSTS Preload maka peneliti melakukan pengecekan pada domain "ukdw.ac.id" beserta subdomain "eclass.ukdw.ac.id" dan "ssat.ukdw.ac.id". Keenam, Penetrasi *SSL Strip*: Dalam rangka mendapatkan analisis yang lebih mendalam, maka peneliti melakukan pengujian penetrasi *SSL Strip* di jaringan Universitas Kristen Duta Wacana.

II. LANDASAN TEORI

A. *ARP Spoof*

ARP Spoof adalah sebuah serangan yang meracuni tabel ARP Korban dengan cara Penyerang akan mengirim pesan bahwa *MAC Address* milik penyerang adalah *MAC gateway*, sehingga nantinya korban memperbarui ARP Tabel nya dan menganggap penyerang adalah *Gateway* [1].

B. *SSL dan TLS*

SSL (*Secure Sockets Layer*) atau TLS (*Transport Layer Security*) adalah sebuah protokol keamanan untuk mengamankan data untuk menjaga kerahasiaan data [1]. Tujuan dari SSL/TLS adalah untuk menjaga *confidentiality* dari serangan penyadap yang dapat melihat isi data, *integrity* untuk melawan dari serangan aktif yang mengubah data, serta *authenticity* yang mengidentifikasi pemilik dari suatu atau beberapa orang dengan menggunakan *certificate*. Pada dasarnya HTTPS dibangun menggunakan teknologi SSL/TLS.

C. *SSL Strip*

SSL Strip adalah serangan yang dibuat oleh Moxie Marlinspike pada tahun 2009 dan telah di presentasikan di acara konferensi BlackHat DC 2009. *SSL stripping* adalah menghilangkan data SSL/TLS dari sebuah *request message* [2]. Sehingga *user* akan mengakses web HTTPS dengan web HTTP.

D. *HSTS*

Untuk menangani *SSL Stripping* maka pada tahun 2012 di buatlah sebuah aturan (*policy*) HSTS atau singkatan dari *HTTP Strict Transport Security* (*HSTS*) digunakan untuk mengatasi serangan *SSL Stripping*. Aturan HSTS ini di deklarasikan di situs web melalui *header respone Strict-Transport-Security* atau bisa diatur pada konfigurasi dari komputer pengguna. Standarisasi mengenai aturan HSTS diatur dalam dokumen RFC 6797.

Alur dari HSTS adalah nantinya jika server ingin berkomunikasi dengan protokol HTTPS maka server akan mengirimkan *HSTS header* ke *browser*. Informasi pada *header* akan di ingat oleh *browser* dengan mengingat domain nya, domain tersebut akan dipaksa untuk menggunakan HTTPS. Apabila suatu saat *user* ingin berkomunikasi menggunakan jalur HTTP maka *browser* akan melakukan konversi dari HTTP ke HTTPS pada *background* nya.

Terdapat 3 bagian penting pada *header* HSTS yang pertama adalah *max-age* yaitu waktu kadaluarsa (harus digunakan), yang kedua adalah *includeSubdomains* yang artinya HSTS akan diterapkan di subdomain yang lain (opsional), yang ketiga adalah *preload*, digunakan untuk mengindikasikan apakah domain telah ditambahkan kedalam *preload list* secara permanen, yang akan di kelola sendiri oleh *browser*.

E. *HSTS Preload*

HSTS Policy bisa diberikan di 2 tempat. Tempat pertama adalah pada *HTTP header* (*directive)* dan tempat kedua adalah di *browser* dengan HSTS Preload.

Jika sebuah domain ditambahkan ke dalam sebuah preload list, maka *browser* akan secara otomatis mendapat aturan HSTS. Sehingga sekalipun *user* baru pertama kali mengunjungi sebuah situs web, maka jalur koneksi yang akan digunakan pertama kali akan langsung dipaksa menjadi HTTPS [3].

F. *Mixed Content*

Mixed Content adalah kondisi dimana sebuah *web page* memiliki *resource* yang *request* nya melalui koneksi yang tidak aman (http) sekaligus juga ada *resource* yang *request* nya melalui koneksi aman (https) [4].

G. *Penggunaan Protokol*

Pada tahun 2019 NIST [5] mengumumkan bahwa server minimal sudah di konfigurasi dengan TLS 1.2 dan TLS 1.3. Penggunaan TLS 1.1 dan TLS 1.0 tidak disarankan. Penggunaan SSL 2.0 dan penggunaan SSL 3.0 tidak boleh digunakan.

H. *Hearthbleed (CVE-2014-0160)*

Hearthbleed adalah sebuah kerentanan yang ditemukan di Pustaka OpenSSL pada 2014 yang memungkinkan penyerang membaca memori pada sistem yang dilindungi menggunakan OpenSSL [6].

I. *BEAST (CVE-2011-3389)*

BEAST (*Browser Exploit Against SSL/TLS*) adalah serangan yang terjadi pada TLS 1.0 yang dikembangkan oleh T.Duong dan J. Riazo [7]. Serangan ini mengambil keuntungan pada *symmetric encryption* dan teknik *cipher block chaining* (*CBC*) untuk menebak *secret key* yang digunakan untuk melakukan enkripsi pada *plaintext*. Jika seorang penyerang dapat menebak sebuah *plaintext box* maka penyerang bisa menebak *symmetric key* dan melakukan pengecekan apakah *chipher text* cocok atau tidak. Serangan BEAST termasuk dalam serangan *brute force* dan bisa ditangani dengan TLS 1.1 dan TLS 1.2

JUTEI Edisi Volume.6 No.2 Oktober 2022
ISSN 2579-3675, e-ISSN 2579-5538
DOI 10.21460/jutei.2022.62.214

J. Secure Renegotiation (RFC 5746)

Pada TLS 1.2 mengizinkan antara *client* dan server untuk melakukan atau memulai *renegotiation* (sebuah *handshake* yang akan membentuk kriptografi yang baru), maka antara *client* dan server tidak memiliki ijin apapun sehingga ada peluang penyerang untuk dapat melakukan interferensi pada lapisan transport *client* yang dapat membuat penyerang melakukan suntikan pada jaringan sebagai interaksi awal antara *client* dan server [8]

K. SWEET32 (CVE-2016-2183, CVE-2016-6329)

Chipers DES dan 3DES biasa digunakan pada TLS, SSH, dan IPSEC memiliki batas ulang tahun sekitar empat miliar blok, hal ini membuat penyerang mampu mendapatkan *cleartext* melalui batas ulang tahun itu terhadap sesi enkripsi yang memiliki durasi panjang seperti sesi HTTPS menggunakan 3DES dalam mode CBC (CVE-2016-2183) [8].

Pada OpenVPN, ketika menggunakan *chiper* blok 64-blok maka akan memudahkan penyerang dari jarak jauh untuk mendapatkan *cleartext* dengan serangan "ulang tahun" terhadap sesi enkripsi yang berdurasi panjang, contoh HTTP-over-OpenVPN (CVE-2016-6329).

Solusi untuk mengatasi serangan Sweet32 adalah dengan menonaktifkan 3-DES serta melakukan pembaharuan pada server yang tidak mendukung *Cipher* yang lebih kuat dari DES dan RC4.

L. FREAK (CVE-2015-0204)

Factoring RSA Export Key (*FREAK*) adalah salah satu serangan yang ditemukan di beberapa *browser* (Safari, Android, Cisco, dan Opera). FREAK diumumkan pada tanggal 3 Maret 2015 oleh peneliti dari INRIA [7]. Negosiasi dalam *export chipper suite* antara *client* dan *server* mengijinkan penyerang untuk menipu *browser* milik client untuk menggunakan *export key* yang lemah. Serangan FREAK melakukan *downgrades chipper suite* yang menggunakan pertukaran kunci RSA kurang dari 512 bit. Hal ini bisa diatasi dengan mematikan *export cipher suite* pada *browser*.

M. DROWN (CVE-2016-0800, CVE-2016-0703)

Decrypting RSA with Obsolete and Weakened Encryption (DROWN) yaitu serangan yang memungkinkan penyerang untuk melakukan dekripsi *chipertext* TLS dengan menggunakan *oracle padding* RSA pada protokol SSLv2. Solusinya adalah dengan tidak menggunakan SSLv2 [7].

N. LOGJAM (CVE-2015-4000)

Pada protokol TLS1.2 dan kebawah, memungkinkan penyerang melakukan *MITM* ketika *ciphersuite DHE_EXPORT* diaktifkan pada server tetapi tidak di *client* [9]. Kelemahan ini membuat penyerang dapat melakukan *cipher-downgrade*. Solusi untuk Logjam adalah dengan menggunakan kunci minimal 2048 bit, mematikan *export cipher suite*, serta melakukan pembaruan pada versi OpenSSH yang menggunakan ECDH (*Elliptic-Curve Diffie-Hellman*) untuk pertukaran kunci.

O. LUCKY13 (CVE-2013-0169)

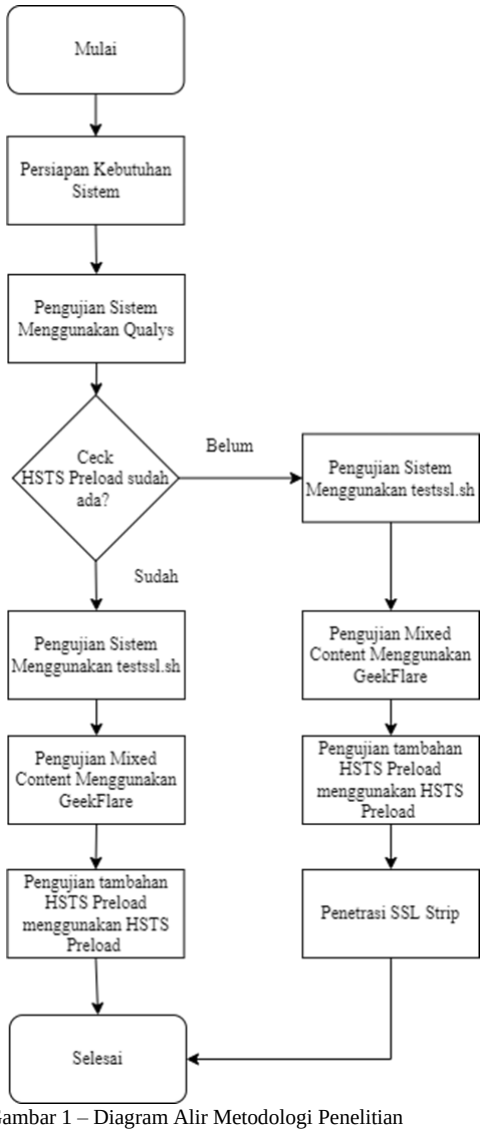
Penyebab dari permasalahan ini adalah pada *padding*, yang digunakan pada CBC Mode yang tidak dilindungi oleh mekanisme validasi integritas milik TLS [10]. Nantinya penyerang dapat melakukan modifikasi pada *padding* yang lewat dan mengamati bagaimana server berperilaku. Jika penyerang dapat mendeteksi bagaimana server bereaksi terhadap *padding* yang sudah di modifikasi, maka informasi akan terbongkar dan *plaintext* akan ditemukan. Solusinya adalah dengan tidak menggunakan CBC Suites dan melakukan implementasi AEAD Cipher Suites.

P. RC4 (CVE-2013-2566, CVE-2015-2808)

RC4 adalah sebuah *Cipher* yang di desain oleh Ron Rivest pada tahun 1987. RC4 digunakan untuk enkripsi pada protokol WEP, WPA-TKIP, SSL/TLS, PPP/MPPE, dll. Pada saat ini algoritma RC4 sudah rusak / hancur [10].

III. METODOLOGI PENELITIAN

A. Diagram Alir Metodologi Penelitian



Gambar 1 – Diagram Alir Metodologi Penelitian

B. Persiapan Kebutuhan Sistem

1) Perangkat Keras yang Digunakan:

- 1 buah laptop digunakan sebagai penyerang uji penetrasi *SSL Strip* serta uji *SSL Test*.

- 1 buah Ponsel Pintar digunakan sebagai korban untuk uji penetrasi *SSL Strip*.

2) Perangkat Lunak yang Digunakan:

- OS : Kali Linux v.2022.3 (Laptop Penyerang).
- Bettercap v3.2 & 1.6.2 (Laptop Penyerang)
- Wireshark (Laptop Penyerang)
- *SSL Test*: Qualys (Aplikasi berbasis web) dan Testssl.sh 3.1 Dev (Terminal).
- *Browser* (Ponsel Pintar Korban)
- Net Analyzer (Ponsel Pintar)

C. Pengujian Sistem Menggunakan Qualys

Pengujian menggunakan *SSL Test* dari Qualys difokuskan untuk mendapatkan hasil parameter Rating, Protokol, HSTS, Max Age, HSTS Preload. Berikut langkah pengujian dengan SSL Test dari Qualys:

1) Membuka situs https://www.ssllabs.com/
2) Memilih menu "Test Your Server"
3) Memasukkan domain "eclass.ukdw.ac.id" atau "ssat.kdw.ac.id" secara bergiliran. Dan centang "Do not show the results on the boards" supaya hasil tidak tertampil pada board. lalu "submit".
4) Lalu langkah selanjutnya adalah menunggu hasil.

D. Pengujian Sistem Menggunakan testssl.sh

Pengujian menggunakan teslssl.sh digunakan untuk mendapatkan hasil parameter kerentanan Hearthbleed, BEAST, Secure Renegotiation, SWEET32, FREAK, DROWN, LUCKY13, dan RC4. Berikut langkah pengujian dengan SSL Test dari testssl.sh:

1) Pilih salah satu direktori untuk dijadikan tempat menyimpan script testssl.sh.
2) Mengunduh perangkat lunak script testssl.sh
3) Setelah selesai masuk ke direktori testssl.sh.
4) Menggunakan perintah secara bergiliran atau membuka tab terminal yang baru dan memasukkan:
"bash testssl.sh ssat.ukdw.ac.id" untuk SSAT dan "bash testssl.sh eclass.ukdw.ac.id" untuk E-Class.
5) Setelah menunggu beberapa saat, hasil nya akan keluar.

E. Pengujian Mixed Content Menggunakan GeekFlare

Pengujian *Mixed Content* digunakan untuk melihat apakah sebuah situs web sudah melakukan transaksi data dengan menggunakan protokol HTTPS atau justru HTTP maupun justru HTTPS dan HTTP. Langkah untuk melakukan uji Mixed Content adalah:

1) Mengunjungi situs web uji *Mixed Content* milik GeekFlare pada: https://geekflare.com/tools/mixed-content-test.
2) Memasukkan subdomain "ssat.ukdw.ac.id" dan "eclass.ukdw.ac.id" secara bergantian.
3) Hasil keluar setelah beberapa saat.

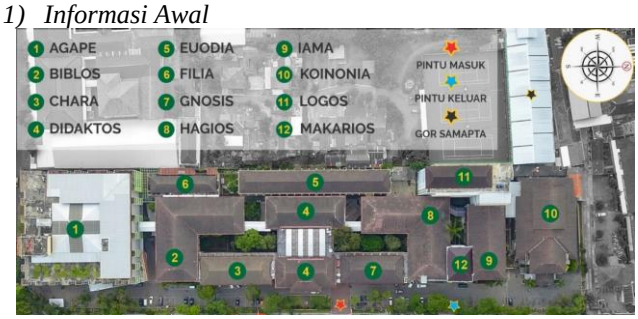
F. Pengujian HSTS Preload Menggunakan HSTS Preload

Penggujian HSTS Preload menggunakan situs web yang disediakan oleh google untuk melakukan pengecekan pada HSTS Preload adalah untuk mendapat informasi tambahan mengenai konfigurasi HSTS Preload. Langkah yang digunakan untuk melakukan uji HSTS Preload adalah:

1) Mengunjungi website uji HSTS Preload yang disediakan Google pada: https://hstspreload.org/
2) Memasukkan subdomain "ssat.ukdw.ac.id" dan "eclass.ukdw.ac.id" secara bergantian.
3) Hasil keluar setelah beberapa saat.

G. Penetrasi SSL Strip

1) Informasi Awal



Gambar 2 – Denah Kampus Universitas Kristen Duta Wacana (https://oka.ukdw.ac.id/peta)

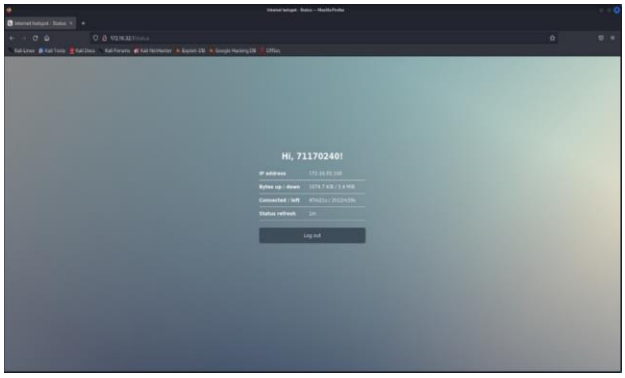
2) Pengujian SSL Strip Di Lab D

Tabel 1 – Perintah dan Metode Menggunakan Bettercap v.2.32

Perintah dan Metode (Bettercap v.2.32)	
net.probe on	Digunakan untuk memeriksa *hosts* yang hidup di jaringan.
set http.proxy.sslstrip true	Digunakan untuk menghidupkan proxy *SSL Stripping.*
set net.sniff verbose false	Digunakan untuk menampilkan setiap paket yang sudah ditangkap dan diuraikan, hanya paket yang berada di layer aplikasi yang akan ditampilkan.
set arp.spoof.targets [IP/MAC/Aliases]	Digunakan untuk menetapkan target yang akan di ditipu, bisa menggunakan IP, range IP, MAC Address atau *aliases*.
set arp.spoof.fullduplex true	Digunakan untuk menyerang target dan gateway (Jika perlindungan ARP ada dan ditaruh di router maka serangan ini akan gagal).
arp.spoof on	Menghidupkan penipu ARP.
http.proxy on	Menghidupkan proxy untuk HTTP
net.sniff on	Menghidupkan pengendusan, data dari Korban akan ditampilkan di layar Penyerang.

3) Pengujian SSL Strip di Beberapa Gedung UKDW

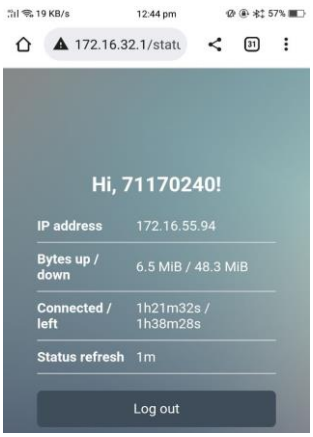
a. Penyerang melakukan login pada hotspot ukdw.



Gambar 3 – Tampilan *Login Captive Portal* (Penyerang)

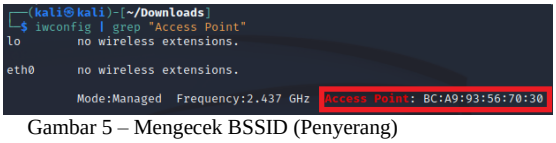
JUTEI Edisi Volume.6 No.2 Oktober 2022
ISSN 2579-3675, e-ISSN 2579-5538
DOI 10.21460/jutei.2022.62.214

b. *Korban melakukan login pada hotspot ukdw*

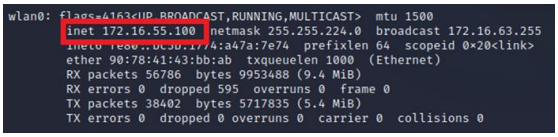


Gambar 4 – Tampilan *Login Captive Portal* (Korban)

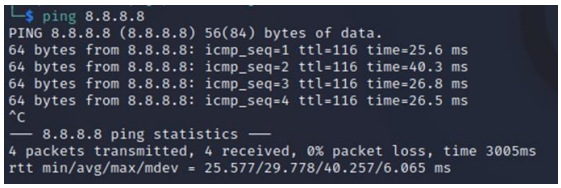
c. *Tes koneksi BSSID, Ping, Ifconfig pada laptop penyerang*



Gambar 5 – Mengecek BSSID (Penyerang)

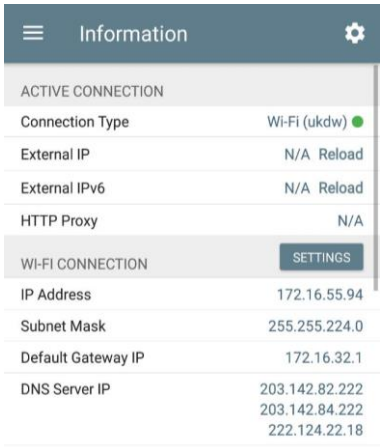


Gambar 6 – Mengecek IP (Penyerang)

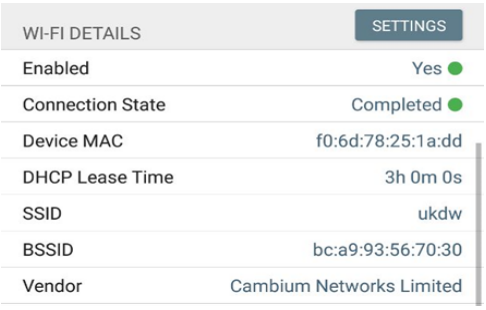


Gambar 7 – Tes Koneksi (Penyerang)

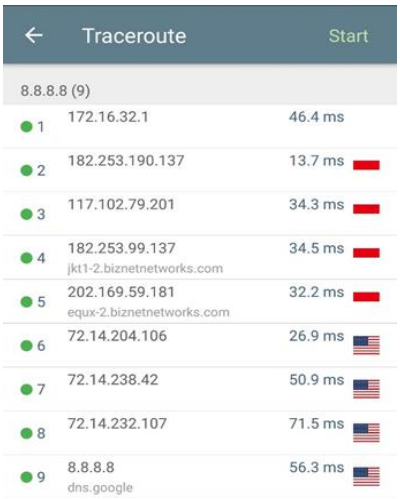
d. *Tes koneksi BSSID, Ping, Ifconfig, traceroute pada ponsel pintar korban.*



Gambar 8 – Informasi Korban



Gambar 9 – Mengecek BSSID (Korban)



Gambar 10 – Traceroute (Korban)

e. *Menghidupkan ARP Spoof, DNS Spoof, Net Sniff, SSL Strip, HTTP Proxy menuju target.*

Gambar 11 – Serangan Diaktifkan (Penyerang)

Pada gambar 11 adalah penyerang menggunakan bettercap 1.6.2 dikarenakan isu yang terjadi pada Bettercap 2.32 yang mengalami masalah pada modul *SSL Strip* [11]. Perintah yang digunakan 'sudo bettercap -T 172.16.55.94 –proxy -P POST'. Perintah tersebut bermaksud untuk:

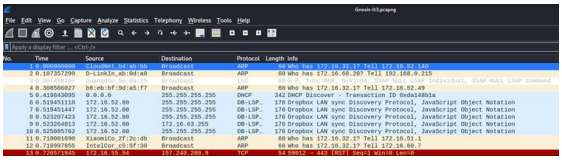
- Sudo bettercap : Menghidupkan alat jaringan bernama bettercap dengan menggunakan akun admin.
- -T 172.16.55.94 : Memasang target korban dengan IP 172.16.55.94.
- --proxy : Proxy http dan *SSL strip* diaktifkan.
- -P POST : Dimana bettercap akan melakukan pengecekan hanya pada data POST.

Terlihat bahwa modul *spoofing*, *sniffer*, *http-proxy*, *sslstrip*, dan *dns server* telah aktif.

f. Cek Traceroute pada ponsel pintar korban apakah aliran data beralih menuju penyerang

Gambar 12 – Traceroute Setelah Serangan Diaktifkan (Korban)

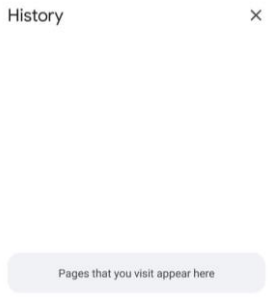
g. Meng-aktifkan Wireshark



Gambar 13 – Wireshark Diaktifkan

Pada gambar 13 menunjukkan wireshark telah berjalan dan digunakan untuk menangkap data dari korban. Penggunaan wireshark digunakan dikarenakan *sniffer* pada bettercap bermasalah setelah peneliti melakukan *upgrade OS* pada Kali Linux.

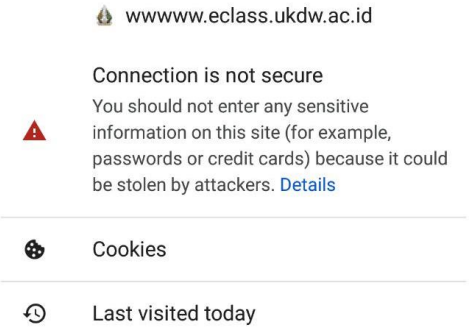
h. Membersihkan Cache pada ponsel pintar korban



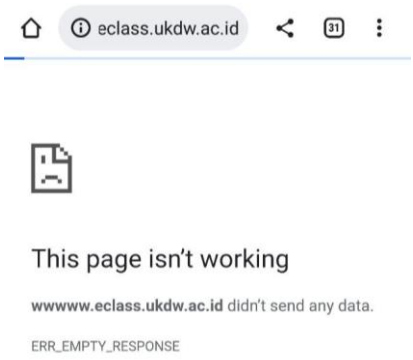
Gambar 14 – Penghapusan *Cache* di *Browser* (Korban)

i. Ases menuju E-Class UKDW

Gambar 15 – Tampilan E-Class Setelah SSL Strip Berhasil Dilakukan



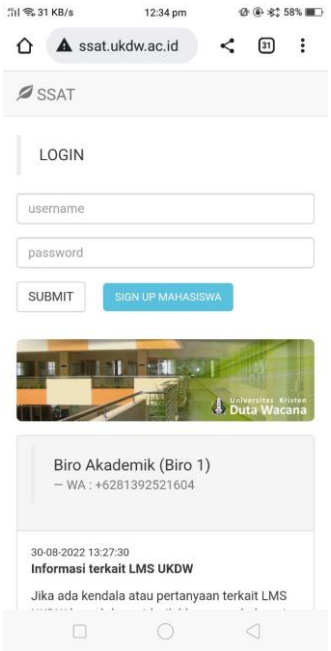
Gambar 16 – Domain dan Koneksi E-Class Setelah Serangan SSL Strip



Gambar 17 – Tampilan Respon Website E-Class Setelah Korban Melakukan Login

Gambar 17 menunjukkan respon situs web ketika korban melakukan *request login* menuju web server E-Class, dalam hal ini bettercap tidak dapat melayani *request*.

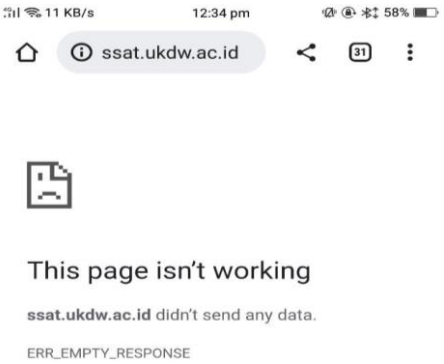
j. Akses menuju SSAT UKDW



Gambar 18 – Tampilan SSAT Setelah SSL Strip Berhasil Dilakukan

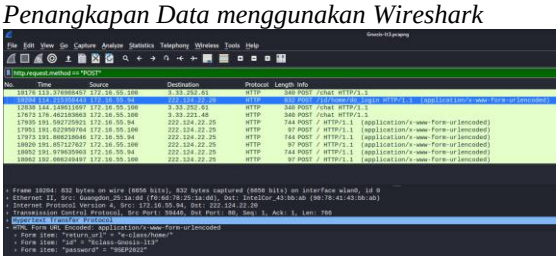
JUTEI Edisi Volume.6 No.2 Oktober 2022
ISSN 2579-3675, e-ISSN 2579-5538
DOI 10.21460/jutei.2022.62.214

Gambar 19 – Domain dan Koneksi SSAT Setelah Serangan SSL Strip



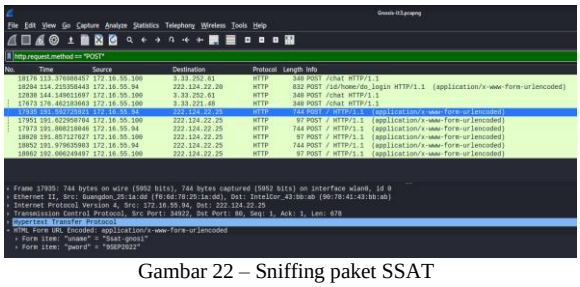
Gambar 20 – Tampilan Respon Situs Web SSAT Setelah Korban Melakukan Login

k. Penangkapan Data menggunakan Wireshark



Gambar 21 – Sniffing paket E-Class

Gambar 21 adalah penangkapan data menggunakan wireshark dengan filter "*http.request.method == "POST"*". Ditemukan IP korban 172.16.55.94 mengirimkan paket POST menuju 222.124.22.20 yang adalah server E-Class berada. Data yang didapat adalah ID "Eclass-Gnosis-lt3" dan data password yang adalah "9SEP2022".



Gambar 22 – Sniffing paket SSAT

Gambar 22 adalah penangkapan data menggunakan wireshark. Ditemukan IP korban 172.16.55.94 mengirimkan paket POST menuju 222.124.22.25 yang adalah server SSAT berada. Data yang didapat adalah uname "Ssat-gnosi" dan data pword yang adalah "9SEP2022".

IV. HASIL DAN ANILIS

A. Hasil Dan Analisis Pengujian Sistem Menggunakan Qualys

Tabel 2 – Perbandingan Sebelum Dan Sesudah Pembaharuan Pada E-Class

No	E - Class		
	Variabel	Sebelum Pembaharuan (05 April 2022)	Sesudah Pembaharuan (17 Mei 2022)
1	Rating	B	A+
2	Protokol	TLS 1.1, TLS 1.2	TLS 1.2
3	HSTS	Tidak Ada	Ada
4	Max-Age	0	31536000
5	HSTS Preload	Tidak Ada	Tidak Ada

Tabel 3 – Perbandingan Sebelum Dan Sesudah Pembaharuan Pada SSAT

No	SSAT		
	Variabel	Sebelum Pembaharuan (05 April 2022)	Sesudah Pembaharuan (17 Mei 2022)
1	Rating	B	A+
2	Protokol	TLS 1.0, TLS 1.1, TLS 1.2	TLS 1.2
3	HSTS	Tidak Ada	Ada
4	MaxAge	0	31536000
5	HSTS Preload	Tidak Ada	Tidak Ada

B. Hasil Dan Analisis Pengujian Sistem Menggunakan testssl.sh

Tabel 4 – Hasil testssl.sh

No	Testssl.sh (02 Agustus 2022)		
	Kerentanan	E-Class	SSAT
1	Hearthbleed	Tidak Rentan	Tidak Rentan
2	BEAST	Tidak Rentan	Tidak Rentan
3	Secure Renegotiation	Telah Didukung	Telah Didukung
4	SWEET32	Tidak Rentan	Rentan
5	FREAK	Tidak Rentan	Tidak Rentan
6	DROWN	Tidak Rentan	Tidak Rentan
7	LUCKY13	Berpotensi Rentan	Berpotensi Rentan
8	RC4	Tidak Rentan	Rentan

C. Hasil Dan Analisis Pengujian Mixed Content Menggunakan GeekFlare

Hasil pengujian Mixed Content menggunakan GeekFlare pada tanggal 18 Agustus 2022 menunjukkan seluruh aliran data pada situs E-Class dan SSAT telah

menggunakan HTTPS dan tidak ada *resources* yang melewati jalur selain HTTPS.

D. Hasil Dan Analisis Pengujian HSTS Preload Menggunakan HSTS Preload

Hasil pengujian HSTS Preload untuk domain ukdw.ac.id pada tanggal 03 Agustus 2022 adalah yang pertama tidak ada perintah *includeSubDomain directive* pada *server* dimana seharusnya sudah ada. Apabila ingin menggunakan *includeSubDomain directive* maka perlu dipastikan subdomain yang lain juga harus sudah mendukung HTTPS, apabila tidak dikawatirkan akan membuat *error* pada subdomain yang lain, lalu hasil yang lain adalah belum ada HSTS Preload.

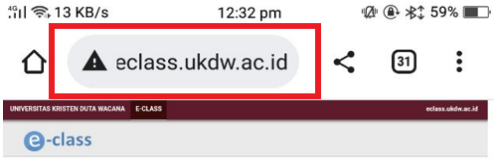
Hasil pengujian HSTS Preload untuk subdomain eclass.ukdw.ac.id dan ssat.ukdw.ac.id pada tanggal 03 Agustsus 2022 adalah seharusnya *HSTS Directive* tidak diletakkan pada subdomain eclass.ukdw.ac.id dan ssat.ukdw.ac.id, belum ada HSTS Preload, lalu seharusnya HSTS Preload sudah ada pada E-Class dan SSAT.

E. Hasil dan Analisis Penetrasi SSL Strip

Tabel 5 – Hasil dan Analisis Penetrasi SSL Strip

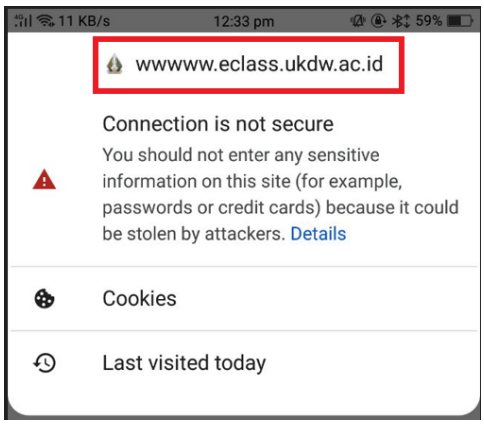
No	Situs	Lokasi	ARP Spoof	SSL Strip
1	E-Class	Lab D	✓	×
	SSAT		✓	✓
2	E-Class	Agape Lt.1	✓	✓
	SSAT		✓	✓
3	E-Class	Biblos Lt.1	×	×
	SSAT		×	×
4	E-Class	Chara Lt.1	×	×
	SSAT		×	×
5	E-Class	Didaktos Lt.2	✓	✓
	SSAT		✓	✓
6	E-Class	Eudia Lt.1	Koneksi Bermasalah	
	SSAT			
7	E-Class	Filia Lt.1	✓	✓
	SSAT		✓	✓
8	E-Class	Gnosis Lt.3	✓	✓
	SSAT		✓	✓
9	E-Class	Hagios Lt.1	×	×
	SSAT		×	×
10	E-Class	Iama Lt.3	×	×
	SSAT		×	×
11	E-Class	Koinonia Lt.1	×	×
	SSAT		×	×
12	E-Class	Logos Lt.3	×	×
	SSAT		×	×
13	E-Class	Makarios	×	×
	SSAT		×	×

F. Hasil Dan Analisis Tampilan URI Korban SSL Strip



Gambar 23 - Tampilan URI Korban SSL Strip – 1

Gambar 23 adalah tampilan *URI* korban yang mengalami *SSL Strip* pada platform *mobile*. Tampilan pada gambar 28 memiliki arti dimana *URI* yang tertampil pada platform mobile sangat terbatas, sehingga tidak bisa menampilkan URI secara penuh, serta memiliki logo "dangerous" yang artinya situs di akses dengan non HTTPS.



Gambar 24 – Tampilan UI Korban SSL Strip – 2

Apabila pada logo "dangerous" di klik maka akan terlihat bahwa *URI* telah berubah menjadi "wwwww.eclass.ukdw.ac.id".

V. KESIMPULAN

Pada Situs web E-class sudah ditambahkan HSTS dengan nilai *Max-Age* 31536000 (1 tahun), dan HSTS Preload belum ditambahkan.

Pada situs web E-Class ditemukan potensi kerentanan terhadap serangan LUCKY13.

Setelah dilakukan pembaruan pada situs SSAT maka protokol yang digunakan pun hanya mengizinkan protokol TLS 1.2 (sudah sesuai standar NIST) dan mendapat nilai A+. Aturan HSTS sudah ditambahkan pada pembaharuan dengan nilai *Max-Age* 31536000 (1 tahun), dan HSTS Preload belum ditambahkan.

Pada situs web SSAT ditemukan kerentanan serangan LUCKY 13 yang "berpotensi Rentan" lalu pada SWEET32 dan RC4 adalah "Rentan".

Belum ada nya HSTS Preload memungkinkan penyerang untuk melakukan *SSL Stripping* diawal koneksi dimana *user* belum mendapat aturan *HSTS directive*.

Tidak ada konten yang menggunakan jalur tanpa enkripsi (HTTP). Pada kasus kali ini E-Class dan SSAT sudah menggunakan jalur HTTPS. Dimungkinkan dengan adanya implementasi HSTS memungkinkan untuk memaksa seluruh *resource* di akses secara terenkripsi.

Penempatan *HSTS Directive* masih dilakukan pada domain dan subdomain dimana seharusnya dilakukan pada domain dan bukan ditambahkan juga pada subdomain.

JUTEI Edisi Volume.6 No.2 Oktober 2022
ISSN 2579-3675, e-ISSN 2579-5538
DOI 10.21460/jutei.2022.62.214

Penempatan *HSTS Directive* pada subdomain diduga karena memperhitungkan kesiapan insfrastruktur.

Penetrasi *SSL Strip* di Universitas Kristen Duta Wacana masih dimungkinkan dengan beberapa kondisi. *Cache* SSAT dan E-Class pada korban harus tidak ada, terkoneksi dengan WAP yang sama, WAP tidak memiliki fitur *antispoofing*, dan *user* harus mengetik URI (eclass.ukdw.ac.id dan ssat.ukdw.ac.id).

Waktu tunggu sebuah *request* untuk melakukan *load* pada halaman web yang terkena *ssl strip* relatif lebih lama dibanding koneksi normal. Pada E-Class waktu tunggu berkisar 10-45 detik, dan SSAT berkisar 2-10 detik. Dimana pada koneksi normal *load* pada kedua situs web tidak mencapai 3 detik.

Penetrasi *SSL Strip* di Universitas Kristen Duta Wacana akan gagal apabila, *User* memiliki *cache* SSAT dan E-Class, *User* terkoneksi dengan WAP yang berbeda, *User* terkoneksi dengan WAP yang sudah memiliki atau mengaktifkan *antispoofing*, serta *User* yang tidak menulis URI (eclass.ukdw.ac.id dan ssat.ukdw.ac.id) tidak akan mendapat koneksi ketika *ssl strip* berjalan, maka *user* akan menyadari bahwa *user* tidak mendapat internet dan bisa saja *User* memilih tidak melanjutkan proses menuju situs web dan mengganti jaringannya.

UCAPAN TERIMA KASIH

Peneliti berterima kasih kepada Tuhan yang memberi Kasih Karunia dan Kebenaran dalam penulis melalui setiap masa dalam hidup, Keluarga tercinta: yang dimana Tuhan menaruh, mempercayakan untuk bertumbuh dan menjadi berkat serta saling memberi dukungan bersama, Bapak Restyandito S.Kom, MSIS., Ph.D selaku Dekan FTI, yang senantiasa memberi contoh serta mengusahakan yang terbaik bagi Mahasiswa/i nya, Ibu Gloria Virginia S.Kom., MAI., Ph.D selaku Kaprodi Informatika, yang telah mengupayakan yang terbaik untuk prodi Informatika, Bapak Ir. Gani Indriyanta, MT selaku Dosen Pembimbing 1, yang telah memberikan arahan, motivasi, ilmunya dan dengan penuh kesabaran membimbing penulis, Bapak I Kadek Dendy Senapartha. S. T., M. Eng selaku Dosen Pembimbing 2, yang telah memberikan ilmu, waktu dan kesabaran dalam membimbing penulis, Bapak Willy Sudiarto Raharjo, S.Kom., M.Cs yang telah bersedia meluangkan waktu untuk penulis untuk memberi masukan pada penelitian penulis, Teman teman terkasih, Yulius, Riel, Paul, mahasiswa/i praktikum Jarkom dan Inlan, dan teman-teman perkuliahan yang telah memberi dukungan moril serta menjadi tempat bertumbuh bersama dalam pengetahuan maupun karakter, Keluarga PPLK (Pak Abet, Pak Arif, Pak Agung, Kak Richard) yang telah memberi dukungan tempat, *snack* dan fasilitas bagi penulis untuk melakukan penelitian, Pihak PUSPINDIKA yang telah memberikan ijin bagi penulis untuk melakukan penelitian jaringan naungan PUSPINDIKA.

VI. DAFTAR PUSTAKA

[1] M. S. Hossain, A. Paul and M. H. Islam, "Survey of the Protection Mechanisms to the SSL-based Session Hijacking Attacks," *Network Protocols and Algorithms*, pp. 83-108, 2018.

[2] K. V. K and A. R. K. P, "Taxonomy of SSL/TLS Attacks," *I. J. Computer Network and Information Security*, pp. 15-24, 2016.

[3] X. Li, C. Wu, S. Ji, Q. Gu and R. Beyah, "HSTS Measurement and an Enhanced Stripping Attack Against HTTPS," *SecureComm*, pp. 489-509, 2017.

[4] "Check Mixed Content (HTTP)," GeekFlare, [Online]. Available: https://geekflare.com/tools/mixed-content-test. [Accessed 7 November 2022].

[5] K. A. McKay and D. A. Cooper, "Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations," *NIST Special Publication 800-52 Revision 2*, 2019.

[6] "Hearthbleed," Hearthbleed, [Online]. Available: https://heartbleed.com/. [Accessed 7 November 2022].

[7] W. S. Raharjo and A. A. Bajuadji, "Analisa Implementasi Protokol HTTPS pada Situs Web Perguruan Tinggi di Pulau Jawa," *ULTIMACS*, pp. 102-111, 2016.

[8] E. Rescorla, M. Ray, S. Dispensa and N. Oskov, "Transport Layer Security (TLS) Renegotiation Indication Extension," February 2010. [Online]. Available: https://www.rfc-editor.org/rfc/rfc5746.

[9] "CVE," [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-2183.

[10] "CVE," [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-4000.

[11] I. Ristić, Bulletproof SSL and TLS, London: Feisty Duck Limited, 2015.

[12] "Github," [Online]. Available: https://github.com/bettercap/bettercap/issues/154.

JUTEI Edisi Volume.6 No.2 Oktober 2022
ISSN 2579-3675, e-ISSN 2579-5538