PENGUJIAN AUTENTIKASI DAN OTORISASI WEB MI-GATEWAY UII BERDASARKAN DOKUMEN OWASP WSTG v4.2

Disusun Oleh:

N a m a : Aditya Wibisono Kuncoro
NIM : 18523217

PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
2022

HALAMAN PENGESAHAN DOSEN PEMBIMBING

PENGUJIAN AUTENTIKASI DAN OTORISASI WEB MI-GATEWAY UII BERDASARKAN DOKUMEN OWASP WSTG v4.2

TUGAS AKHIR

Disusun Oleh:

N a m a : Aditya Wibisono Kuncoro

NIM : 18523217

Yogyakarta, 29 Agustus 2022

Pembimbing,

(Fayruz Rahma, S.T., M.Eng.)

HALAMAN PENGESAHAN DOSEN PENGUJI

PENGUJIAN AUTENTIKASI DAN OTORISASI WEB MI-GATEWAY UII BERDASARKAN DOKUMEN OWASP WSTG v4.2

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 29 Agustus 2022

Tim Penguji

Fayruz Rahma, S.T., M.Eng. _____

Anggota 1

Galang Prihadi Mahardhika, S.Kom., M.Kom. _____

Anggota 2

Moh. Idris, S.Kom., M.Kom. _____

Mengetahui,
Ketua Program Studi Informatika – Program Sarjana
Fakultas Teknologi Industri
Universitas Islam Indonesia

(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Aditya Wibisono Kuncoro

NIM : 18523217

Tugas akhir dengan judul:

PENGUJIAN AUTENTIKASI DAN OTORISASI WEB MI-GATEWAY UII BERDASARKAN DOKUMEN OWASP WSTG v4.2

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Klaten, 29 Agustus 2022

(Aditya Wibisono Kuncoro)

HALAMAN PERSEMBAHAN

Alhamdulillahi Robbil ‘Alamin. Segala puji dan syukur ke hadirat Allah Subhana Wa Ta’ala yang telah memberikan rahmat, ridho, dan karunia-Nya serta nikmat yang tiada tara kepada saya. Shalawat serta salam kepada Nabi Muhammad Shallallahu ‘Alaihi Wasallam, sebagai pembawa risalah Allah terakhir dan penyempurna seluruh risalah-Nya yang telah membawa umatnya dari zaman yang gelap gulita ke zaman yang terang benderang. Tugas akhir ini kupersembahkan untuk semua orang yang aku cintai. Terutama kepada Mama tersayang yang tidak pernah lelah memberikan kasih sayang, bimbingan akhlak, dan doa dari kecil hingga sekarang. Kepada Ayah tercinta yang selalu memberikan kasih sayang, nasihat menghadapi kehidupan, pentingnya kerja keras dan doa. Kepada Ibu pembimbingku yang selalu memberikan semangat dan motivasi menyelesaikan tugas akhir dengan sepenuh hati. Kepada sahabat-sahabatku, terima kasih atas segala kebersamaan, bantuan, dukungan, pengalaman, nasihat, dan doa yang telah diberikan.

HALAMAN MOTO

“Jalan awal terbaik untuk mewujudkan segala impian Anda adalah bangun dan bangkit dari tempat tidur.”
(Paul Valéry)

“Diamku adalah sebuah perjalanan, untuk membuatmu diam.”
(Danar Widiyanto)

“Belum terlambat untuk menjadi apa pun yang kita inginkan”.
(George Eliot)

“Semuanya tidak terlihat mungkin sampai semuanya selesai”.
(Nelson Mandela)

“Segala sesuatu yang bisa Kau bayangkan adalah nyata”.
(Pablo Picasso)

“Hidup ini seperti sepeda. Agar tetap seimbang, Kau harus terus bergerak”.
(Albert Einstein)

“Satu-satunya hal yang harus Kita takuti adalah ketakutan itu sendiri”.
(Franklin D. Roosevelt)

“Hidup ini terlalu misterius untuk Kau yang jalani dengan terlalu serius”.
(Mary Engelbreit)

“Saat kita memperbaiki hubungan dengan Allah, niscaya Allah akan memperbaiki segala sesuatunya untuk Kita”.
(Dr. Bilal Philips)

“Kebahagiaan dirasakan oleh orang-orang yang bisa merasa puas pada dirinya”.
(Aristoteles)

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Dengan mengucap Alhamdulillah, puji dan syukur ke hadirat Allah Subhana Wa Ta'ala yang telah memberikan berkat rahmat dan hidayah-Nya, sehingga tugas akhir yang berjudul "**Pengujian Autentikasi dan Otorisasi Web Mi-Gateway UII Berdasarkan Dokumen OWASP WSTG v4.2**" dapat diselesaikan dengan baik. Shalawat serta salam tidak lupa senantiasa dilimpahkan kepada Nabi Muhammad Shallallahu 'Alaihi Wasallam, yang telah membawa kita dari zaman jahiliyah menuju ke zaman terang benderang.

Laporan tugas akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata-1 (S1) di Program Studi Informatika – Program Sarjana, Fakultas Teknologi Industri, Universitas Islam Indonesia. Selain itu, tugas akhir ini juga sebagai sarana untuk menerapkan ilmu dan teori yang telah didapatkan selama menjalani masa studi di Program Studi Informatika – Program Sarjana, Fakultas Teknologi Industri, Universitas Islam Indonesia.

Akhirnya, dengan segala kerendahan hati izinkanlah penulis untuk menyampaikan rasa terima kasih dan penghargaan yang setinggi-tingginya atas motivasi, bantuan, bimbingan, dan doa. Penulis menyampaikan rasa dan penghargaan tersebut kepada:

1. Kedua orang tua saya yang tidak pernah berhenti memberikan do'a, dukungan dan motivasi sehingga penulis dapat menyelesaikan laporan tugas akhir.
2. Bapak Hendrik, S.T., M.Eng, selaku Ketua Jurusan Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.
3. Bapak R. Teduh Dirgahayu, S.T., M.Sc., Ph.D, selaku Ketua Program Studi Informatika – Program Sarjana, Fakultas Teknologi Industri, Universitas Islam Indonesia.
4. Ibu Fayruz Rahma S.T., M.Eng, selaku Dosen Pembimbing 1 tugas akhir yang telah memberikan masukan, arahan, serta dorongan sehingga penelitian ini dapat terlaksana sehingga tugas akhir ini dapat diselesaikan.
5. Bapak Moh. Idris, S.Kom., M.Kom dan Bapak Galang Prihadi Mahardhika, S.Kom., M.Kom, selaku Dosen Penguji 1 dan Dosen Penguji 2 yang telah memberi masukan, arahan, serta dorongan sehingga tugas akhir ini dapat diselesaikan.
6. Mutiara Santi Hidayah yang tidak pernah bosan memberikan motivasi dan dukungan kepada penulis dalam setiap proses mengerjakan tugas akhir ini.

7. Sahabat-sahabat terbaik saya Ananda, Daffa, Diko, Damastri, Haris, Fafa, Yanu dan Tiara yang selalu berusaha menghibur dan men-*support*.
8. Teman-teman Informatika angkatan 2018 terima kasih atas pengalaman kuliah yang tidak terlupakan.
9. Kepada semua pihak yang telah membantu baik secara langsung maupun tidak langsung, semoga Allah Swt. menjadikannya amal baik yang senantiasa mendapatkan balasan dan kebaikan berlipat ganda.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Yogyakarta, 29 Agustus 2022

(Aditya Wibisono Kuncoro)

SARI

Pesatnya perkembangan teknologi memengaruhi cara individu, organisasi, dan pelaku bisnis dalam melakukan proses distribusi informasi. Informasi telah menjadi aset yang tak ternilai di era yang terus berkembang ini. Universitas Islam Indonesia menggunakan kemajuan dari teknologi, khususnya kemajuan internet, untuk memberikan kemudahan melalui sistem aplikasi web kepada *civitas* akademika maupun pihak luar guna memudahkan dalam penyampaian informasi. Web yang dimiliki oleh Program Studi Informatika Program Magister Universitas Islam Indonesia yaitu MI-Gateway UII. Web MI-Gateway UII menjadi *platform* utama bagi calon mahasiswa, mahasiswa dan dosen Program Magister Informatika. MI-Gateway memiliki beberapa fitur mulai dari pendaftaran mahasiswa baru, penilaian dosen, hingga pendaftaran tesis.

Keamanan adalah salah satu aspek terpenting dari segala hal. Selain kemajuan teknologi, pentingnya keamanan jaringan untuk mencegah serangan dari pihak luar yang tidak bertanggung jawab yang dapat memengaruhi proses bisnis yang sedang berjalan. *Penetration Testing (pentest)* diperlukan untuk mengetahui seberapa rentan jaringan web terhadap serangan pihak luar. Dalam pengujian ini, penguji menyimulasikan dirinya sebagai pihak luar yang mencoba masuk ke jaringan web.

Pengujian ini menggunakan metode *OWASP WSTG v4.2* tahun 2020. Metode ini dipilih karena fokus dari metode tersebut yaitu spesifik terhadap kerentanan sistem berdasarkan inspeksi pada kode program dan pengujian penetrasi. Pertimbangan lainnya dalam memilih metode tersebut karena metode tersebut sangat cocok digunakan dalam pengujian keamanan aplikasi web berdasarkan poin-poin pengujian yang cukup banyak dan berkaitan dengan keamanan web.

Selama penelitian berlangsung ditemukan hasil kemungkinan celah keamanan pada beberapa pengujian, meliputi: *Testing for Weak Lock Out Mechanism (WSTG-ATHN-03), Test Remember Password Functionality (WSTG-ATHN-05), Testing for Weak Security Question Answer (WSTG-ATHN-08)* dan *Testing Directory Traversal/File Include (WSTG-ATHZ-01).* Beberapa kemungkinan celah keamanan tersebut dianalisis berdasarkan daftar celah keamanan *OWASP WSTG v4.2* dengan hasil yang diperoleh web MI-Gateway UII *broken access control, insecure design* serta *identification and authentication failures.*

Kata kunci: *OWASP WSTG v4.2,* Keamanan Informasi, web, *vulnerability*.

GLOSARIUM

Attacker	Seseorang yang mencoba untuk menyerang, menyusupi dan mengambil data dari sebuah sistem.
Cyber Security	Tindakan yang dilakukan untuk melindungi sistem komputer dari serangan atau akses ilegal.
Firewall	Sistem keamanan yang melindungi komputer pada jaringan internet.
Host	Sistem dalam server yang saling terhubung secara langsung dengan server yang lain.
IP Address	Identitas berupa angka yang digunakan untuk menghubungkan semua komputer dalam jaringan internet.
Kali Linux	Sistem operasi terbuka berbasis *debian* yang berisi banyak *tools* digunakan dalam pengujian keamanan sistem informasi.
Scanning	Proses memindai sistem atau target yang akan diuji.
Server	Sistem komputer sebagai penyedia jenis layanan tertentu pada jaringan komputer.
Threats	Ancaman keamanan yang berasal dari dalam maupun luar sistem.
URL	Rangkaian karakter yang digunakan untuk mengakses suatu situs web.

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

UNIVERSITAS ISLAM INDONESIA

BAB I
PENDAHULUAN

1.1 Latar Belakang

Di era sekarang ini banyak sekali aplikasi yang digunakan untuk mempermudah pekerjaan. Aplikasi muncul dengan berbagai macam fitur dan layanan, kemampuan untuk menyimpan data atau informasi penting, membagikan data, ataupun menemukan informasi dengan mudah. Banyak perusahaan dan institusi besar menggunakan aplikasi untuk menyimpan data dan informasi.

Data dan informasi merupakan suatu hal yang sangat penting bagi sebuah perusahaan atau institusi. Akses data dan informasi semakin mudah dengan adanya internet, namun tidak semua informasi dan data dapat diakses secara bebas oleh semua pihak. Informasi yang memuat data pribadi dikatakan informasi yang bersifat konfidensial, diartikan bahwa hanya pihak yang berwenang yang dapat mengaksesnya. Data tersebut digunakan untuk menunjang proses bisnis pada sebuah perusahaan atau institusi yang tersimpan pada ruang penyimpanan dan dilengkapi dengan sistem keamanan.

Keamanan informasi erat kaitannya dengan *CIA (Confidentiality, Integrity, and Availability)* yang dijadikan sebagai acuan dalam sebuah web menjadi salah satu parameter dalam melakukan analisis celah keamanan (Guntoro, Costaner, and Musfawati 2020). Tujuannya adalah mengidentifikasi serta mengetahui jenis-jenis serangan yang dapat terjadi akibat kerentanan dan kelemahan pada sistem. Karena semakin berharga suatu data dan informasi pasti akan semakin rentan menjadi target serangan, tidak terkecuali dengan cara meretasnya atau melakukan *hacking*.

MI-Gateway UII merupakan aplikasi berbasis web yang dimiliki oleh Program Studi Informatika Program Magister Universitas Islam Indonesia. Web MI-Gateway UII memiliki beberapa fitur seperti SIM-PMB, SIM-TESIS, SIM-NKMD dan SIM-Publikasi. MI-Gateway UII memerlukan pengujian keamanan web dalam upaya menguji autentikasi dan otorisasi layanan web MI-Gateway UII. Karena aplikasi berbasis *web* ini menjadi portal bagi calon mahasiswa maupun mahasiswa aktif, terdapat berbagai macam data penting dari mahasiswa maupun dosen.

OWASP WSTG v4.2 digunakan sebagai panduan komprehensif dalam pengujian keamanan aplikasi dan layanan web MI-Gateway UII. Dalam *OWASP WSTG* v4.2 terdapat sebelas poin pengujian keamanan web yang dapat digunakan oleh *developer* sebagai panduan pengujian

keamanan web. *OWASP WSTG* v4.2 merupakan referensi utama dalam melakukan pengujian keamanan web, selain itu juga di setiap poin pengujiannya berisi detail pengujian yang membantu dalam proses pengujian.

Autentikasi dan otorisasi merupakan poin pengujian yang dipilih pada penelitian ini karena berkaitan dalam keamanan web MI-Gateway UII dan fokus dalam mengamankan *resource corporate.* Autentikasi merupakan proses dimana seorang pengguna mendapatkan hak akses terhadap suatu *entity*. Sedangkan, otorisasi merupakan proses penentuan apakah pengguna tersebut diizinkan atau ditolak untuk diberikan akses terhadap *resources* tertentu. Tujuan dilakukannya pengujian autentikasi dan otorisasi ini untuk peningkatan keamanan terhadap asset informasi web Mi-Gateway UII terhadap ancaman jaringan maupun ancaman internet.

Alat pengujian keamanan web MI-Gateway UII terbagi ke dalam tiga kategori yakni *operating system, scanning tools* dan *word processing*. Terkait dengan *scanning tools* yang digunakan yaitu *OWASP ZAP, Nikto, BurpSuite, SQLMap* dan *THC-Hydra*. Alat bantu dalam proses pemindaian telah direkomendasikan oleh panduan pengujian keamaan *OWASP WSTG* v4.2.

1.2 Rumusan Masalah

Berdasarkan uraian yang telah dijelaskan pada latar belakang di atas, dapat ditarik beberapa rumusan masalah sebagai berikut:

a. Bagaimana cara mengetahui celah keamanan yang terdapat pada web MI-Gateway UII?
b. Bagaimana hasil pengujian keamanan web MI-Gateway UII?

1.3 Tujuan Penelitian

Berdasarkan uraian yang telah dijelaskan pada latar belakang dan rumusan masalah di atas, tujuan dari penelitian ini adalah sebagai berikut:

a. Melakukan analisis celah keamanan web MI-Gateway UII menggunakan metode pengujian *OWASP WSTG v4.2*.
b. Mengetahui celah keamanan web MI-Gateway UII sehingga dapat menjadi acuan dalam memperbaiki celah keamanan pada web MI-Gateway UII.

1.4 Batasan Masalah

Untuk memfokuskan masalah yang ada, diperlukan sebuah batasan-batasan agar bisa terfokus dengan masalah yang ada. Oleh sebab itu, batasan masalah dalam kasus ini sebagai berikut:

a. Web yang akan diuji adalah web milik Program Studi Informatika Program Magister Universitas Islam Indonesia yaitu MI-Gateway UII.
b. Pengujian dilakukan berdasarkan panduan pengujian *OWASP WSTG v4.2* dengan kategori pengujian autentikasi dan otorisasi.
c. Tidak melakukan perbaikan pada celah keamanan web MI-Gateway UII.

1.5 Manfaat Penelitian

Manfaat yang diharapkan didapatkan dari penelitian ini adalah sebagai berikut:

a. Untuk institusi, dengan penelitian ini institusi bisa mendapatkan manfaat positif yaitu berupa evaluasi terhadap web yang dapat ditingkatkan kualitas keamanan datanya.
b. Untuk pengguna website, dengan penelitian ini user dapat lebih waspada karena web telah dilakukan evaluasi melalui pengujian terhadap keamanan data yang pada web.
c. Untuk peneliti lain, dengan adanya penelitian ini bisa dijadikan sebagai salah satu referensi untuk melaksanakan penelitian selanjutnya.

1.6 Metodologi Penelitian

Metodologi penelitian ini dilakukan agar dalam proses pengujian yang dilakukan dapat lebih terarah, sesuai rencana dan mencapai tujuan yang diharapkan. Adapun metodologi yang diterapkan dalam pembuatan tugas akhir ini adalah sebagai berikut:

a. Analisis Pertanyaan

Analisis pertanyaan dalam penelitian ini digunakan untuk memetakan latar belakang masalah yang akan dipecahkan sebagai bentuk analisis sistem dan penelitian yang hendak dilakukan dengan membuat pertanyaan yang terdiri dari 5W *(What, Where, Why, Who, When)* + 1H *(How)* dan disertai dengan jawaban.

b. Studi Literatur

Studi literatur digunakan dalam memetakan kajian pustaka yang dijadikan pendukung penelitian. Dapat berupa teori yang didapatkan melalui buku, jurnal, artikel maupun laporan terdahulu yang terkait dengan penelitian.

c. Pengumpulan Data dan Identifikasi Sistem

Pengumpulan data *(footprinting)* dalam penelitian ini dilakukan dengan mencari data atau informasi target yang hendak diuji yaitu web MI-Gateway UII. Selanjutnya dilakukan identifikasi sistem dengan cara *scanning* web target untuk mengetahui celah keamanannya.

d. Pengujian Penetrasi

Pengujian penetrasi dilakukan pada web target berdasarkan panduan pengujian *OWASP WSTG v4.2* yang berfokus pada pengujian autentikasi dan otorisasi.

e. Analisis Hasil dan Penyusunan Dokumentasi Laporan

Setelah dilakukan pengujian, selanjutnya dilakukan analisis terhadap hasil pengujian yang dilakukan. Setelah diketahui hasilnya, selanjutnya menyusun laporan penelitian berdasarkan pengujian penetrasi pada web target menggunakan panduan pengujian *OWASP WSTG v4.2*.

1.7 Sistematika Penulisan

Untuk memberikan gambaran secara menyeluruh mengenai masalah yang akan dibahas dalam penulisan laporan tugas akhir ini, sistematika laporan ini dibagi menjadi lima bab. Adapun penjabarannya sebagai berikut:

BAB I PENDAHULUAN

Bab pendahuluan berisi mengenai latar belakang masalah, rumusan masalah, tujuan masalah, batasan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan laporan pengujian keamanan pada web MI-Gateway UII berdasarkan panduan *OWASP WSTG v4.2*.

BAB II LANDASAN TEORI

Bab ini membahas tentang dasar teori yang diterapkan dalam *penetration testing* menggunakan metode *OWASP WSTG v4.2*. Selain itu, dalam bab ini juga terdapat penjelasan tentang metode dan *tools* yang digunakan untuk melakukan *penetration testing*.

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang metode dan alat bantu yang dilakukan dalam penelitian. Selain itu, terdapat langkah-langkah proses pengujian yang dilakukan berdasarkan panduan *OWASP WSTG v4.2*.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi tentang analisis pengujian celah keamanan berdasarkan panduan keamanan *OWASP WSTG v4.2* dan berdasarkan daftar hasil kerentanan yang ada pada web MI-Gateway UII.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi penutup yang meliputi kesimpulan-kesimpulan dari hasil pengujian yang telah dilakukan sebelumnya yang berupa hasil analisis pengujian yang telah dilakukan dan terdapat saran-saran dari hasil pengujian.

BAB II
LANDASAN TEORI

Sebelum melakukan penelitian, dibutuhkan sumber daya berupa ilmu pengetahuan terkait penelitian yang akan dilakukan. Sumber daya tersebut dapat berupa penelitian serupa yang telah ada sebelumnya untuk dijadikan acuan penelitian, serta beberapa teori dasar terkait pengetahuan seputar pengujian sistem. Sehingga dari sumber daya ini, diharapkan dapat memberikan arahan serta gambaran dalam melakukan penelitian.

2.1 Dasar Teori

Pada dasar teori ini digunakan untuk menjelaskan kajian pustaka yang telah dilakukan dan dapat digunakan untuk mendukung penelitian berdasarkan teori-teori penunjang yang diperoleh dari membaca buku, jurnal, artikel, situs web maupun laporan penelitian terdahulu.

2.1.1 Keamanan Informasi

Definisi keamanan informasi menurut (ISO - ISO/IEC 17799:2005 n.d.) upaya untuk melindungi atau mengambil tindakan dalam mendeteksi dan mencegah akses tidak sah yang dapat mengakibatkan pencurian informasi, kerusakan sistem informasi, hingga kerugian pada proses bisnis yang dikelola. Dua hal yang menjadi masalah utama pada keamanan informasi yaitu *threats* (ancaman) dan *vulnerability* (kerentanan) (Guntoro, Costaner, and Musfawati 2020). Jadi, dapat disimpulkan bahwa keamanan inforamsi adalah mencegah dan mendeteksi tindakan yang berupa akses tidak sah, pencurian informasi, perubahan program atau kerusakan fisik terhadap sistem informasi yang dapat menyebabkan kehilangan data dan informasi serta kerugian terhadap proses bisnis yang ada.

Dalam buku berjudul *Principles of Incident Response and Disaster Recovery* (Michael E. Whitman 2020), ada beberapa faktor dalam keamanan sistem informasi yaitu:

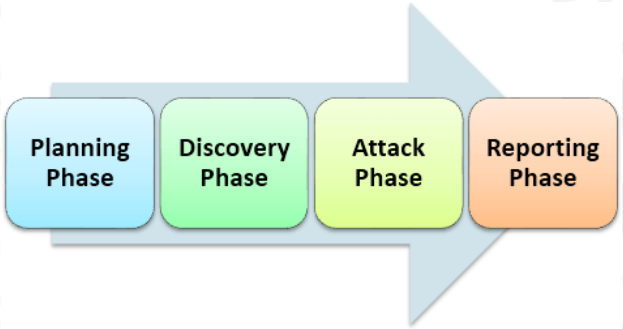
a. Kesalahan manusia (*Acts of human error or failure*): ancaman karena kesalahan manusia di mana kejadian tersebut bukan disengaja atau tanpa maksud jahat.
b. Hak Atas Kekayaan Intelektual atau yang biyasa disingkat HAKI (*Compromises to intellectual property* (IP)): ancaman dari pelanggaran dalam penggunaan HAKI seperti hak cipta, rahasia dagang, merek dagang, hak paten. Pelanggaran HAKI yang paling umum adalah pembajakan perangkat lunak.

c. Pelanggaran yang disengaja (*Deliberate acts of trespass*): mengakses secara tidak sah ke informasi yang bersifat rahasia dan pribadi. Contohnya seorang *hacker* menggunakan perangkat lunak untuk mendapatkan akses ke informasi secara ilegal.
d. Tindakan untuk tujuan pemerasan: menuntut kompensasi untuk mengembalikan rahasia informasi yang diperoleh oleh penyerang.
e. Tindakan disengaja untuk sabotase atau vandalisme: upaya untuk menghancurkan aset atau merusak citra organisasi.
f. Tindakan pencurian yang disengaja: mengambil barang orang lain secara ilegal.
g. Serangan dengan perangkat lunak: perangkat lunak yang berbahaya yang dirancang untuk merusak, menghancurkan, atau menolak layanan ke sistem, termasuk virus, *worm, trojan horse, backdoors*, serangan Denial of Service (DoS) dan Distributed Denial of Service (DDoS).
h. Kejadian alam: Tak terduga dan sering tidak dapat diramalkan, termasuk kebakaran, banjir, gempa bumi, petir, badai, letusan gunung berapi.
i. Penyimpangan dalam kualitas pelayanan, oleh penyedia layanan. Produk atau jasa terhenti atau tidak dapat berjalan sebagai mana mestinya seperti listrik, air, *bandwidth* jaringan, dll.
j. Kerusakan atau kesalahan teknis dari peralatan: cacat bawaan peralatan yang menyebabkan sistem bekerja tidak sesuai dengan diharapkan, menyebabkan layanan tidak dapat diberikan dengan baik atau kurangnya ketersediaan.
k. Kesalahan dan kegagalan Software: termasuk *bug* dan kondisi tertentu yang belum teruji. Mungkin termasuk cara pintas (*shortcut*) yang sengaja dibuat oleh *programmer* untuk alasan tertentu tetapi lupa untuk dihapus.
l. Teknik dan peralatan yang telah usang: infrastruktur yang sudah ketinggalan zaman menyebabkan sistem tidak dapat diandalkan dan tidak dapat dipercaya.

Untuk mengurangi risiko terhadap keamanan suatu informasi dan menjaga proses bisnis tetap berjalan sebagaimana mestinya agar tidak menimbulkan kerugian diperlukan adanya pengujian terhadap suatu sistem atau web. Tujuannya untuk mencari celah keamanan yang terdapat pada sistem tersebut untuk segera dilakukan pencegahan lebih dini terhadap serangan oleh pihak yang tidak bertanggung jawab dan dapat menimbulkan kerugian.

2.1.2 Penetration Testing

Pengujian penetrasi adalah metode pengujian sistem komputer atau jaringan untuk tujuan mengevaluasi keamanan sistem komputer atau jaringan. Evaluasi dilakukan dengan melakukan simulasi serangan pada suatu sistem atau jaringan untuk menemukan kerentanan keamanan akibat kelemahan pada sistem, konfigurasi yang salah atau kelemahan operasional dalam proses teknis. Laporan hasil uji penetrasi akan memberikan informasi kepada pemilik sistem tentang celah keamanan pada sistem mereka yang dapat digunakan sebagai tolak ukur untuk menjalankan sistem keamanan komputer guna memecahkan masalah kelemahan sistem sehingga tindakan dapat segera diambil untuk mencegahnya. Proses pengujian penetrasi dapat dilihat pada Gambar 2. 1.



Gambar 2. 1 Proses *Penetration Testing* (ISBuzz 2021)

2.1.3 *Open Web Application Security Project* (OWASP)

Open Web Application Security Project atau OWASP adalah kerangka kerja sebuah *framework* yang bersifat *open source*, dibangun oleh suatu organisasi dalam menemukan dan memperbaiki celah keamanan pada suatu aplikasi berbasis web. Terdapat sebelas panduan yang digunakan untuk menguji keamanan sebuah web menurut standar yang dikeluarkan oleh *OWASP v4* yaitu *Information Gathering, Configuration and Deploy Management Testing, Identity Management Testing, Authentication Testing, Authorization Testing, Session Management Testing, Data Validation Testing, Error Handling, Cryptography, Business Logic Testing, Client Side Testing* (Owasp 2017).

2.1.4 *Web Security Testing Guide* (WSTG)

Web Security Testing Guide menghasilkan sumber daya pengujian keamanan siber utama untuk pengembang aplikasi web dan profesional keamanan. *WSTG* adalah panduan komprehensif untuk menguji keamanan aplikasi web dan layanan web. *WSTG* dibuat oleh

upaya kolaboratif dari para profesional keamanan siber dan sukarelawan di bidang keamanan sistem, *WSTG* menyediakan kerangka kerja praktik terbaik yang digunakan oleh penguji penetrasi dan organisasi di seluruh dunia (Drake 2020).

A. Authentication Testing

Pengujian autentikasi atau *authentication testing* adalah suatu tindakan yang dilakukan untuk mengonfirmasi segala sesuatu yang mengatakan suatu hal itu adalah benar. Autentikasi objek dapat dilakukan dengan melakukan konfirmasi asal objek tersebut, sedangkan melakukan autentikasi terhadap seseorang dapat dilakukan dengan memverifikasi identitasnya. Berdasarkan dokumen OWASP WSTG v4.2, *authentication testing* terdiri dari sepuluh tipe pengujian yaitu:

1. *Testing for Credentials Transported over an Encrypted Channel* (WSTG-ATHN-01)

Kredensial pengguna ini harus selalu ditransfer melalui saluran terenkripsi (HTTPS) untuk menghindari penyerang melakukukan *sniffing* jaringan untuk melihat kredensial dan mungkin digunakan untuk mencuri akun pengguna. Fakta bahwa lalu lintas yang dienkripsi tidak berarti aman dari penyerang, tergantung pada algoritma enkripsi yang digunakan dan kunci yang diterapkan oleh aplikasi.

2. *Testing for Default Credentials* (WSTG-ATHN-02)

Kredensial standar seringkali dikustomisasi oleh administrator dengan konfigurasi minimal. Kredensial standar dapat ditemukan pada beberapa *software* maupun *hardware* yang setelah di-*install* tidak dikonfigurasi dengan benar dan kredensial standar yang disediakan untuk autentikasi dan konfigurasi awal tidak pernah berubah. Dalam beberapa situasi ketika pengguna mendaftar akun baru pada aplikasi, kata sandi dibuat *default* dengan karakteristik standar.

3. *Testing for Weak Lock Out Mechanism* (WSTG-ATHN-03)

Mekanisme penguncian akun digunakan untuk keseimbangan antara melindungi akun dari akses tidak sah dan melindungi pengguna dari penolakan akses resmi. Dengan percobaan menebak *login password*, *username* atau kode pada fungsi pertanyaan *two-factor authentication* (2FA) akun pengguna akan dikunci setelah tiga sampai lima kali upaya yang gagal dan akan dibuka setelah jangka waktu yang ditentukan *administrator*.

4. *Testing for Bypassing Authentication Schema* (WSTG-ATHN-04)

Menguji skema autentikasi merupakan cara kerja dalam proses autentikasi serta menggunakan informasi tersebut untuk menghindari mekanisme autentikasi. Dalam keamanan

sistem, autentikasi merupakan proses verifikasi identitas digital pengguna. Implementasi autentikasi digunakan oleh sebagian besar sistem aplikasi untuk mendapatkan akses ke informasi pribadi atau untuk menjalankan tugas. Ancaman keamanan pada skema autentikasi yang dapat dilewati dengan melewatkan halaman *login*.

5. *Testing for Vulnerable Remember Password* (WSTG-ATHN-05)

Penggunaan fungsi dalam sebuah aplikasi yaitu "ingat saya" memungkinkan pengguna untuk tetap diautentikasi untuk jangka waktu yang lama tanpa meminta kredensial pengguna lagi. Pengolaan kata sandi termasuk di dalam *browser* yang memberikan layanan pada pengguna untuk menyimpan kredensial mereka dengan cara yang aman kemudian diisikan secara otomatis pada formulir *login* tanpa campur tangan pengguna.

6. *Testing for Browser Cache Weaknesses* (WSTG-ATHN-06)

Browser dapat menyimpan informasi untuk tujuan *caching*. *Caching* merupakan peningkatan kinerja sehingga informasi yang ditampilkan sebelumnya tidak perlu untuk diunduh kembali. Apabila informasi sensitif ditampilkan kepada pengguna, informasi ini dapat disimpan dengan tujuan *cache* atau dapat dibatalkan dengan menekan tombol kembali pada *browser*. Penguji akan melakukan pemeriksaan pada aplikasi dengan memberi intruksi kepada *browser* untuk tidak menyimpan data dan informasi sensitif.

7. *Testing for Weak Password Policy* (WSTG-ATHN-07)

Mekanisme yang mudah dan umum digunakan adalah kombinasi kata sandi statis yang mudah dihafal dan ditebak. Dalam tindakan peretasan, mengungkapkan kredensial pengguna dengan kombinasi yang paling umum seperti: 123456, qwerty, dll. Tujuan pengujian ini adalah menghindari serangan *brute force* berdasarkan kamus kata sandi yang tersedia.

8. *Testing for Weak Security Question Answer* (WSTG-ATHN-08)

Sering disebut dengan pertanyaan dan jawaban keamanan pemulihan pada beberapa kasus lupa kata sandi. Pertanyaan ini diisikan pada saat pembuatan akun baru pengguna, yang dilakukan kombinasi antara pertanyaan dan jawaban. Fungsi dari pertanyaan ini untuk memulihkan kata sandi yang terlupakan atau penyetelan kata sandi yang lemah.

9. *Testing for Weak Password Change or Reset Functionalities* (WSTG-ATHN-09)

Perubahan dan pengaturan ulang kata sandi pengguna pada suatu sistem aplikasi merupakan mekanisme standar. Pada saat kata sandi diubah, dapat dilakukan melalui aplikasi langsung atau melalui *direct email* yang nanti akan diisi oleh pengguna tanpa campur tangan *administrator*.

10. *Testing for Weaker Authentication in Alternative Channel* (WSTG-ATHN-10)

Jika mekanisme autentikasi utama tidak memiliki kerentanan apapun, dapat terjadi kerentanan pada saluran alternatif untuk akun pengguna yang sama. Digunakan untuk menghindari saluran utama atau mengekspos informasi yang kemudian dapat membantu serangan terhadap saluran utama. Saluran alternatif ini terpisah dan menggunakan *host* yang berbeda dari saluran utama. Pengujian dilakukan untuk mengidentifikasi saluran alternatif menerapkan autentikasi yang tepat agar terlindungi dari serangan.

B. Authorization Testing

Pengujian otorisasi atau *authorization testing* merupakan suatu konsep yang digunakan untuk memberikan izin akses kepada seseorang yang diizinkan untuk mengakses sumber daya yang dituju. Pengujian otorisasi harus memahami proses otorisasi yang dilakukan setelah proses autentikasi berhasil dilakukan dengan menghindari mekanisme otorisasi menggunakan informasi tersebut. Verifikasi dilakukan setelah mendapatkan kredensial yang valid mengenai serangkaian peran dan hak istimewa yang terdifinisi dengan baik. Berdasarkan dokumen OWASP WSTG v4.2, *authorization testing* terdiri atas empat tipe pengujian:

1. *Testing Directory Traversal File Include* (WSTG-ATHZ-01)

Server dan aplikasi web menerapkan mekanisme autentikasi dalam mengontrol akses ke *file* dan sumber daya aplikasi. Penggunaan metode validasi input yang belum dirancang atau diterapkan dengan baik, penyerang dapat melakukan eksploitasi sistem dalam membaca atau menulis file yang seharusnya tidak dapat diakses. *Access Control Lists* (ACL) digunakan dalam identifikasi pengguna atau grup mana yang dapat mengakses, mengubah atau menjalankan file pada server.

2. *Testing for Bypassing Authorization Schema* (WSTG-ATHZ-02)

Merupakan aktivitas memverifikasi dengan menerapkan skema otorisasi untuk setiap peran atau hak istimewa pengguna dalam mendapatkan akses ke fungsi dan sumber daya yang dicadangkan. Dalam setiap peran pengguna yang spesifik dimiliki dengan mempertimbangkan akses sumber daya tanpa autentikasi, akses sumber daya setelah *logout* atau akses fungsi dan sumber daya yang seharusnya hanya dapat diakses oleh *user* istimewa.

3. *Testing for Privilege Escalation* (WSTG-ATHZ-03)

Peningkatan hak istimewa dari satu tahap ke tahap lainnya yang dilakukan oleh administrator tanpa adanya campur tangan pengguna. Peningkatan hak istimewa dilakukan ketika pengguna mendapatkan akses ke lebih banyak sumber daya atau fungsionalitas dalam

aplikasi. Hal ini seharusnya dicegah oleh aplikasi sebab adanya cacat dalam aplikasi yang menyebabkan perubahan eskalasi hak istimewa dan menjadi celah bagi penyusup untuk mendapatkan hak istimewa dalam aplikasi dengan eskalasi yang lebih besar.

4. *Testing for Insecure Direct Object References* (WSTG-ATHZ-04)

Insecure Direct Object References (IDOR) dapat terjadi ketika aplikasi menyediakan akses langsung ke objek berdasarkan input yang disediakan pada pengguna. Referensi objek langsung yang tidak aman akan memberi celah kepada penyerang melewati otorisasi dan mengakses sumber daya secara langsung dengan modifikasi nilai parameter yang digunakan. Hal ini disebabkan bahwa aplikasi dapat mengambil input yang diberikan pengguna dan menggunakannya untuk mengambil objek tanpa melakukan pemerikasaan otorisasi yang memadai.

2.1.5 Scanning Tools

Scanning Tools atau alat pemindai pada kerentanan aplikasi web merupakan alat otomatis yang memindai aplikasi web, tujuannya untuk mencari kerentanan keamanan seperti skrip lintas situs, injeksi *SQL*, injeksi perintah, lintasan jalur dan konfigurasi server yang tidak aman. *Scanning tools* merupakan alat komersial yang *open source* dan semua alat bantu ini memiliki kekuatan dan kelemahannya sendiri (Sunardi, Riadi, and Raharja 2019).

2.1.6 Black Box Testing

Black box testing merupakan proses di mana penguji sebagai orang luar yang sama sekali tidak memahami dan memiliki informasi tentang sistem atau jaringan yang akan diuji sehingga penguji harus mencari segala informasi yang berkaitan dengan sistem tersebut untuk dilakukan analisis dan menentukan metode yang digunakan pada saat melakukan serangan (Kumari 2020).

2.1.7 Web Analysis Scanning

Web analysis scanning merupakan tahap seorang penyerang melakukan analisis secara mendalam terhadap web target yang akan diserang. Terdapat banyak cara dalam melakukan *web analysis scanning* antara lain dengan cara manual menggunakan *browser* atau menggunakan *tools vulnerability scanner* (Nagendran et al. 2019).

2.2 Penelitian Terkait

Penelitian *Web Application Penetration Testing* (Nagendran et al. 2019) menghasilkan Penetrasi Aplikasi *Web* sebagai kebutuhan pengujian keamanan sistem informasi saat ini. Situs web membutuhkan pendekatan pertahanan yang mendalam untuk mengurangi kelemahan keamanan. Sangat penting untuk menguji penetrasi setiap aplikasi *web* untuk menemukan celah kerentanan yang dapat diretas oleh *cyber black-hat.*

Penggunaan dokumen OWASP WSTG berkaitan dengan tools OWASP ZAP sebagai tools untuk melakukan pengujian keamanan suatu web. Pada penelitian *Vulnerability Assessment and Penetration Testing of Web Application* (Nagpure and Kurkure 2018) kerentanan keamanan dapat mengakibatkan pelanggaran integritas data, mencuri data rahasia atau memengaruhi ketersediaan layanan aplikasi *web*. Dengan demikian tugas mengamankan aplikasi web adalah salah satu yang paling penting.

Menurut penelitian Penerapan Metode ISSAF dan OWASP versi 4 Untuk Uji Kerentanan *Web Server* (Dirgahayu, Prayudi, and Fajaryanto 2016) untuk mengamankan web server dari serangan *hacker* sebaiknya para pemilik server web melakukan *self test* terhadap server mereka sendiri. Melalui *self test* ini, para pemilik server web akan mengetahui letak kerentanan dari sistem yang ada. Salah satu metode *self test* ini adalah *penetration test*. Metode ini sama dengan aktivitas *hacking* namun dilakukan secara legal.

Dengan demikian penggunaan metode OWASP (*Open Web Application Security Project*) dapat digunakan sebagai acuan dalam melakukan pengujian pada suatu sistem lebih spesifiknya untuk aplikasi web. Kerentanan dapat ditemukan melalui langkah–langkah pengujian berdasarkan metode yang digunakan. Menggunakan *penetration testing black box* peneliti akan menggali dari awal informasi yang didapat kemudian melakukan analisis serta menentukan jenis serangan yang akan dilakukan. Selanjutnya akan memberikan *report* seberapa rentan aplikasi web tersebut.

Tabel 2. 1 Metode dan *Tools* dalam Analisis Celah Keamanan *Website*

Peneliti (Tahun)	**Target**	**Kerangka Kerja**	***Tools***
K Nagendran, A Adithyan, R Chethana, P Camillus, K. B Bala Sri Varshini (2019)	Aplikasi Berbasis Web (Nama Aplikasi tidak disebutkan)	OWASP Top 10	Nmap, Nikto, Acunetix, W3af

Peneliti (Tahun)	**Target**	**Kerangka Kerja**	***Tools***
Sangeeta Nagpure, Sonal Kurkure (2018)	Aplikasi E-Commerce dan Aplikasi Cloud (Nama Aplikasi tidak disebutkan)	OWASP Top 10 dan VAPT	Acunetix, Burp Suite, OWASP Zap
Balume Mburano, Weisheng Si (2019)	Web pemindai kerentanan	OWASP dan WAVSEP	Arachi, ZAP
Moh Yunus (2019)	Aplikasi Web (Nama Aplikasi tidak disebutkan)	OWASP v.4	ZAP, Netsparker, HAVIJ 1.15
Guntoro, Guntoro Costaner, Loneli Musfawati, Musfawati (2020)	OJS Universitas Lancang Kuning	ISSAF dan OWASP	OWASP ZAP, Mozilla firefox, Google Chrome, Brutus, WebScarab, Wfuzz, Dirb, OWASP CSRF Tester, Zenmap, Whois, Acunetix, SSL Scan, Low Orbit Ion Canon, SQLmap
Adetya Putra Dewanto (2018)	Domain uii.ac.id	OWASP Top 10	OWASP ZAP, WPScan, Web Browser, WhoIs, Nmap, Xpobe2, Zenmap

Berdasarkan Tabel 2. 1 diperoleh kesimpulan pada setiap metode penelitian yang digunakan sebagai berikut: menurut Nagendran et. al. (2019) dalam pengujiannya menggunakan metode *OWASP TOP 10* untuk menguji aplikasi web. Pengujian ini dilakukan untuk mengamankan *web* dari *black hat hacker*. Alat bantu yang digunakan *Nmap, Nikto, W3af, Acunetix* dan *CVSS*. Diperoleh kesimpulan bahwa penggunaan mekanisme *proxy* dalam pengujian penetrasi lebih baik dari modifikasi *log*.

menurut Nagpure dan Kurkure (2018) yang melakukan pengujian pada aplikasi *e-commerce* dan aplikasi *cloud* menggunakan metode *OWASP Top 10* dan *VAPT*. Alat bantu yang digunakan yaitu *Acunetix, BurpSuite, OWASP ZAP*. Diperoleh hasil bahwa pengujian penetrasi yang dilakukan secara manual lebih efektif dalam hal akurasi. Atas dasar kemampuan

aplikasi dan pengetahuan penguji akan menerapkan serangan yang lebih baik sehingga hasil penetrasi dapat diperoleh secara maksimal.

menurut Mburano dan Si (2019) yang melakukan pengujian penetrasi menggunakan metode *OWASP* dan *WAVSEP* pada aplikasi *web vulnerability scanner*. Alat bantu yang digunakan *Arachi* dan *OWASP ZAP*. Diperoleh hasil bahwa metode *OWASP* memberi pengujian yang lebih *expert* dan menjadi tolak ukur utama dalam pengujian, sedangkan *WAVSEP* sebagai tolak ukur sekunder untuk melengkapi hasil evaluasi.

menurut Yunus (2019) dalam penelitiannya melakukan analisis celah keamanan aplikasi *web* berdasarkan metode *OWASP version 4* dengan lima kategori penelitian yaitu *Authentication Testing, Authorization Testing, Session Management Testing, Input Validation Testing* dan *Error Handling*. Alat bantu yang digunakan dalam pengujian *OWASP ZAP, Netsparker* dan *HAVIJ 1.15*. Kesimpulan dari hasil pengujian yang dilakukan yaitu metode *OWASP version 4* dapat digunakan sebagai standar dalam penilaian analisis kerentanan dan keamanan suatu aplikasi berbasis web.

Berdasarkan penelitian yang dilakukan Guntoro et. al. (2020) dalam analisis keamanan pada *web server Open Journal System (OJS)* pada suatu universitas dengan metode yang digunakan yaitu *ISSAF* dan *OWASP*. Dengan alat bantu yang digunakan *OWASP ZAP, Mozilla firefox, Google Chrome, Brutus, WebScarab, Wfuzz, Dirb, OWASP CSRF Tester, Zenmap, Whois, Acunetix, SSL Scan, Low Orbit Ion Canon, SQLmap*. Mendapatkan hasil bahwa metode *ISSAF* memiliki kerentanan terhadap serangan *DoS*. Sementara itu, metode *OWASP* tergolong aman. Namun, kelemahan *OWASP* terdapat pada *weak lock out mechanism* di mana kesalahan percobaan *login* seorang *user* berulang kali tidak diblokir.

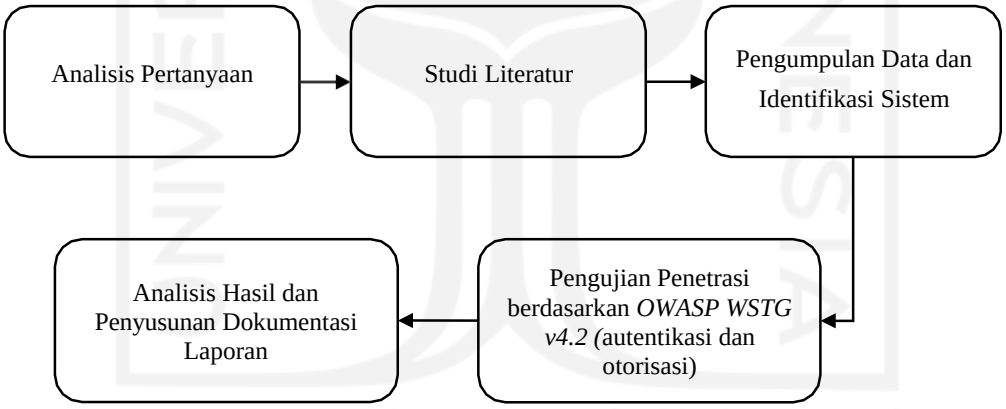
Berdasarkan penelitian Putra Dewanto (2018) yang melakukan pengujian *penetration testing* terhadap *domain uii.ac.id* menggunakan metode *OWASP Top 10*. Dengan alat bantu yang digunakan *OWASP ZAP, WPScan, Web Browser, WhoIs, Nmap, Xpobe2, Zenmap*, diperoleh hasil yaitu metode *OWASP Top 10* dapat menjadi panduan dasar dalam pengujian penetrasi aplikasi web.

BAB III
METODOLOGI PENELITIAN

Dalam melakukan penelitian, metodologi dibutuhkan sebagai tata cara yang digunakan dalam memperoleh suatu hasil penelitian. Kemudian, tata cara tersebut disusun secara sistematis agar proses penelitian dapat terlaksana secara terarah dan tertib. Tata cara yang tersusun secara sistematis ini, diperoleh berdasarkan acuan teori yang telah dipelajari sebelumnya.

3.1 Metode Penelitian

Dalam melakukan pengujian terhadap web MI-Gateway UII ini terdapat beberapa tahapan penelitian yang mengadaptasi diagram proses pengujian yang dilakukan oleh Yunus (2019). Perbedaan terletak pada metode pengujian yang digunakan dalam penelitian ini yaitu *OWASP WSTG v4.2* dengan fokus pengujian terletak pada pengujian autentikasi dan otorisasi.



Gambar 3. 1 Diagram Alur Penelitian

Berdasarkan Gambar 3. 1 dapat dilihat bahwa tahapan penelitian ini terdiri dari lima tahap, sebagai berikut:

3.1.1 Analisis Pertanyaan

Tahapan analisis pertanyaan dilakukan dengan pembuatan pertanyaan 5W + 1H disertai jawaban yang berkaitan dengan penelitian yang akan dilaksanakan. Pertanyaan sebagai berikut:

1. Apakah yang diteliti dalam penelitian ini?

Jawaban: penelitian ini hendak meneliti standar keamanan suatu web MI-Gateway berdasarkan aspek autentikasi dan otorisasi.

2. Di mana penelitian ini dilaksanakan?

   Jawaban: penelitian ini dilaksanakan di lingkungan Universitas Islam Indonesia.

3. Menagapa penelitian ini perlu dilakukan?

   Jawaban: penelitian ini perlu dilakukan karena keamanan terhadap segala macam ancaman jaringan maupun internet adalah hal yang utama. Manajemen keamanan terhadap informasi adalah suatu keharusan yang harus di antisipasi dengan tujuan keamanan total terhadap asset informasi. Pintu masuk untuk mendapat akses melalui autentikasi dan otorisasi. Jadi, perlu dilakukan analisis apakah web tersebut sudah dilengkapi dengan standar keamanan yang memenuhi berdasarkan acuan standar keamanan *OWASP*.

4. Siapa yang dijadikan objek dalam penelitian?

   Jawaban: yang hendak dijadikan objek penelitian adalah web MI-Gateway UII.

5. Kapan penelitian ini dilaksanakan?

   Jawaban: penelitian ini dimulai sejak Februari 2021 tepatnya awal semester enam dengan target selesai pada akhir semester delapan.

6. Bagaimana alur dalam penelitian ini?

   Jawaban: alur penelitian ini telah dimuat ke dalam *flowchart* tahapan penelitian seperti yang tercantum pada Gambar 3. 1.

3.1.2 Studi Literatur

Tahap studi literatur ini digunakan untuk menjelaskan kajian pustaka yang telah dilaksanakan dan dapat digunakan untuk mendukung penelitian berdasarkan teori-teori penunjang yang diperoleh dari membaca buku, jurnal, artikel, situs web maupun laporan penelitian terdahulu. Hasil dari dilakukannya studi literatur ini berupa sekumpulan referensi terkait berdasarkan rumusan masalah, dengan tujuan sebagai dasar teori dalam melakukan penelitian serta memperkuat permasalahan (Dirgahayu, Prayudi, and Fajaryanto 2016). Tahap studi literatur ini dilakukan dengan pemetaan enam literatur terkait yang membahas mengenai metode pengujian celah keamanan *website, tools* yang digunakan dan kesimpulan dari setiap literatur terkait.

3.1.3 Pengumpulan Data dan Identifikasi Sistem

Tahapan pengumpulan data dan identifikasi sistem dilakukan dengan pengumpulan data target penelitian dalam hal ini adalah web MI-Gateway UII menggunakan *tools terminal* dan *Nmap* yang terdapat pada *Kali Linux*. Adapun analisis alat dan kebutuhan sistem yang dibutuhkan dalam penelitian ini sebagai berikut:

A. Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam pengumpulan data penelitian tercantum pada Tabel 3. 1 berikut:

Tabel 3. 1 Spesifik Perangkat Keras

Komponen	Spesifikasi Minimum	Spesifikasi yang digunakan
Processor	Pentium 4 atau procesor AMD64	AMD Ryzen 5 3500U (2.1 GHz)
RAM	512 MB	8 GB DDR4-2400 SDRAM
Storage	10GB	512 GB PCIe® NVMe™ M.2 SSD
VGA	128MB	AMD Radeon™ Vega 8 Graphics

Berdasarkan Tabel 3. 1 dapat dilihat spesifikasi perangkat keras yang digunakan dalam penelitian ini terdiri dari *Processor* yang memiliki spesifikasi *AMD Ryzen 5 3500U (2.1 GHz)*. Kapasitas memori internal pada *processor* memiliki spesifikasi, selain itu komponen *RAM* sebesar *8 GB DDR4-2400 SDRAM* dan komponen penyimpanan (*SSD*) sebesar 512GB. Komponen *VGA* dengan spesifikasi *AMD Radeon™ Vega 8 Graphics.*

B. Perangkat Lunak

Spesifikasi perangkat keras yang digunakan dalam pengumpulan data penelitian tercantum pada Tabel 3. 2 berikut:

Tabel 3. 2 Spesifik Perangkat Lunak

Komponen	Spesifikasi yang digunakan
Operating System	- Windows 10 version 19044.1826 - Kali Linux 2021.2 VirtualBox AMD64
Scanning Tools	- OWASP ZAP - Nikto - BurpSuite - SQLMap - THC-Hydra
Word Processing	Microsoft® Word 2019 MSO (Version 2206 Build 16.0.15330.20260)

Berdasarkan Tabel 3. 2 dapat dilihat spesifikasi perangkat lunak yang digunakan dalam penelitian ini terdiri dari *Operating System* yang memiliki spesifikasi *Windows 10 version 19044.1826* dan untuk mesin virtual menggunakan *operating system kali linux 2021.2 AMD64*. *Scanning tools* yang digunakan dalam penelitian ini yaitu *OWASP ZAP, Nikto, BurpSuite, SQLMap, THC-Hydra*. Untuk pengolah kata (*Word Processing*) menggunakan *Microsoft Word 2019 MSO version 2206*.

3.1.4 Pengujian Penetrasi Berdasarkan OWASP WSTG v4.2 (Autentikasi dan Otorisasi)

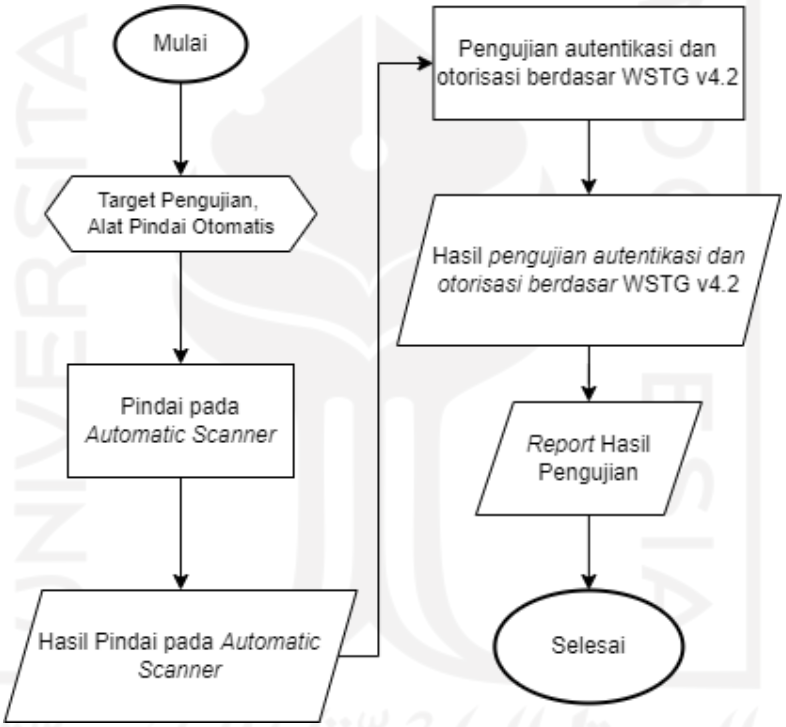
Tahapan pengujian penetrasi ini diawali dengan identifikasi celah keamanan *(vulnerability identification*) menggunakan *scanning tools* otomatis yaitu *OWASP ZAP*. Pada tahap pengujian ini fokus pada pengujian autentikasi dan otorisasi. Metode *OWASP WSTG v4.2* sebagai panduan pengujian autentikasi dan otorisasi seperti yang tertera pada Tabel 3.2 berikut:

Tabel 3. 3 Panduan Pengujian Keamanan *Web* pada *OWASP WSTG v4.2*

Kategori Pengujian	**Kode Pengujian**	**Tahapan Pengujian**
Authentication Testing	*WSTG-ATHN-01*	*Testing for Credentials Transported over an Encrypted Channel*
	WSTG-ATHN-02	*Testing for Default Credentials*
	WSTG-ATHN-03	*Testing for Weak Lock Out Mechanism*
	WSTG-ATHN-04	*Testing for Bypassing Authentication Schema*
	WSTG-ATHN-05	*Testing for Vulnerable Remember Password*
	WSTG-ATHN-06	*Testing for Browser Cache Weaknesses*
	WSTG-ATHN-07	*Testing for Weak Password Policy*
	WSTG-ATHN-08	*Testing for Weak Security Question Answer*
	WSTG-ATHN-09	*Testing for Weak Password Change or Reset Functionalities*
	WSTG-ATHN-10	*Testing for Weaker Authentication in Alternative Channel*
Authorization Testing	*WSTG-ATHZ-01*	*Testing Directory Traversal File Include*
	WSTG-ATHZ-02	*Testing for Bypassing Authorization Schema*
	WSTG-ATHZ-03	*Testing for Privilege Escalation*
	WSTG-ATHZ-04	*Testing for Insecure Direct Object References*

Berdasarkan Tabel 3. 3 dapat dilihat bahwa metode *OWASP WSTG v4.2* yang digunakan dalam pengujian ini terdapat dua kategori pengujian celah keamanan. Dalam kategori pengujian autentikasi (*authentication*) terdapat sepuluh poin pengujian dengan kode pengujian *ATHN*. Untuk kategori pengujian otorisasi (*authorization*) terdapat empat poin pengujian dengan kode pengujian *ATHZ*. Jadi, total poin pengujian secara keseluruhan terdapat empat belas poin pengujian.

Dalam melakukan pengujian pada web MI-Gateway-UII menggunakan metode OWASP WSTG v4.2 memiliki beberapa tahapan seperti yang dapat dilihat pada diagram alur yang terdapat pada Gambar 3. 2.



Gambar 3. 2 Diagram Alur pengujian

Dalam proses pengujian penetrasi web MI-Gateway UII ini memiliki diagram alur (*flowchart*) seperti pada Gambar 3. 2. Pertama yang dilakukan adalah menyiapkan target pengujian yaitu web MI-Gateway UII kemudian menjalankan aplikasi otomatisasi untuk memulai proses *scanning* terhadap target pengujian. Selanjutnya hasil pindai dijadikan bahan pada *report* hasil pengujian setelah pengujian berakhir. Kemudian dilakukan proses pengujian autentikasi menggunakan *tools* sebagai pendukung proses pengujian. Kemudian dilakukan proses pengujian otorisasi pada target pengujian menggunakan alat bantu yang direkomendasikan oleh WSTG v4.2. Setelah target dilakukan pengujian maka muncul hasil

dari uji kerentanan tersebut. Setelah proses selesai kemudian hasil *scanning*, hasil pengujian autentikasi dan otorisasi berdasarkan dokumen WSTG v4.2 akan dimasukkan ke dalam *report* dari hasil pengujian target.

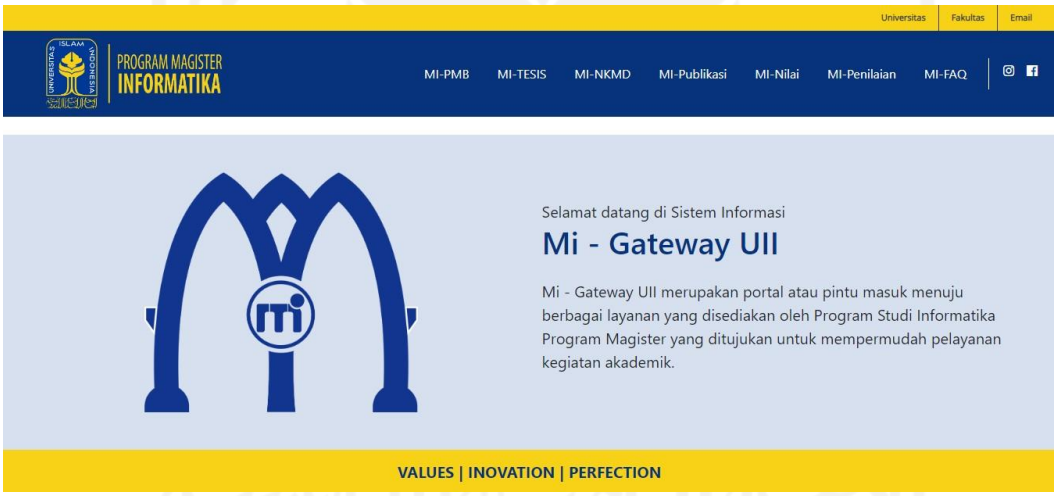
3.1.5 Analisis Hasil dan Pengumpulan Dokumentasi Laporan

Pada tahapan analisis hasil dan pengumpulan dokumentasi laporan pengujian ini dilakukan dengan pembuatan laporan yang berisi langkah pengujian dan analisanya. Untuk pengumpulan dokumentasi laporan dilakukan pada saat pengujian selesai dilakukan.

3.2 Alur Pengujian

Pada alur pengujian terdapat beberapa langkah yang dilakukan dalam melakukan pengujian keamanan web MI-Gateway UII.

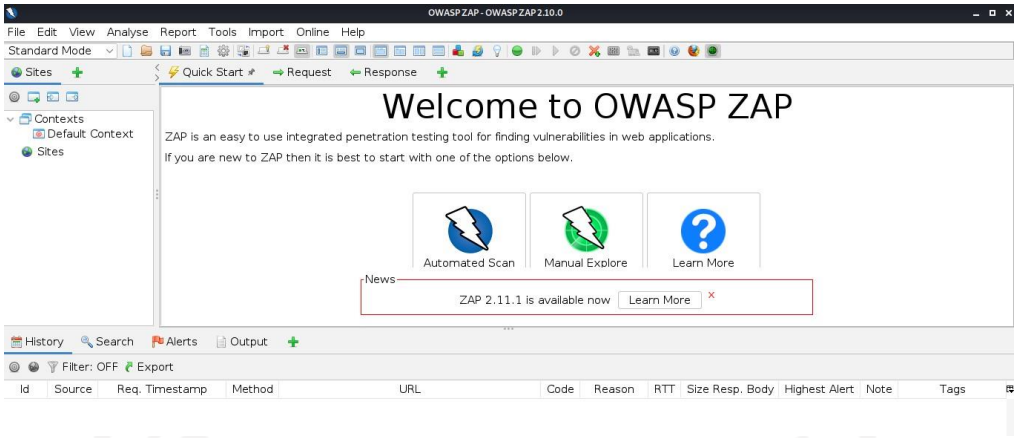
3.2.1 Identifikasi Target Pengujian dan Alat Pindai Otomatis Web MI-Gateway UII



Gambar 3. 3 Halaman Utama Web MI-Gateway UII

MI-Gateway UII merupakan aplikasi berbasis web yang dimiliki oleh Program Studi Informatika Program Magister Universitas Islam Indonesia. Web MI-Gateway UII memiliki beberapa fitur seperti SIM-PMB, SIM-TESIS, SIM-NKMD dan SIM-Publikasi. Pengujian ini akan melakukan eksploitasi pada fitur MI-PMB karena berkaitan dengan keamanan data dan informasi yang dimiliki oleh mahasiswa Program Studi Informatika Program Magister Universitas Islam Indonesia.

OWASP ZAP



Gambar 3. 4 Halaman Utama *OWASP ZAP*

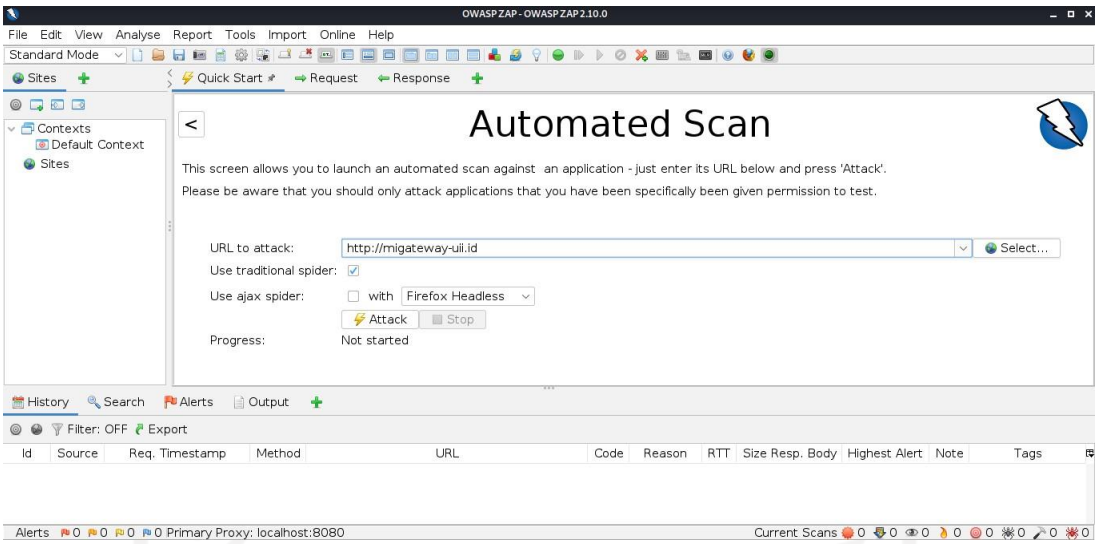
OWASP ZAP merupakan alat pemindai otomatis yang sangat populer dan sering digunakan oleh para penguji keamanan web. Terdapat dua pilihan metode proses *scanning* yakni *Automated Scan* dan *Manual Explore*. Pada penelitian ini akan menggunakan dua metode dalam proses pemindaian yaitu *Automated Scan* dan *Manual Explore*.

3.2.2 Proses Pemindaian dengan *Automatic Scanner*

Proses pemindaian dilakukan menggunakan *tools* OWASP ZAP v2.10.0. Proses *scanning* dilakukan dengan *automated scan* serta dikombinasikan dengan *manual explore* untuk memperoleh hasil yang lebih menyeluruh. Berikut merupakan langkah melakukan *automated scan* menggunakan OWASP ZAP:

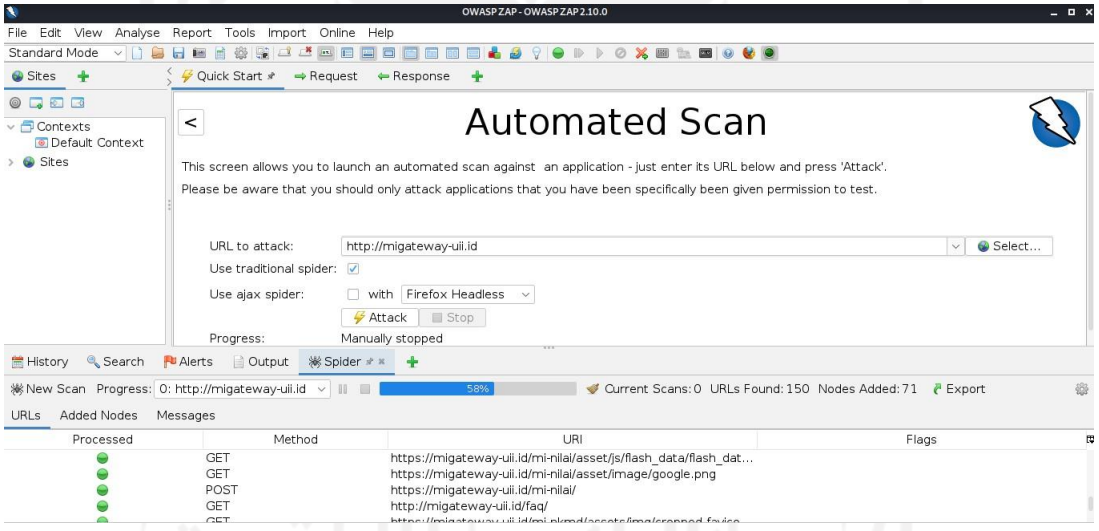
Automated Scan

Pada halaman utama aplikasi OWASP ZAP, pilih *automated scan* kemudian masukkan URL target serta *browser* yang akan digunakan untuk melakukan *automated scan*. Lalu klik *Attack* seperti Gambar 3. 5.



Gambar 3. 5 Proses *scanning* otomatis dengan OWASP ZAP

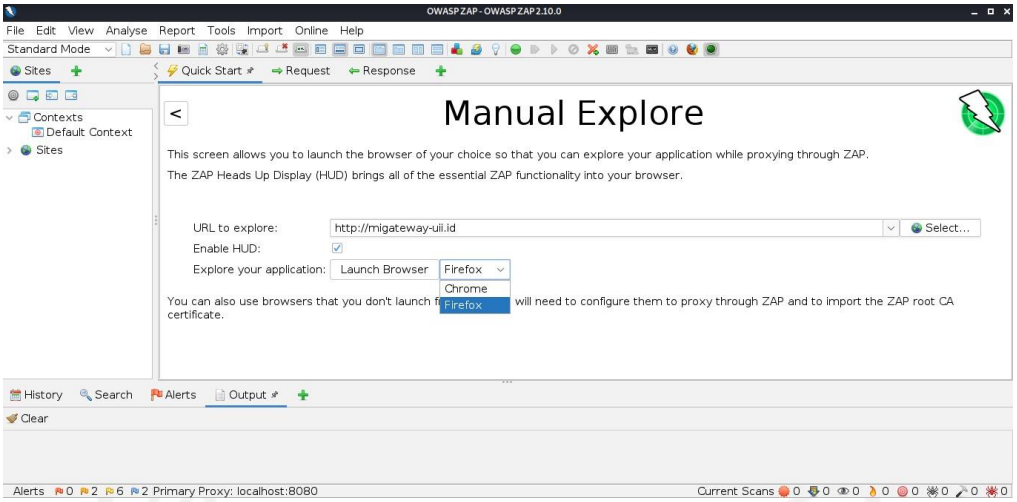
Tunggu beberapa saat proses *automated scan* berjalan hingga selesai. OWASP ZAP akan secara otomatis mendeteksi kerentanan yang ada pada *URL* target. Seperti terlihat pada Gambar 3. 6.



Gambar 3. 6 Proses *scanning* secara otomatis sedang berjalan

Langkah dalam melakukan proses *manual explore* pada OWASP ZAP sebagai berikut:

Pada halaman utama aplikasi OWASP ZAP, pilih *manual explore* kemudian masukkan URL target serta browser yang akan digunakan untuk melakukan *manual explore* kemudian klik *launch browser* untuk memulai proses seperti terlihat pada Gambar 3. 7.



Gambar 3. 7 Tampilan *manual explore* pada *OWASP ZAP*

Tunggu beberapa saat hingga *browser* akan muncul dan proses *manual explore* siap dilakukan. OWASP ZAP akan secara otomatis mendeteksi kerentanan disetiap *load* atau intruksi yang dijalankan terhadap halaman target seperti pada Gambar 3. 8.

Gambar 3. 8 Tampilan proses *manual explore* sedang berjalan pada *browser*

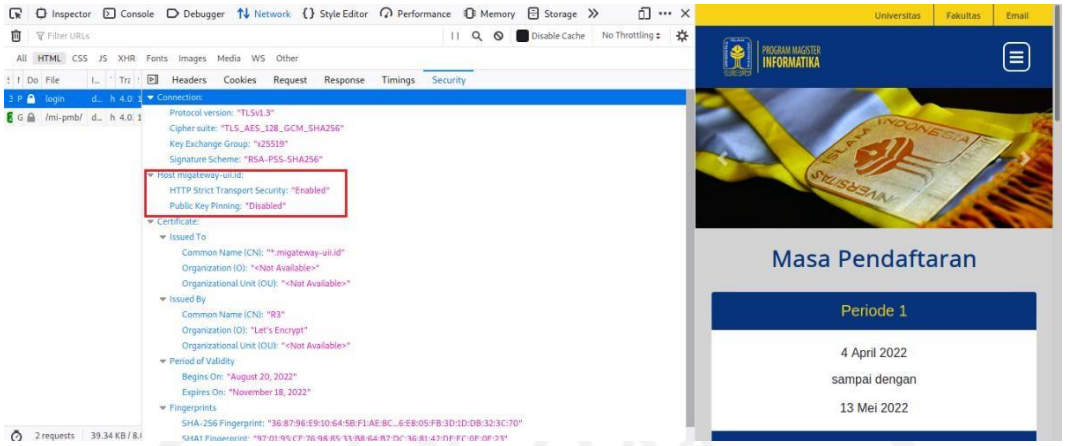
3.2.3 Pengujian Autentikasi

Berdasarkan *OWASP WSTG v*4.2 terdapat sepuluh poin pada pengujian autentikasi yang dilakukan secara *Black-Box*. Berikut merupakan langkah-langkah pengujian autentikasi berdasarkan *OWASP WSTG v*4.2:

A. WSTG-ATHN-01 Testing for Credentials Transported over an Encrypted Channels

Tujuan : memastikan penggunaan *HTTPS* pada web MI-Gateway UII.

Tools : *Browser* dan *Developer Tools.*



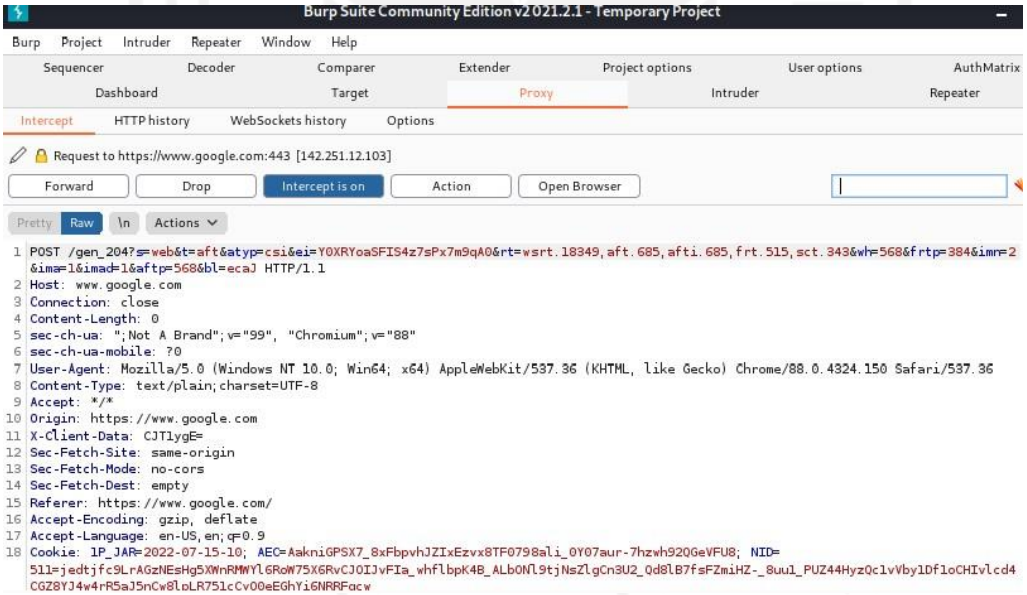
Gambar 3. 9 Tampilan *developer tools* pada *browser*

Pada **Error! Reference source not found.** proses pengujian menggunakan *developer t ools* untuk memastikan bahwa web MI-Gateway UII telah menerapkan *HTTPS*.

B. WSTG-ATHN-02 Testing for Default Credentials

Tujuan : memastikan kredensial yang digunakan oleh sistem bukan secara *default*.

Tools : *BurpSuite* dan *THC-Hydra*.



Gambar 3. 10 Proses *intercept* dengan *BurpSuite*

Pada Gambar 3. 10 dilakukan proses *intercept* untuk memastikan bahwa kredensial yang digunakan oleh web MI-Gateway bukan secara *default*.

Gambar 3. 11 Proses *rdp* dengan *THC-Hydra*

Pada Gambar 3. 11 dilakukan proses *rdp* dengan *tools THC-Hydra* untuk menemukan kredensial yang digunakan web MI-Gateway UII.

C. WSTG-ATHN-03 Testing for Weak Lock Out Mechanism

Tujuan : memastikan mekanisme penguncian akun untuk melindungi akun dari akses tidak sah dan melindungi pengguna dari penolakan akses resmi.

Tools : *Mozilla Firefox.*

Gambar 3. 12 Percobaan *login* gagal untuk memastikan penguncian akun

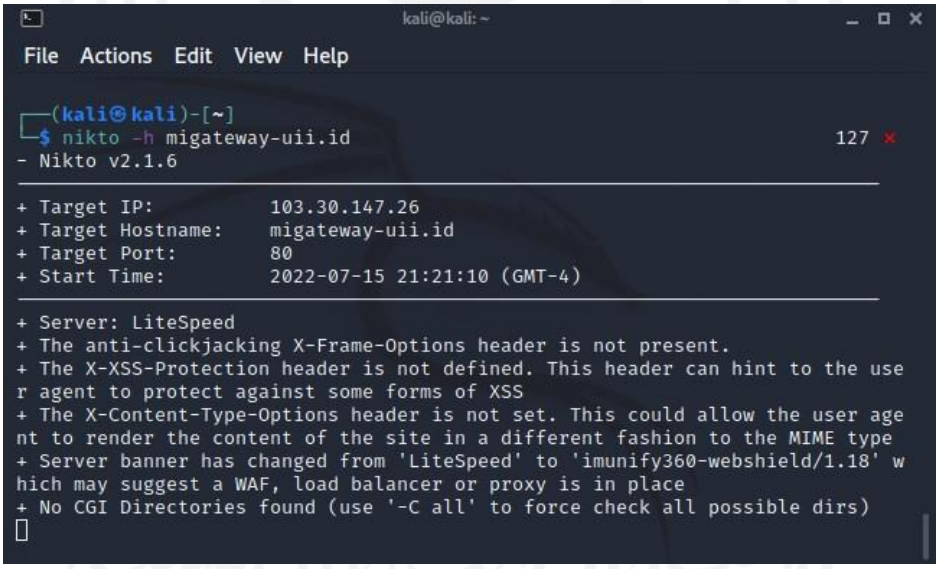
Pada Gambar 3. 12 digunakan *browser Mozilla Firefox* dalam melakukan percobaan *login* gagal untuk memastikan mekanisme penguncian akun.

D. WSTG-ATHN-04 Testing for Bypassing Authentication Schema

Tujuan : menguji *browser* dapat menyimpan informasi untuk tujuan *caching.*

Tools : *SQLMap* dan *Nikto.*

Gambar 3. 13 *Tools SQLMap* untuk menguji skema autentikasi



Gambar 3. 14 *Tools Nikto* untuk menguji skema autentikasi

Pada Gambar 3. 13 dan Gambar 3. 14 penggunaan *tools SQLMap* dan *Nikto* untuk melakukan pengujian melewati skema autentikasi melalui halaman *login*.

E. WSTG-ATHN-05 Testing for Vulnerable Remember Password

Tujuan : memastikan fitur 'Ingat Password' digunakan oleh web MI-Gateway UII.

Tools : *Mozilla Firefox.*

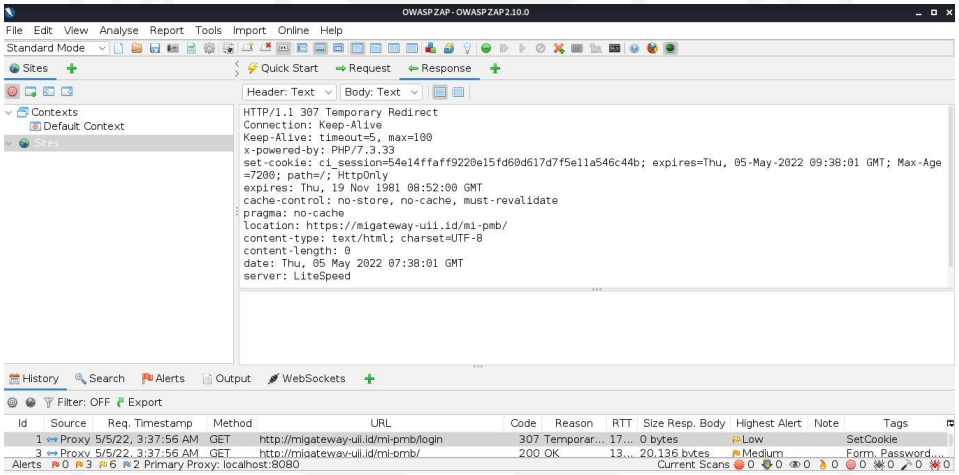
Gambar 3. 15 Fitur ‘Ingat *Password*’ tidak ditemukan

Pada Gambar 3. 15 tidak ditemukan fitur ‘Ingat *Password*’ pada web MI-Gateway UII.

F. WSTG-ATHN-06 Testing for Browser Cache Weaknesses

Tujuan : memastika *browser* dapat menyimpan informasi untuk tujuan *caching.*

Tools : *Google Chrome* dan *OWASP ZAP.*



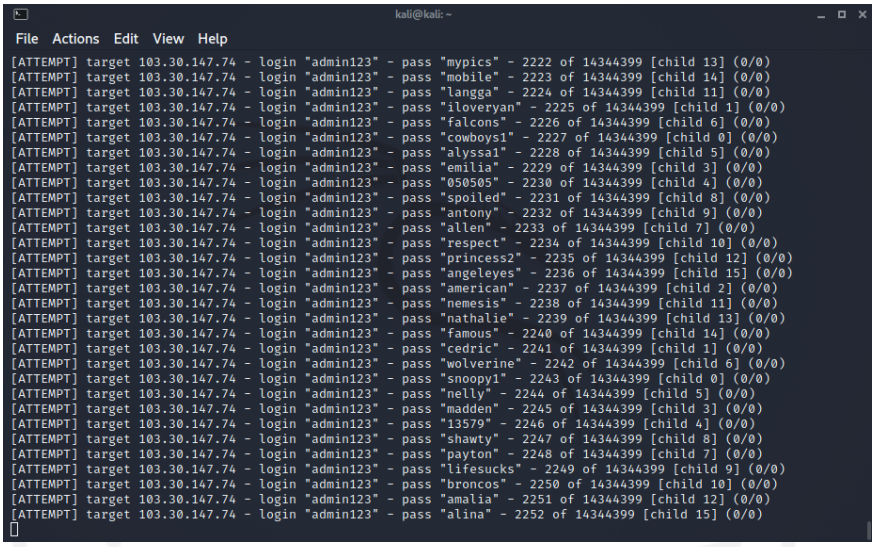
Gambar 3. 16 *Tools OWASP ZAP* untuk pencarian *caching* pada web MI-Gateway UII

Pada Gambar 3. 16 digunakan *tools OWASP ZAP* untuk melakukan *scanning* pada web MI-Gateway UII dalam proses pencarian *caching*.

G. WSTG-ATHN-07 Testing for Weak Password Policy

Tujuan : percobaan *brute force* untuk memastikan tidak digunakan kombinasi kata sandi statis yang mudah dihafal dan ditebak.

Tools : *THC-Hydra.*



Gambar 3. 17 *Brute force* dengan *Tools THC-Hydra*

Pada Gambar 3. 17 dilakukan percobaan *brute force* untuk memastikan tidak diterapkan kombinasi *username* dan *password* statis yang mudah ditebak.

H. WSTG-ATHN-08 Testing for Weak Security Question Answer

Tujuan : memastikan diterapkannya skema menjawab pertanyaan pada halaman *login* gagal.

Tools : *Mozilla Firefox*

Gambar 3. 18 Percobaan *login* gagal

Dilakukan percobaan *login* gagal untuk memastikan mekanisme menjawab pertanyaan kredensial seperti pada Gambar 3. 18.

I. WSTG-ATHN-09 Testing for Weak Password Change or Reset Functionalities

Tujuan : menguji mekanisme pengaturan ulang kata sandi

Tools : *Mozilla Firefox*

Gambar 3. 19 Tampilan ubah kata sandi

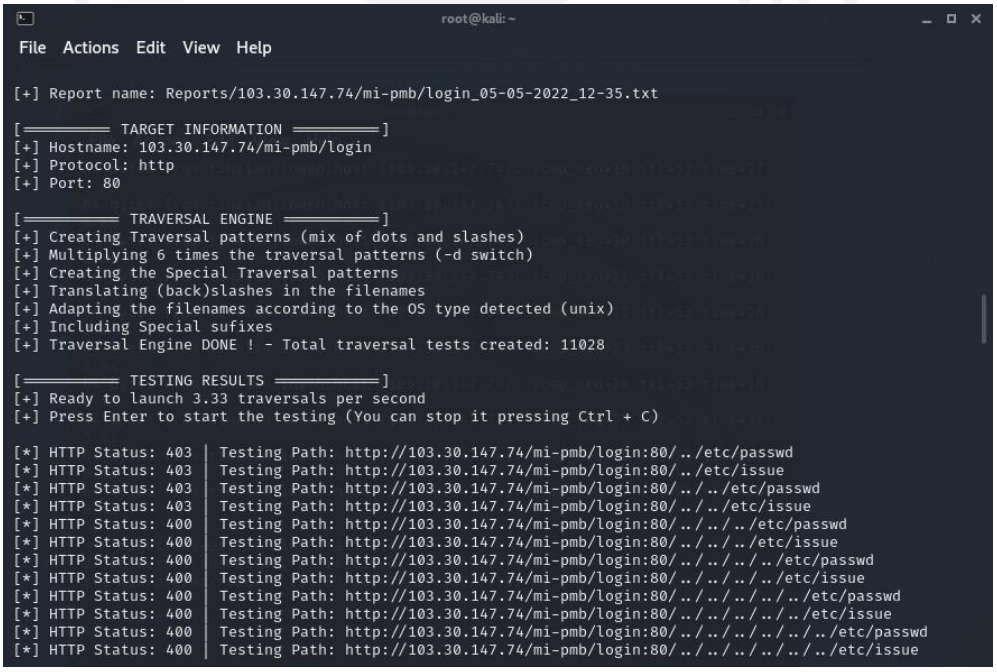
Pada Gambar 3. 19 ditampilkan halaman ubah kata sandi pada mekanisme pengujian pengaturan ulang kata sandi.

3.2.4 Pengujian Otorisasi

A. WSTG-ATHZ-01 Testing Directory Traversal File Include

Tujuan : memastikan identifikasi pengguna dapat mengakses, mengubah atau menjalankan file pada server sesuai dengan *role* nya.

Tools : *DotDotPwn.*



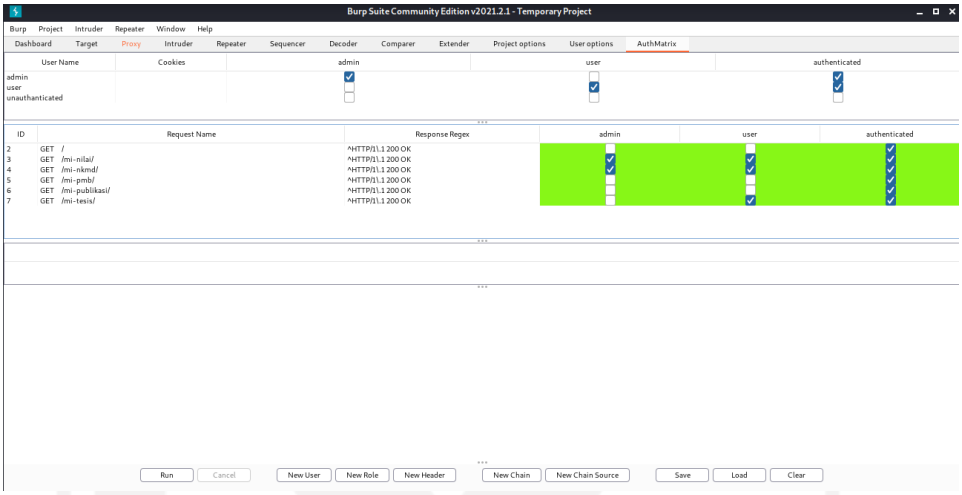
Gambar 3. 20 *Tools DotDotPwn* untuk identifikasi pengguna sesuai *role* nya.

Pada Gambar 3. 20 dilakukan proses *scanning* untuk mengidentifikasi pengguna sesuai dengan *role* nya.

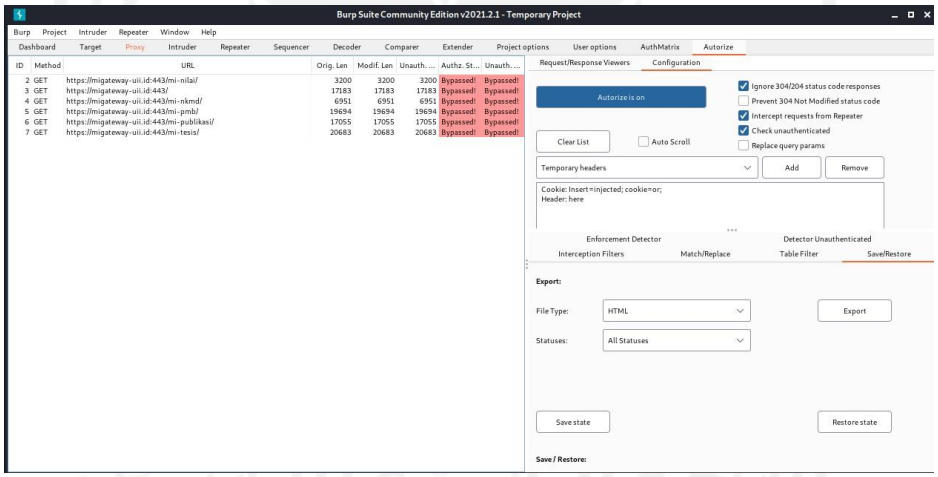
B. WSTG-ATHZ-02 Testing for Bypassing Authorization Schema

Tujuan : memverifikasi hak istimewa pengguna dalam mendapatkan akses ke fungsi dan sumber daya yang dicadangkan.

Tools : *BurpSuite* dan *AuthMatrix.*



Gambar 3. 21 *Tools BurpSuite* dalam proses *scanning* hak akses *user*



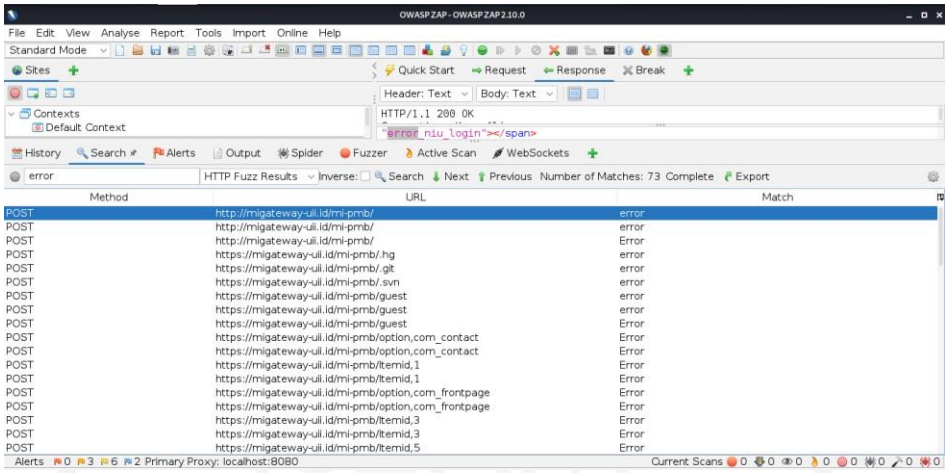
Gambar 3. 22 *AuthMatrix* ditambahkan pada *tools BurpSuite*

Pada Gambar 3. 21 dan Gambar 3. 22 *tools BurpSuite* digunakan dalam proses *scanning* hak istimewa pengguna.

C. WSTG-ATHZ-03 Testing for Privilege Escalation

Tujuan : memastikan tidak ada peningkatan hak *user* tanpa izin *administrator*.

Tools : *OWASP ZAP.*



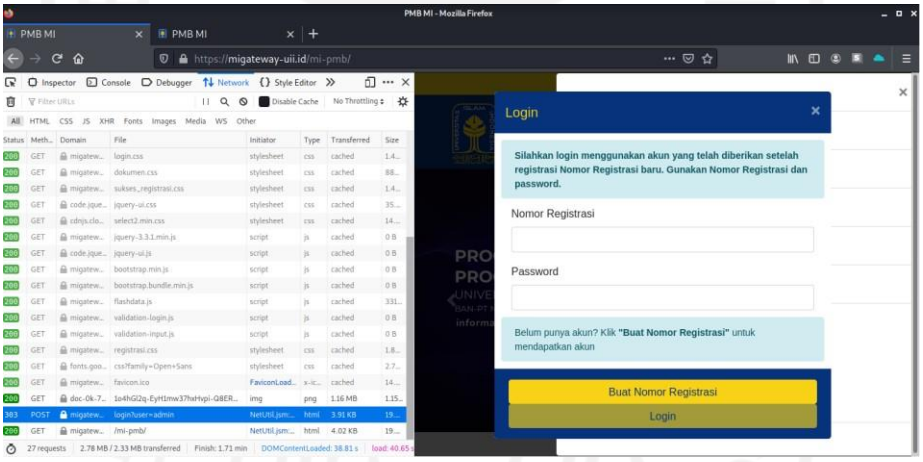
Gambar 3. 23 Metode *fuzz* digunakan pada *tools OWASP ZAP*

Penggunaan metode *fuzz* pada *tools OWASP ZAP* untuk memastikan tidak ada eskalasi hak istimewa pengguna tanpa izin administrator.

D. WSTG-ATHZ-04 Testing for Insecure Direct Object References

Tujuan : menguji *direct object references* yang tidak aman agar tidak memberi celah keamanan dengan melewati proses otorisasi.

Tools : *Developer Tools* dan *Mozilla Firefox.*



Gambar 3. 24 Halaman edit parameter dengan *developer tools* pada *browser*

Pengujian menggunakan *developer tools* pada *browser* untuk memastikan tidak terdapat celah keamanan pada proses melewati otorisasi web MI-Gateway UII.

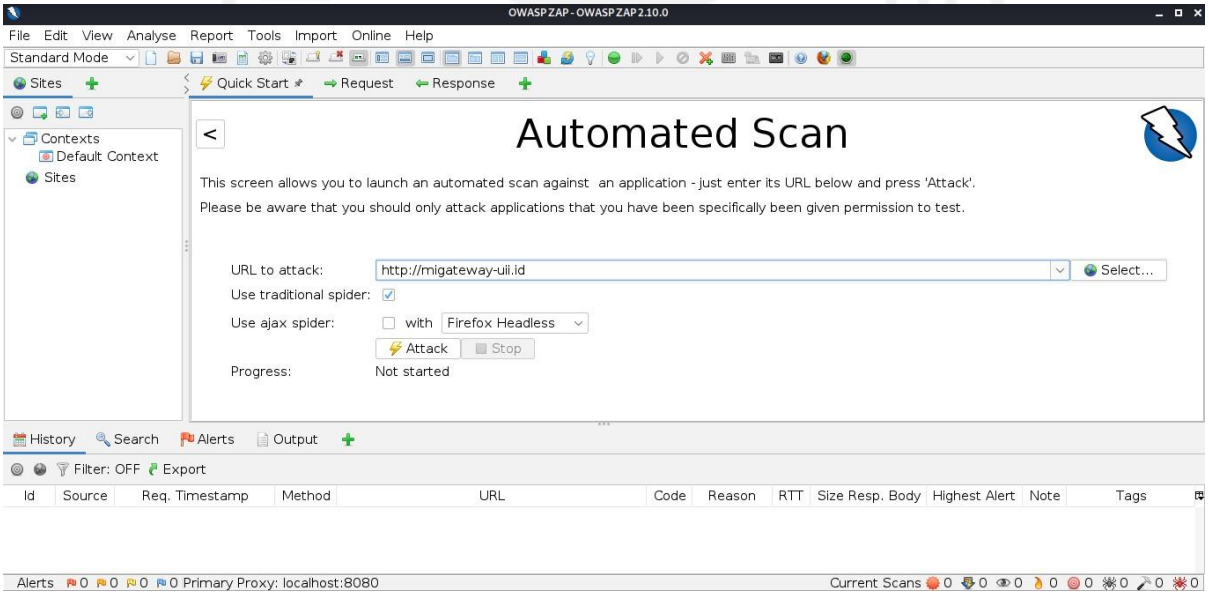
BAB IV
PENGUJIAN SISTEM DAN ANALISIS

Tahap akhir dalam suatu penelitian ini adalah melakukan penganalisisan hasil yang akan memberikan suatu keluaran penelitian. Sesuai Metodologi yang telah ditentukan sebelumnya, terdapat dua tipe kegiatan dalam Penelitian ini. Dari kedua kegiatan tersebut maka akan dianalisis menjadi sebuah *Reporting* sederhana. Setelah implementasi terselesaikan, maka diperlukannya dokumentasi keseluruhan hasil dan pembahasan terkait implementasi tersebut.

4.1 Proses

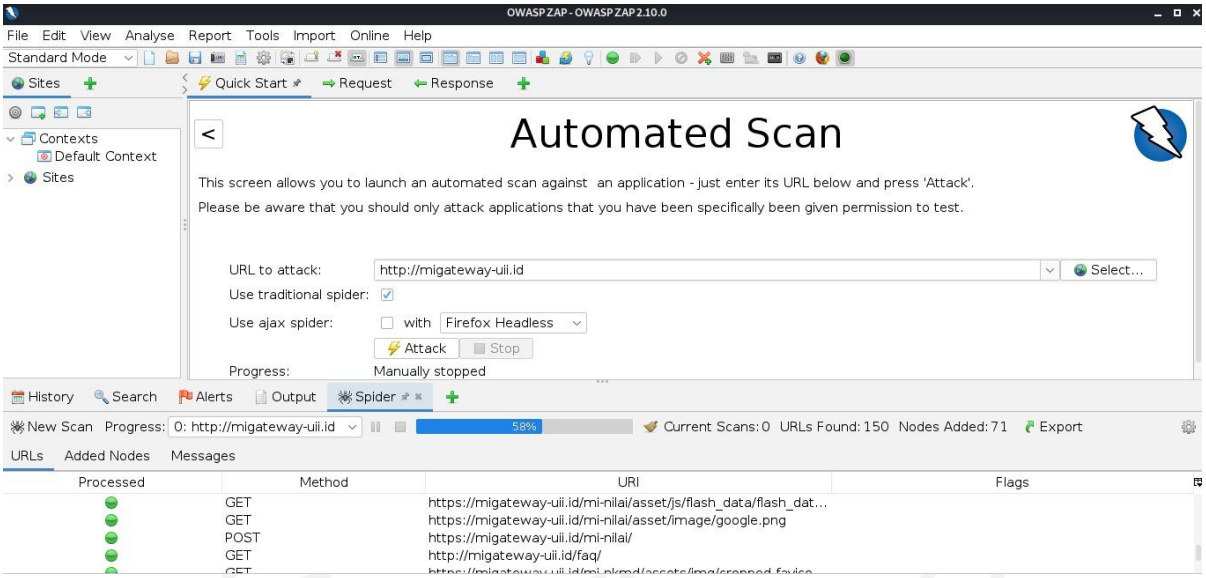
Proses *scanning* dilakukan menggunakan *tools* OWASP ZAP v2.10.0. Proses *scanning* dilakukan dengan *automated scan* serta dikombinasikan dengan *manual explore* untuk memperoleh hasil yang lebih menyeluruh. Berikut merupakan langkah melakukan *automated scan* menggunakan OWASP ZAP:

1. Pada halaman utama aplikasi OWASP ZAP, pilih *automated scan* kemudian masukkan URL target serta *browser* yang akan digunakan untuk melakukan *automated scan*. Lalu klik *Attack* seperti Gambar 4. 1.



Gambar 4. 1 Proses *scanning* otomatis dengan OWASP ZAP

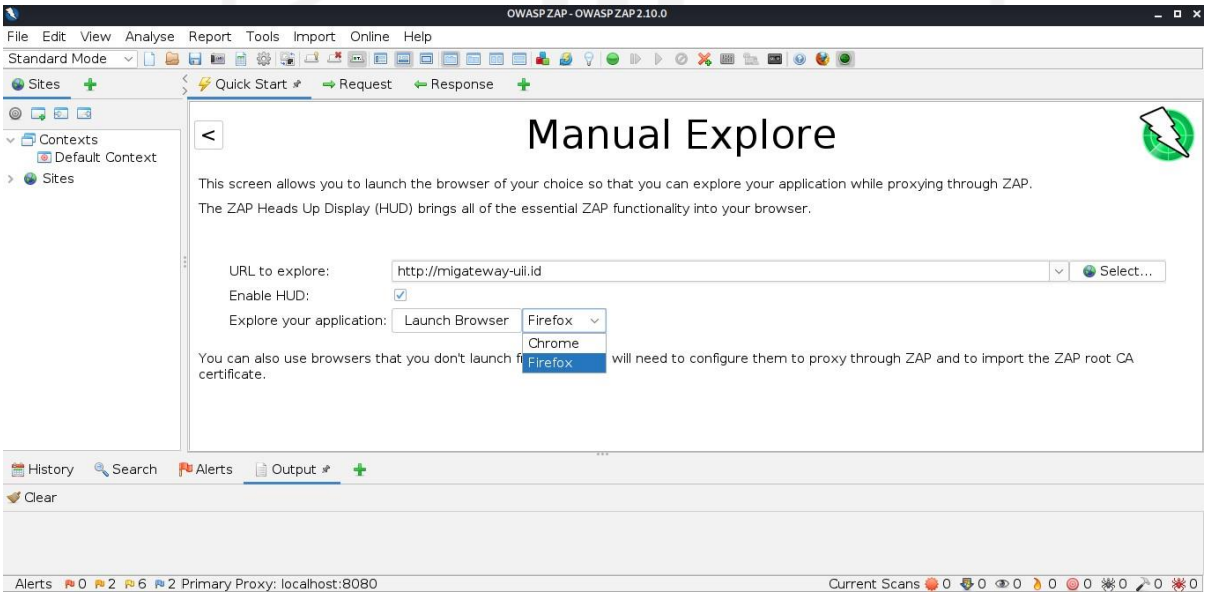
2. Tunggu beberapa saat proses *automated scan* berjalan hingga selesai. OWASP ZAP akan secara otomatis mendeteksi kerentanan yang ada pada *URL* target. Seperti terlihat pada Gambar 4. 2.



Gambar 4. 2 Tampilan proses *scanning* berjalan *automated scan* pada OWASP ZAP

Langkah dalam melakukan proses *manual explore* pada OWASP ZAP sebagai berikut:

1. Pada halaman utama aplikasi OWASP ZAP, pilih *manual explore* kemudian masukkan URL target serta browser yang akan digunakan untuk melakukan *manual explore* kemudian klik *launch browser* untuk memulai proses seperti terlihat pada Gambar 4. 3.



Gambar 4. 3 Proses *manual explore* dengan OWASP ZAP

2. Tunggu beberapa saat hingga *browser* akan muncul dan proses *manual explore* siap dilakukan. OWASP ZAP akan secara otomatis mendeteksi kerentanan disetiap *load* atau intruksi yang dijalankan terhadap halaman target seperti pada Gambar 4. 4.

Gambar 4. 4 Tampilan proses *manual explore* sedang berjalan pada *browser*

Dari proses *scanning* yang telah dilakukan menggunakan aplikasi otomatisasi *OWASP ZAP* yang menunjukan jumlah kemungkinan celah yang ada pada *web* target menurut level tingkat ancaman di sini terbagi menjadi 3 kategori berdasarkan efek yang ditimbulkan dari celah keamanan tersebut yaitu *High*, *Medium*, dan *Low*. Hasil *report* yang dikeluarkan aplikasi *OWASP ZAP* berupa format *.html* akan berupa tabel seperti Tabel 4. 1.

Tabel 4. 1 Zap *Scanning Report*

ZAP Scanning Report		
Summary of Alerts		
Risk Level	**Number of Alerts**	
High	1	
Medium	5	
Low	8	
Infomational	5	
Alerts		
Name	**Risk Level**	**Number of Instances**
Cookie Without SameSite Attribute	High	1
Directory Browsing	Medium	7
Vulnerable JS Library	Medium	20

ZAP Scanning Report		
X-Frame-Options Header Not Set	Medium	29
Absence of Anti-CSRF Tokens	Low	15
Cookie Without SameSite Attribute	Low	1
Cross-Domain JavaScript Source File Inclusion	Low	10
Incomplete or No Cache-control and Pragma HTTP Header Set	Low	46
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low	25
X-Content-Type-Options Header Missing	Low	134
Information Disclosure - Suspicious Comments	Informational	24
Timestamp Disclosure - Unix	Informational	25

Hasil *report* berisikan *risk* level celah keamanan, kategori atau nama celah, lokasi celah berada, metode, parameter dan juga solusi untuk menghadapai celah keamanan tersebut seperti yang ditunjukkan

Tabel 4. 2.

Tabel 4. 2 Detail Zap *Scanning Report*

ZAP Scanning Report	
Alert Detail	
High (Medium)	**Cookie Without SameSite Attribute**
Description	A cookie has been set without the SameSite attribute, which means that the cookie can be sent as a result of a 'cross-site' request. The SameSite attribute is an effective counter measure to cross-site request forgery, cross-site script inclusion, and timing attacks.
Solution	Ensure that the SameSite attribute is set to either 'lax' or ideally 'strict' for all cookies.
Medium (Medium)	**Directory Browsing**

ZAP Scanning Report	
Description	It is possible to view the directory listing. Directory listing may reveal hidden scripts, include files, backup source files, etc. which can be accessed to read sensitive information.
Solution	Disable directory browsing. If this is required, make sure the listed files does not induce risks.
Low (Medium)	**Absence of Anti-CSRF Tokens**
Description	No Anti-CSRF tokens were found in a HTML submission form. A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an action as the victim. The underlying cause is application functionality using predictable URL/form actions in a repeatable way. The nature of the attack is that CSRF exploits the trust that a web site has for a user. By contrast, cross-site scripting (XSS) exploits the trust that a user has for a web site. Like XSS, CSRF attacks are not necessarily cross-site, but they can be. Cross-site request forgery is also known as CSRF, XSRF, one-click attack, session riding, confused deputy, and sea surf. CSRF attacks are effective in a number of situations, including: * The victim has an active session on the target site. * The victim is authenticated via HTTP auth on the target site. * The victim is on the same local network as the target site. CSRF has primarily been used to perform an action against a target site using the victim's privileges, but recent techniques have been discovered to disclose information by gaining access to the response. The risk of information disclosure is dramatically increased when the target site is vulnerable to XSS, because XSS can be used as a platform for CSRF, allowing the attack to operate within the bounds of the same-origin policy.
Solutions	Phase: Architecture and Design Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid. For example, use anti-CSRF packages such as the OWASP CSRFGuard. Phase: Implementation Ensure that your application is free of cross-site scripting issues, because most CSRF defenses can be bypassed using attacker-controlled script. Phase: Architecture and Design

ZAP Scanning Report	
	Generate a unique nonce for each form, place the nonce into the form, and verify the nonce upon receipt of the form. Be sure that the nonce is not predictable (CWE-330). Note that this can be bypassed using XSS. Identify especially dangerous operations. When the user performs a dangerous operation, send a separate confirmation request to ensure that the user intended to perform that operation. Note that this can be bypassed using XSS. Use the ESAPI Session Management control. This control includes a component for CSRF. Do not use the GET method for any request that triggers a state change. Phase: Implementation Check the HTTP Referer header to see if the request originated from an expected page. This could break legitimate functionality, because users or proxies may have disabled sending the Referer for privacy reasons.
Infromational (Medium)	**Information Disclosure-Suspicious Comments**
Description	The response appears to contain suspicious comments which may help an attacker. Note: Matches made within script blocks or files are against the entire content not only comments.
Solutions	Remove all comments that return information that may help an attacker and fix any underlying problems they refer to.

4.2 Pembahasan

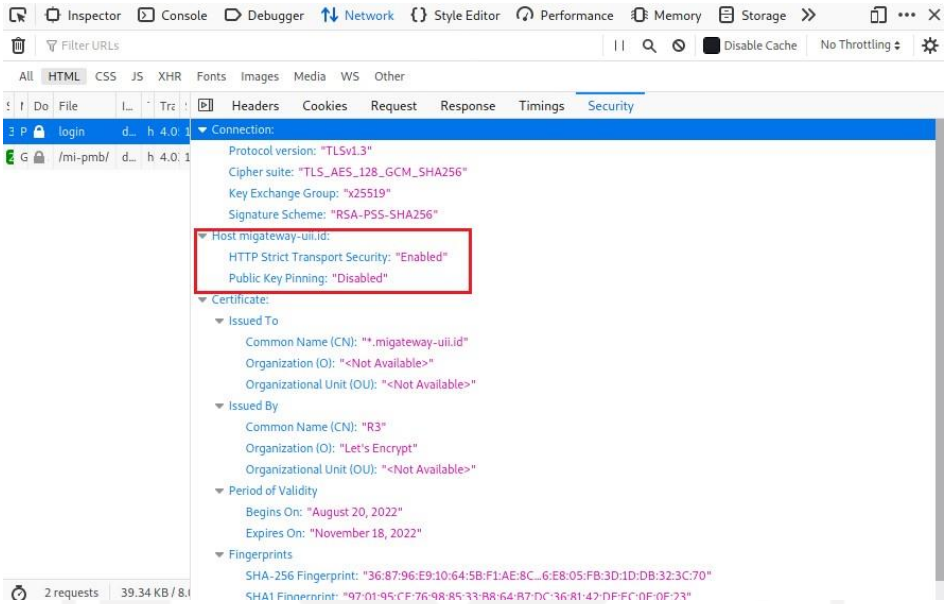
Bab ini membahas mengenai pengujian yang dilakukan kepada objek penelitian, kemudian melakukan analisis *vulnerability* yang terdapat pada *website* atau objek penelitian. Tahap pengujian *website* menggunakan dua poin penting berdasarkan pada panduan *OWASP WSTG v4.2* yaitu *authentication testing* dan *authorization testing*.

4.2.1 Authentication Testing

A. WSTG-ATHN-01 Testing for Credentials Transported over an Encrypted Channel

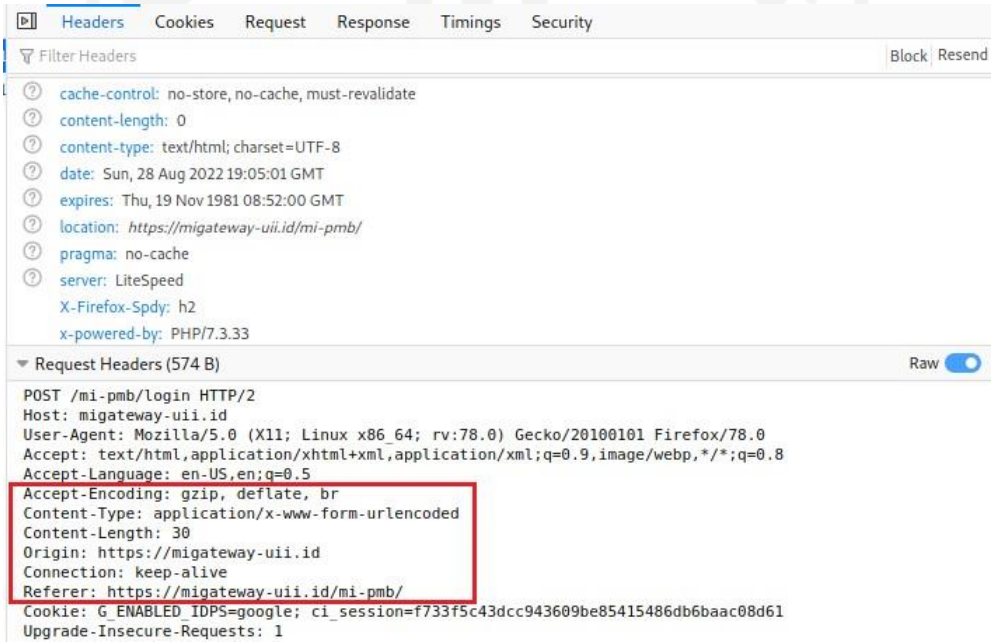
Untuk menguji transportasi kredensial, memantau lalu lintas perpindahan data antara klien dan server aplikasi web yang memerlukan kredensial. Dilakukan menggunakan web

browser dengan membuka menu *developer tools* serta mengakses halaman target pengujian pada bagian *login page.*



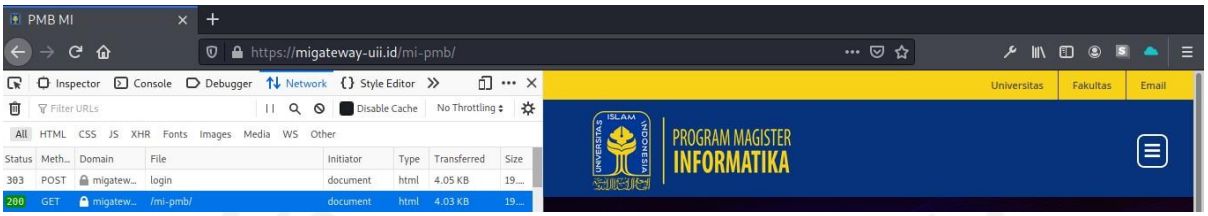
Gambar 4. 5 Hasil penerapan *HTTPS* pada *login page* dengan *developer tools*

Pada Gambar 4. 5 dapat dilihat bahwa *HTTPS* telah diterapkan dengan status *enabled* pada web MI-Gateway UII. Berdasarkan Gambar 4. 6 dapat dilihat bahwa pada *request headers* dari web MI-Gateway UII telah menerapkan *HTTPS* dalam *request* menuju *host* web MI-Gateway UII serta proses *encoding* juga telah diterapkan.



Gambar 4. 6 Tampilan *request headers* pada *login page* web MI-Gateway UII

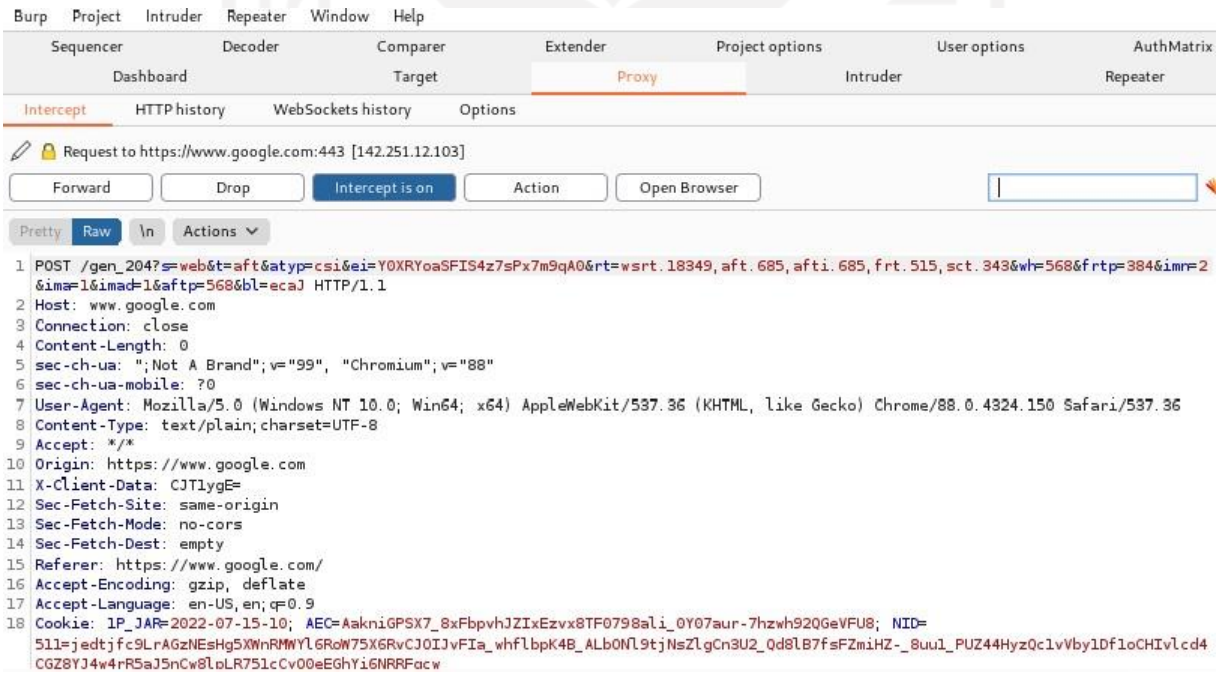
Dari hasil yang didapatkan setelah dilakukan t*esting for credentials transported over an encrypted channel* tidak ditemukan celah keamanan sebab halaman *login* telah menerapkan *HTTPS*, sedangkan web yang tidak menerapkan *HTTPS* dapat dikatakan tidak menerapkan proses enkripsi dan dapat dikatakan tidak aman (Penetration testing of Credential Data over Encrypted Channel 2022) seperti pada Gambar 4. 7.



Gambar 4. 7 Tampilan web MI-Gateway UII menerapkan *HTTPS*

B. WSTG-ATHN-02 Testing for Default Credentials

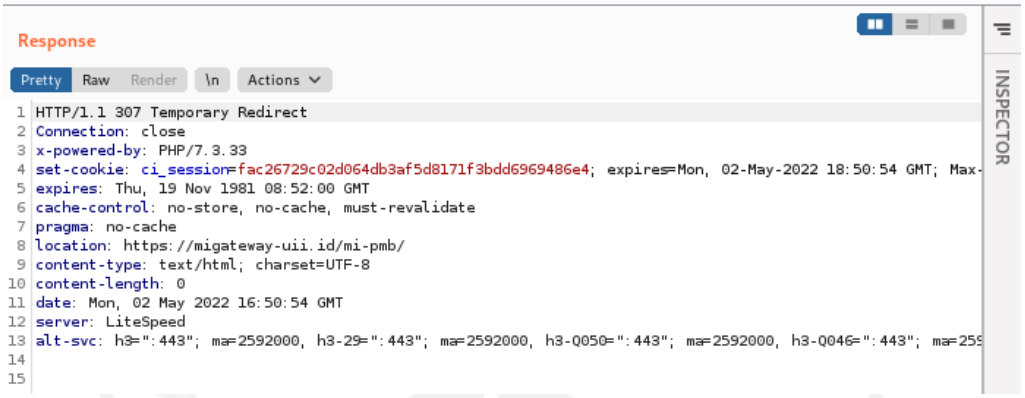
Pengujian terhadap kredensial default pada sebuah aplikasi. Dengan menerapkan metode *black-box testing* maka anggapannya penguji telah melakukan identifikasi terhadap sistem aplikasi. Implementasi *Burp-Suite* dengan mengaktifkan *intercept* dan memantau *forward* serta *drop* yang terjadi pada saat mengakses web target.



Gambar 4. 8 Proses *intercept* dengan *tools Burp Suite*

Berdasarkan prediksi kredensial secara *default* dari halaman *login* menggunakan *tools Burp-Suite* diperoleh hasil bahwa *response* yang diberikan *307 Temporary Redirect* yang

artinya pengalihan *URI* yang merespons permintaan halaman pada lokasi berbeda, namun untuk *redirect* disarankan lebih baiknya menggunakan *302 Found* karena sebagai pengalihan *get* metode yang bersifat sementara dan *HTTP* dapat berubah (302 Found - HTTP 2022) seperti pada Gambar 4. 9.



Gambar 4. 9 Tampilan halaman *response* pada *Burp Suite*

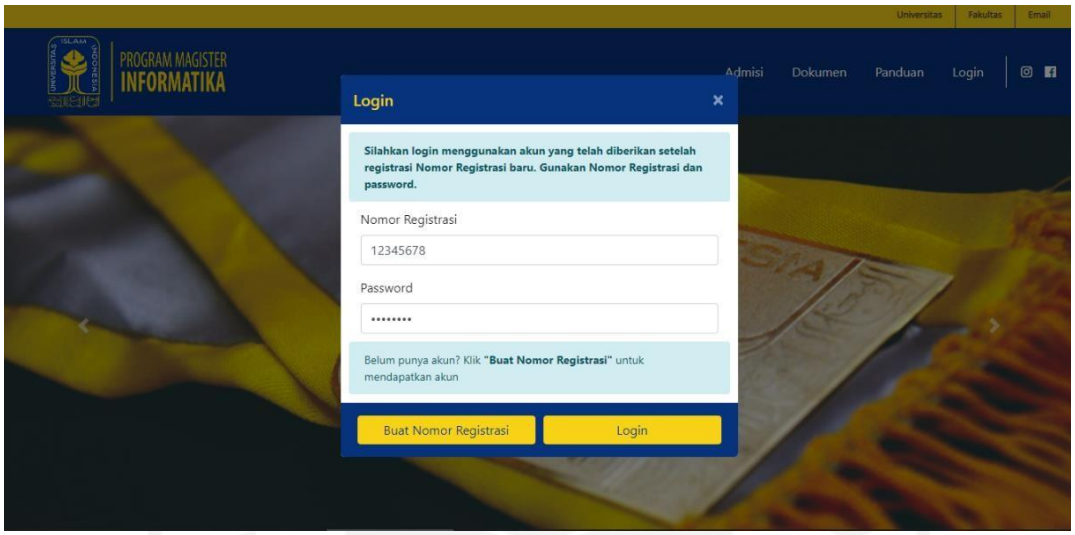
Berdasarkan prediksi kredensial yang dilakukan dengan memasukkan *username* dan *password default* menggunakan *tools THC-Hydra* dengan metode *rdp* atau memasukkan *IP Address* dalam proses *testing* pada halaman login web MI-Gateway UII tidak ditemukan *username* atau *password default* seperti pada Gambar 4. 10.

Gambar 4. 10 Metode *rdp* atau dengan *IP Address* pada *tools THC-Hydra*

Berdasarkan hasil t*esting for default credentials* dengan *tools BurpSuite* dan *THC-Hydra* melalui metode *intercept* ditemukan hasil penerapan *307 redirects*, sedangkan metode *rdp* diperoleh hasil bahwa tidak terdapat *username* dan *password* secara *default* sehingga dalam pengujian ini dapat disimpulkan tidak memiliki celah keamanan.

C. WSTG-ATHN-03 Testing for Weak Lock Out Mechanism

Dengan pengujian mekanisme penguncian maka dilakukan percobaan dengan memasukkan *username* dan *password* yang salah sebanyak 3 – 5 kali percobaan. Percobaan dilakukan pada menu *login* menggunakan metode *username* & *password* seperti Gambar 4. 11.

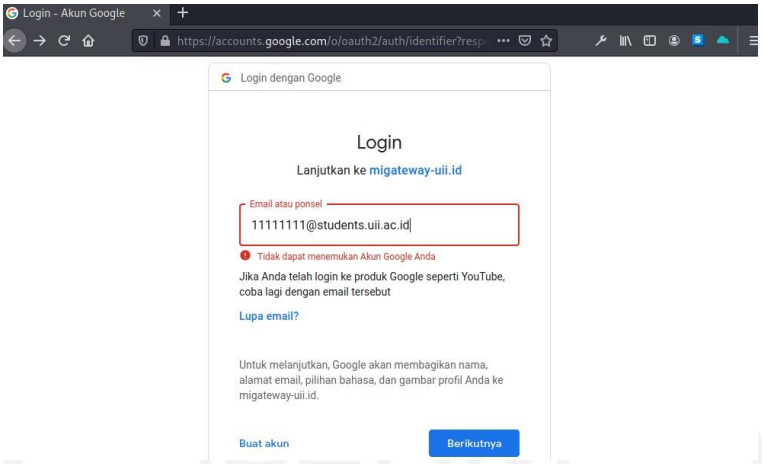


Gambar 4. 11 Tampilan halaman login

Namun, tidak diterapkannya metode *lock out mechanism* atau sering dijumpai seperti gambar untuk verifikasi akun pengguna yang terkunci. Hanya muncul keterangan *username* atau *password* salah seperti yang dapat dilihat pada Gambar 4. 12.

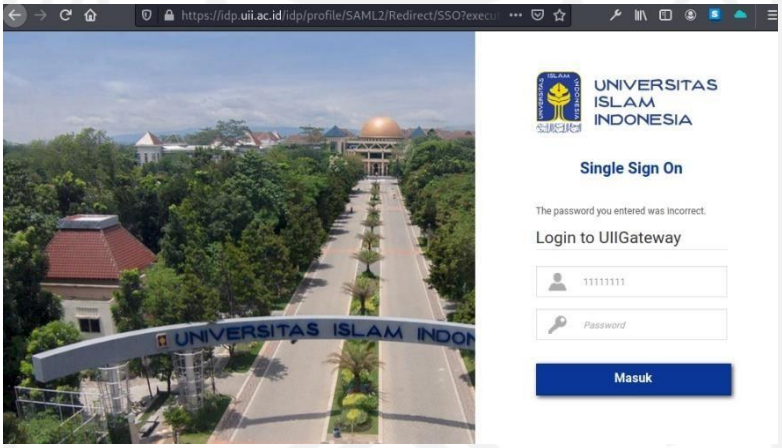
Gambar 4. 12 Tampilan percobaan dengan *username* & *password* yang salah

Kemudian percobaan dilakukan pada menu *login* menggunakan metode *OAuth* atau dengan *email google UII*. Dapat dilihat pada Gambar 4. 13.



Gambar 4. 13 Tampilan percobaan *login* dengan *email* UII

Setelah dilakukan percobaan *login* sebanyak tiga sampai lima kali tidak ditemukan adanya penerapan *lock out mechanism*. Selanjutnya percobaan *login* dilakukan pada menu *login* menggunakan metode SSO (*Single Sign On*) dapat dilihat pada Gambar 4. 14.



Gambar 4. 14 Tampilan percobaan *login* dengan *SSO*

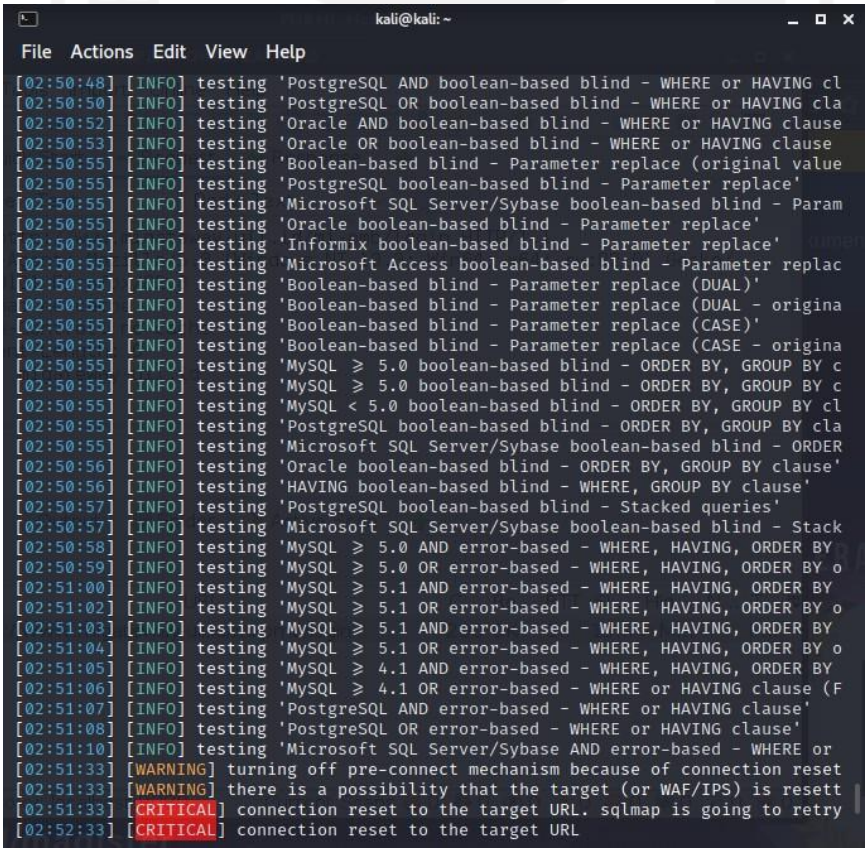
Dengan melakukan percobaan *login* sebanyak tiga sampai lima kali pada menu *login* SSO (*Single Sign On*) tidak ditemukan penerapan dari *lock out mechanism*. Dapat disimpulkan bahwa tidak diterapkannya *lock out mechanism* berarti terdapat celah pada poin pengujian ini. Dengan percobaan paksa kombinasi antara *username* dan *password* akan memberi dampak *broken authentication*.

D. WSTG-ATHN-04 Testing for Bypassing Authentication Schema

Pengujian melewati skema autentikasi dilakukan menggunakan *tools sqlmap* dan *nikto.* Beberapa percobaan dilakukan dengan melakukan *scanning* serta *injecting*.

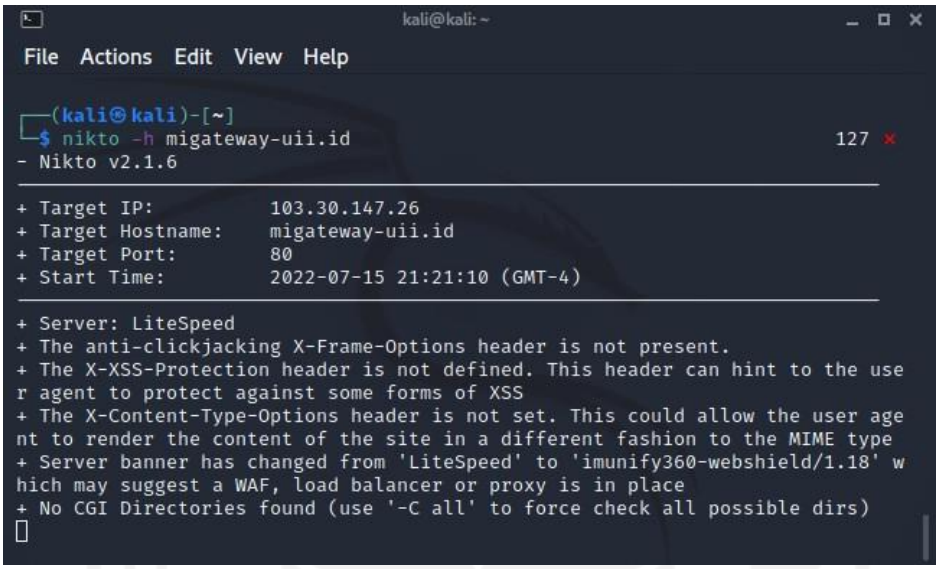
Gambar 4. 15 Tampilan *tools sqlmap*

Pada *sqlmap* kemudian dilakukan proses *scanning* berdasarkan *URL* target yaitu migateway-uii.id. Proses akan berjalan seperti pada Gambar 4. 16.



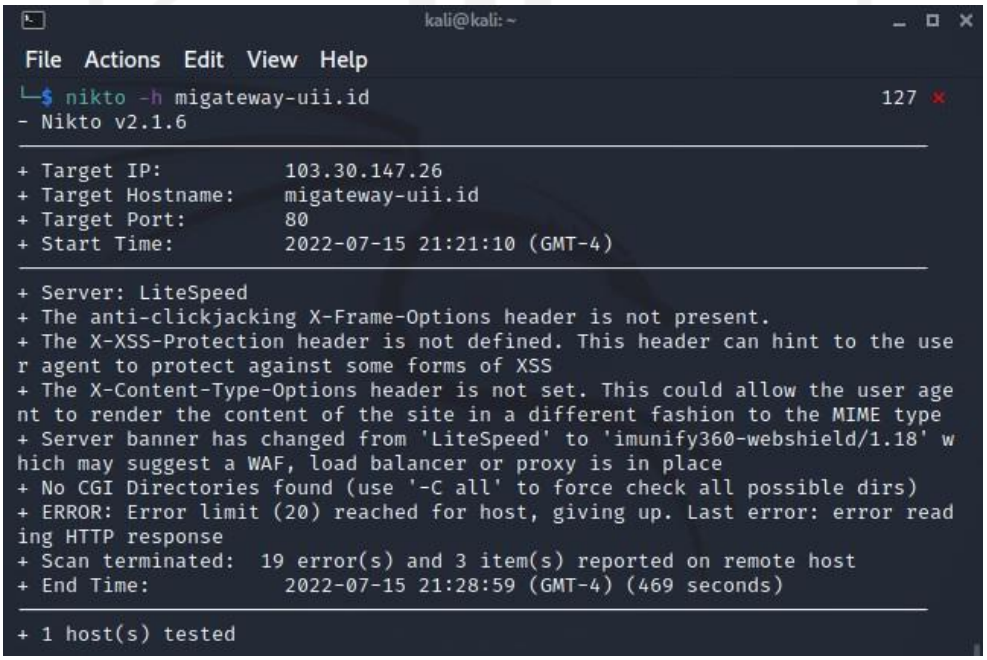
Gambar 4. 16 Hasil pengujian skema autentikasi dengan *tools SQLMap*

Berdasarkan pengujian skema metode autentikasi yang diterapkan oleh *website*, menggunakan *tools SQLMap* mendeteksi adanya penggunaan *WAF (Web Application Firewall)/IPS (Intrusion Prevention System). WAF Modsecurity* berguna untuk perlindungan terhadap *HTTP header* secara penuh dalam memproteksi *http* dan sebagai sebagai perlindungan web secara *real-time* (Bangkit Wiguna, Adi Prabowo, and Ananda 2020).



Gambar 4. 17 Tampilan proses *scanning* dengan *tools nikto*

Pengujian dilanjutkan menggunakan *tools nikto*, dengan memasukkan *URL* target dan proses *scanning* akan berjalan untuk mendeteksi skema autentikasi seperti pada Gambar 4. 18.



Gambar 4. 18 Hasil *scanning* dengan *tools nikto*

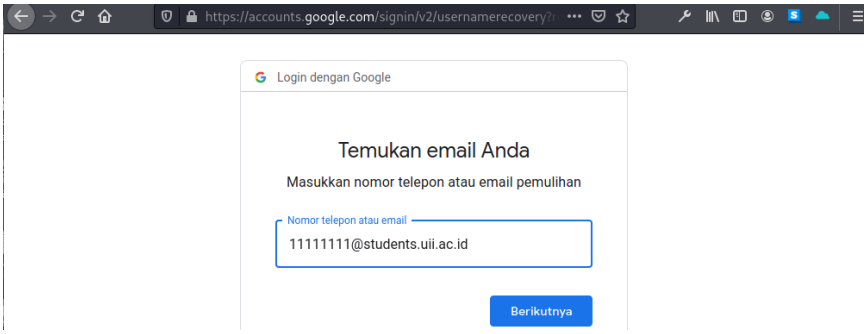
Setelah pengujian dilakukan menggunakan *tools nikto* yang memberikan hasil dari proses *scanning*. Maka dapat disimpulkan bahwa terdapat celah pada pengujian ini karena *website* belum mengimplementasikan beberapa *header* yang berfungsi dalam melindungi *content* pada sistem aplikasi. Namun, web MI-Gateway UII telah mengimplementasikan *WAF Modsecurity* untuk memberikan perlindungan terhadap serangan melalui *HTTP*. Dengan diterapkannya *WAF* mampu meminimalisir serangan terhadap *SQL Injection* karena sistem akan membatasi dalam eksekusi *SQL Injection* pada web MI-Gateway UII.

E. WSTG-ATHN-05 Testing for Vulnerable Remember Password

Pengujian dengan memperhatikan celah pada fungsi ingat *password*. Dengan fitur ingat *password* atau tetap *login* akan memberikan *cache* pada *browser* yang digunakan

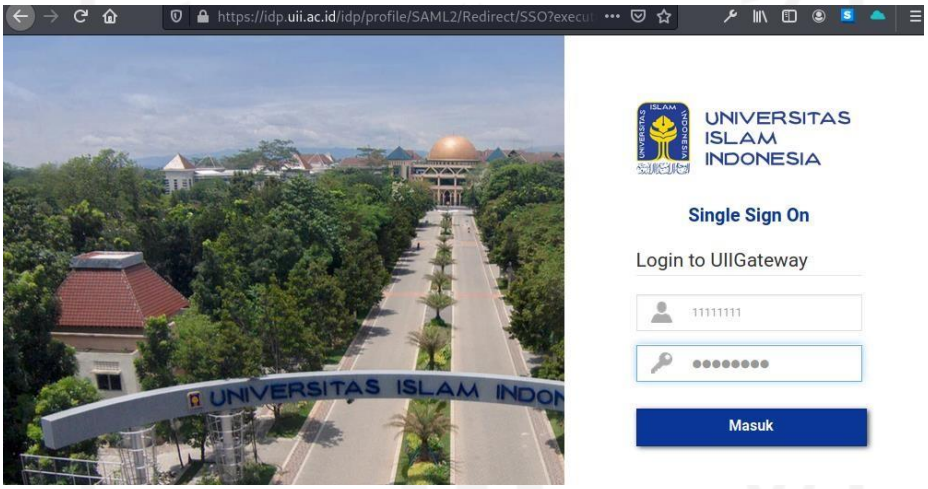
Gambar 4. 19 Halaman *login* tidak menerapkan ingat *password* ataupun tetap *login*

Pengujian menggunakan *browser mozilla firefox* seperti pada Gambar 4. 19. Dapat dilihat bahwa menu *login* menggunakan metode *username* dan *password* tidak menerapkan fitur ingat *password* atau tetap *login*.



Gambar 4. 20 Halaman *login email* UII tidak menerapkan ingat *password* atau tetap *login*

Dilakukan percobaan *login* untuk memperoleh hasil bahwa halaman *login* menggunakan *email* UII tidak menerapkan ingat *password* atau tetap *login* dapat dilihat pada Gambar 4. 20.



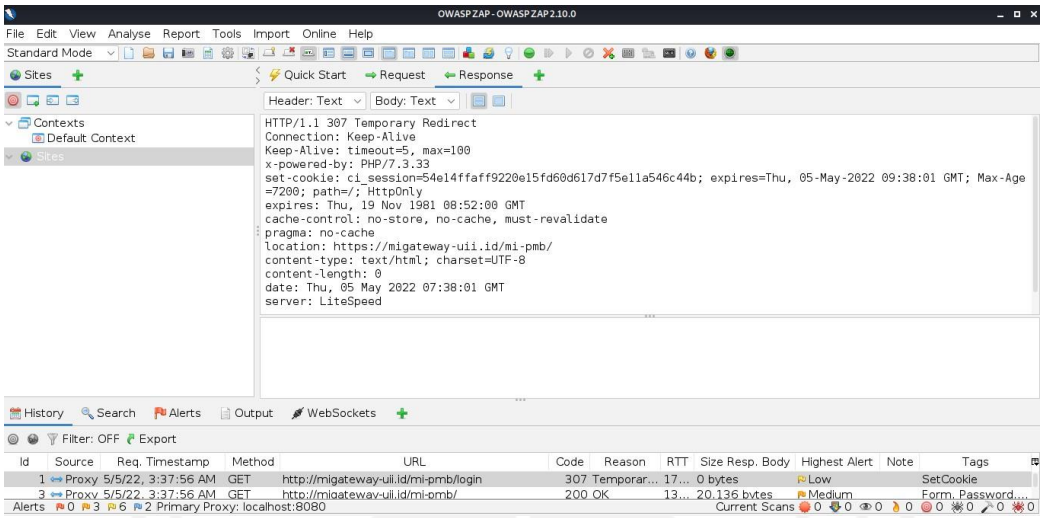
Gambar 4. 21 Halaman *login* SSO tidak menerapkan ingat *password* atau tetap *login*

Dilakukan percobaan *login* menggunakan metode SSO (*Single Sign On*) tidak ditemukan fitur ingat *password* atau tetap *login* dapat dilihat pada Gambar 4. 21. Dalam pengujian ingat *password* atau tetap *login*, dikatakan tidak terdapat celah pada pengujian ini. Dengan tidak diterapkannya fitur ingat *password* atau tetap *login*, akan meminimalkan penyimpanan pada *cache browser*. Apabila tetap *login* atau ingat *password* diterapkan akan menyimpan data pada *cache browser* yang memiliki celah untuk diretas jika tidak mengaktifkan *antivirus* yang dapat menghapus *cache browser* secara otomatis untuk menghindari pencurian data.

F. WSTG-ATHN-06 Testing for Browser Cache Weaknesses

Berdasarkan pengujian yang dilakukan menggunakan *tools OWASP ZAP* tidak ditemukan adanya *cache* atau *no-cache*, dapat diartikan bahwa *website* tidak menyimpan informasi penting pada *cache* seperti Gambar 4. 22. Selain itu, pada saat memeriksa dari tombol

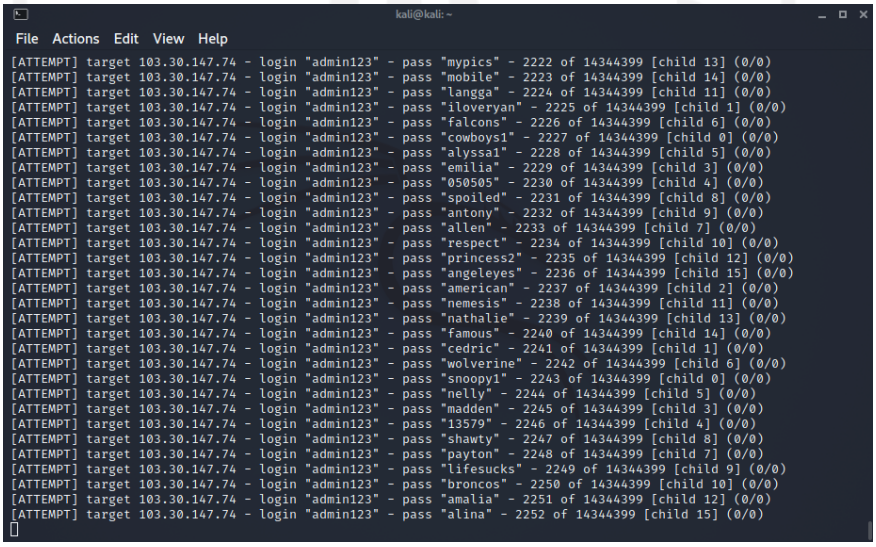
back menggunakan *tools Google Chrome* sumber daya tidak dapat ditampilkan dan diharuskan melalui skema autentikasi atau *login* kembali sehingga dikatakan tidak terdapat celah pada pengujian ini.



Gambar 4. 22 Hasil *OWASP ZAP no-cache browser*

G. WSTG-ATHN-07 Testing for Weak Password Policy

Dalam pengujian kebijakan sandi lemah atau *weak password policy* menggunakan *tools THC-Hydra* dengan melakukan *brute force* tidak ditemukan *password* yang sesuai. Selain itu, pengujian ini membutuhkan waktu yang lama. *Firewall* yang terdapat pada *website* juga dapat mengidentifikasi serta memblokir aktifitas yang mencurigakan. Oleh sebab itu, dapat disimpulkan bahwa tidak terdapat celah pada pengujian ini.



Gambar 4. 23 Hasil *brute force* dengan *tools THC-Hydra*

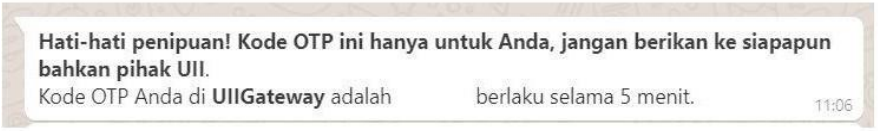
H. WSTG-ATHN-08 Testing for Weak Security Question Answer

Pengujian ini melakukan percobaan dengan menjawab pertanyaan keamanan yang lemah dengan memeriksa skema pertanyaan. Namun, pada web MI-Gateway UII diperoleh hasil bahwa *website* tidak menerapkan skema pertanyaan keamanan sehingga disimpulkan bahwa terdapat celah pada pengujian ini.

I. WSTG-ATHN-09 Testing for Weak Password Change or Reset Functionalities

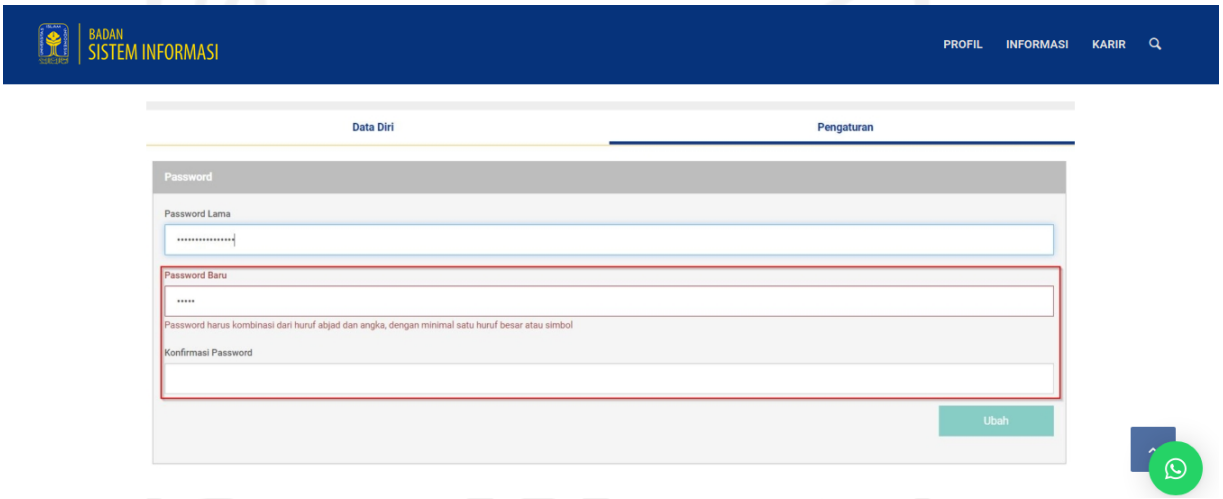
Pada pengujian ini dimaksudkan perubahan sandi atau *reset* sandi yang lemah dengan mekanisme perubahan sandi dilakukan pemeriksaan perubahan atau pengaturan ulang kata sandi. Halaman *reset password* melalui *website bsi-uii.ac.id,* kemudian mengisi *NIM* seperti Gambar 4. 24. Akan dikirim nomer OTP sebagai verifikasi bahwa pemilik akun sebenarnya. Selanjutnya, *verifikasi* ulang *password* lama dan *password* baru untuk perubahan *password* seperti pada Gambar 4. 27.

Gambar 4. 24 Halaman *reset password* pada *login SSO*



Gambar 4. 25 Halaman pesan *OTP* yang dikirimkan via *whatsapp*

Gambar 4. 26 Halaman verifikasi kode *OTP*



Gambar 4. 27 Halaman *verifikasi ulang password* pada *SSO*

Namun, pada web MI-Gateway UII diperoleh hasil bahwa web tidak menerapkan skema perubahan sandi atau *reset* sandi yang lemah. Perubahan sandi dapat dilakukan melalui *SSO (Single Sign On)* dan perubahan sandi ini menggunakan kode *OTP*. Jadi, disimpulkan bahwa tidak terdapat celah pada pengujian ini.

J. WSTG-ATHN-10 Testing for Weaker Authentication in Alternative Channel

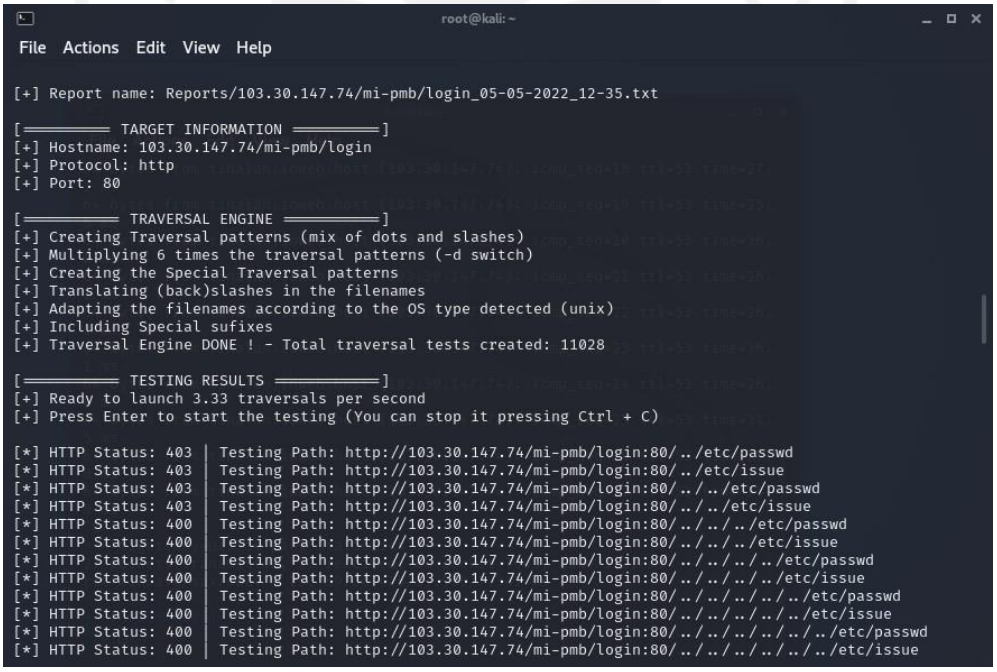
Pada pengujian ini web MI-Gateway UII sebagai layanan yang disediakan oleh BSI UII menerapkan *SSO (Single Sign On)* sebagai fungsi dari *login* yang melakukan *redirect* URL. Jadi, disimpulkan bahwa tidak terdapat celah pada pengujian ini.

Gambar 4. 28 Halaman *SSO (Single Sign On)*

4.2.2 Authorization Testing

A. WSTG-ATHZ-01 Testing Directory Traversal File Include

Berdasarkan pengujian yang dilakukan menggunakan *tools DotDotPwn* dengan melakukan proses *scanning* melalui *HTTP* tidak ditemukan injeksi yang berhubungan dengan *path traversal* seperti pada Gambar 4. 29.

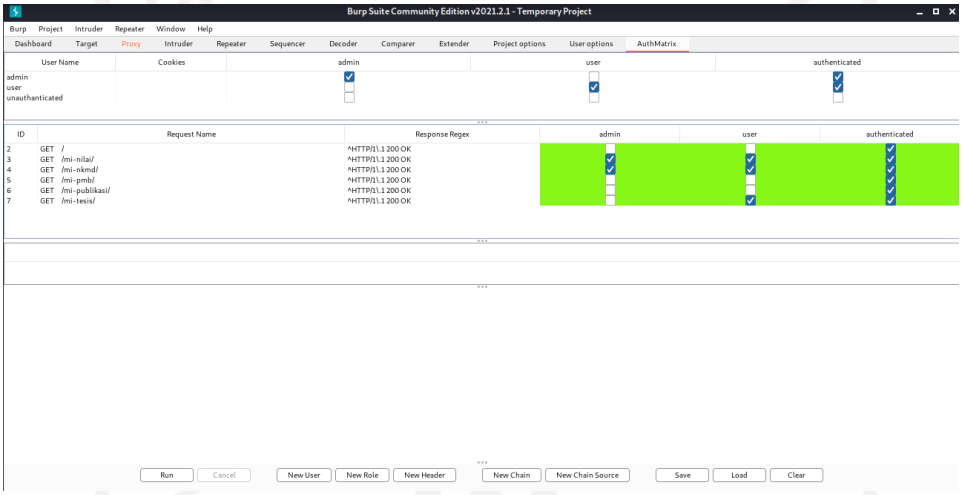


Gambar 4. 29 Hasil *scanning* dengan *tools DotDotPwn*

DotDotPwn akan memberikan pesan '*VULNERABLE!'* jika terdapat celah pada *path traversal* dari web (Sharma 2020). Setelah dilakukan pengujian *testing directory traversal file include* menggunakan *tools DotDotPwn* tidak ditemukan pesan kerentanan. Oleh karena itu, pengujian *directory traversal file include* tidak terdapat celah keamanan.

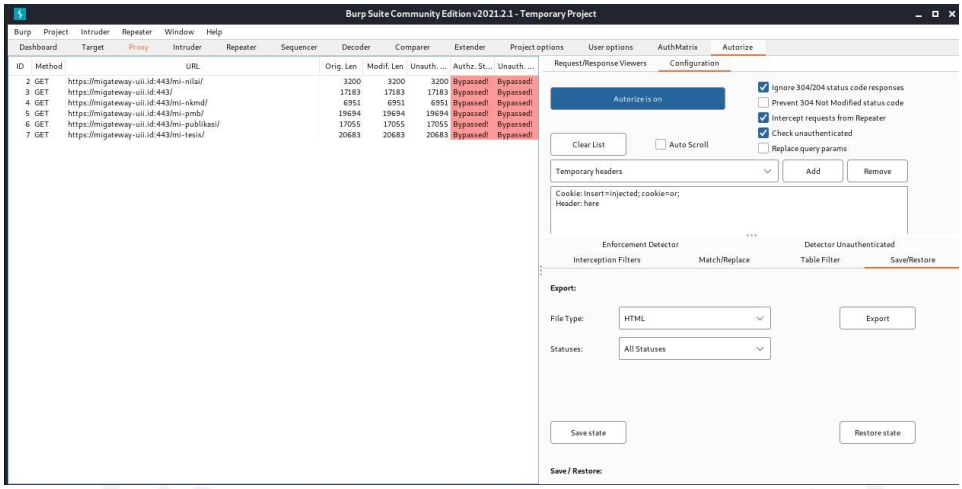
B. WSTG-ATHZ-02 Testing for Bypassing Authorization Schema

Pengujian untuk melewati skema otorisasi dengan menggunakan *tools BurpSuite* ditambahkan *add-on AuthMatrix* untuk verifikasi apakah skema otorisasi telah diterapkan pada setiap *role*-nya. Dengan pengujian yang dibagi kedalam 3 *role* yaitu *admin*, *user* dan *unauthenticated* kemudian *request* yang dijalankan adalah migateway/mi-nilai/, migateway/mi-nkmd/, migateway/mi-pmb/, migateway/mi-publikasi/, migateway/mi-tesis/. Diperoleh hasil pengujian pada Gambar 4.19 dengan kolom sel berwarna hijau yang mengindikasikan tidak ada pelanggaran yang terjadi.



Gambar 4. 30 Halaman *tools AuthMatrix* untuk *testing bypassing schema*

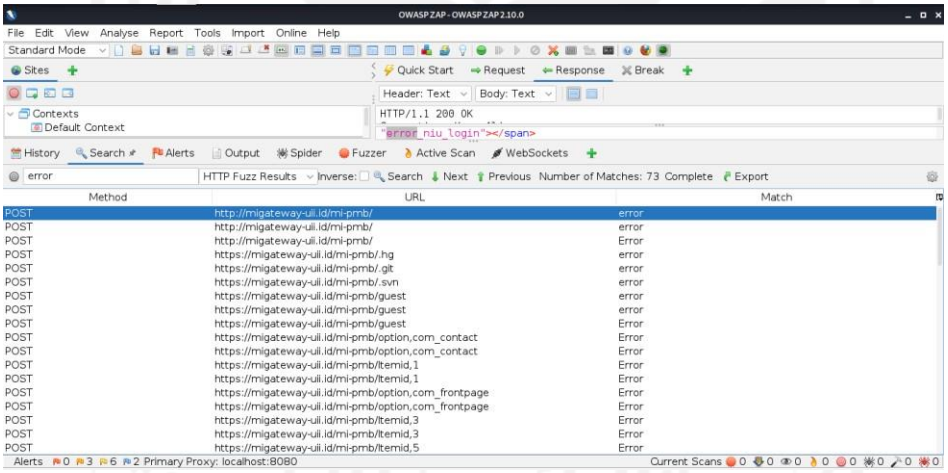
Dalam pengujian melewati skema otorisasi juga menggunakan *tools BurpSuite* ditambahkan *add-on Autorize* seperti pada Gambar 4.18. Diperoleh hasil bahwa semua poin pengujian *bypassing* terlewati atau tidak ada pelanggaran yang terjadi. Dapat disimpulkan dalam pengujian ini tidak terdapat celah keamanan.



Gambar 4. 31 Halaman *tools Autorize* dalam pengujian *bypassing scheme*

C. WSTG-ATHZ-03 Testing for Privilege Escalation

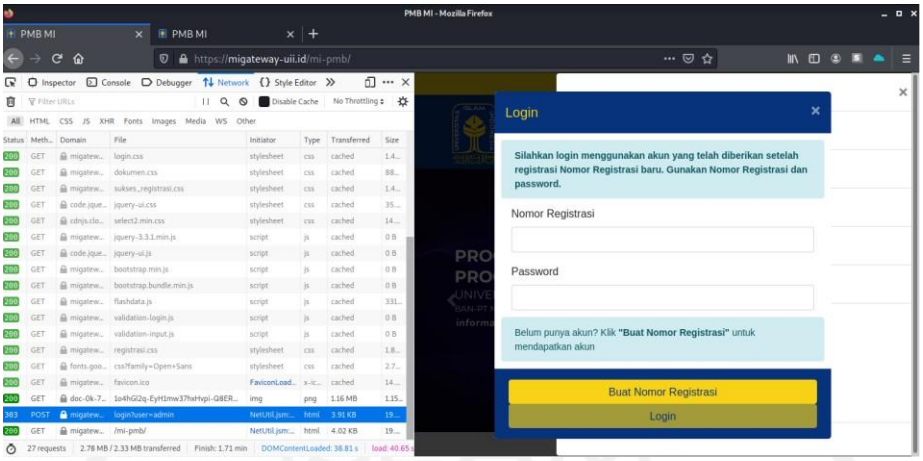
Dalam pengujian eskalasi hak istimewa menggunakan *tools OWASP ZAP* yang melakukan injeksi dengan metode *Fuzz*. Pada pengujian ini, *tools OWASP ZAP* memberikan respon *error* sehingga proses injeksi tidak dapat diselesaikan, maka disimpulkan tidak terdapat celah keamanan pada pengujian ini seperti pada



Gambar 4. 32 Halaman *tools OWASP ZAP* dengan metode *Fuzz*

D. WSTG-ATHZ-04 Testing for Insecure Direct Object References

Dalam pengujian referensi objek langsung menggunakan *tools Mozilla Firefox* melakukan modifikasi *URL* untuk melakukan *direct* kepada referensi objek. Penguji melakukan percobaan pembuatan jadwal seleksi pada *website* migateway-uii.id/mi-pmb. Namun, pengujian ini dikatakan tidak berhasil karena memerlukan *login* terlebih dahulu dan disimpulkan bahwa tidak terdapat celah keamanan pada pengujian ini.



Gambar 4. 33 Halaman *edit* parameter *URL* menggunakan *tools Mozilla Firefox*

4.3 Analisis

4.3.1 Metode OWASP WSTG v4.2

Setelah dilakukannya pengujian, kemedian dilakukan pemetaan dari hasil yang diperoleh dalam bentuk tabel seperti yang tertera pada Tabel 4. 3. Selanjutnya dilakukan pembahasan sesuai dengan standar keamanan OWASP WSTG v4.2.

Tabel 4. 3 Hasil Pengujian menggunakan metode *OWASP WSTG v4.2*

Kategori Pengujian	**Tahapan Pengujian**	**Aktivitas**	***Tools***	**Status**	**Hasil**
Authentication Testing	*Testing for Credentials Transported over an Encrypted Channel (WSTG-ATHN-01)*	Memastikan data kredensial terenkripsi dengan menggunakan *HTTPS* dalam komunikasi antara klien ke server.	• *Mozilla Firefox*	Tidak ditemukan	Telah menerapkan *HTTPS*
	Testing for Default Credentials (WSTG-ATHN-02)	Memprediksi kredensial secara *default* dan validasi pada halaman *login*	• *BurpSuite* • *THC-Hydra*	Tidak ditemukan	Tidak ditemukan *username default* untuk *login user*
	Testing for Weak Lock Out Mechanism	Melakukan beberapa kali *login* dengan *username* dan *password* yang	• *Mozilla Firefox*	Ditemukan	Tidak ada mekanisme penguncian akun pada *user invalid login*

Kategori Pengujian	Tahapan Pengujian	Aktivitas	*Tools*	Status	Hasil
	(WSTG-ATHN-03)	salah untuk menguji mekanisme penguncian akun			
	Testing for Bypassing Authentication Schema (WSTG-ATHN-04)	Melakukan pengujian skema melewati otentikasi dengan metode *request* secara langsung atau paksa untuk menguji sudah diterapkannya autentikasi disetiap layanan.	• *SQL Map* • *Nikto*	Ditemukan	Belum diterapkan *anti-clikjacking X-Frame-Options.* Serta *X-Content-Type-Options* header juga belum diterapkan. Namun, telah diterapkan *WAF Modsecurity* untuk memproteksi *HTTP* dan sebagai *tools* kontrol *traffic* pada *website*, serta perlu dilakukan pembaruan dari sisi server.
	Testing for Vulnerable Remember Password (WSTG-ATHN-05)	Melakukan pengujian kredensial pengguna pada sesi yang dihasilkan dengan melihat *log password* yang disimpan.	• *Mozilla Firefox*	Tidak Ditemukan	Pada halaman *login* tidak diterapkan *remember password* karena *credentials* tidak disimpan di *browser storage*, kemungkinannya disimpan pada partisi lain yang justru lebih aman.
	Testing for Browser	Melakukan pengujian *cache*	• *Mozilla Firefox*	Tidak ditemukan	*Cache browser* tidak ditemukan

Kategori Pengujian	Tahapan Pengujian	Aktivitas	*Tools*	Status	Hasil
	Cache Weaknesses (WSTG-ATHN-06)	*browser* untuk menemukan informasi sensitif yang tersimpan pada sisi klien. Serta pengujian terhadap tombol "kembali" untuk melihat sumber daya yang ditampilkan pada halaman sebelumnya.	• OWASP ZAP		dan fungsi tombol *back* pada *browser* tidak melakukan simpan *log*.
	Testing for Weak Password Policy (WSTG-ATHN-07)	Pengujian untuk memeriksa ketahanan *website* terhadap beberapa masukan kata sandi dengan metode *brute force.*	• *THC-Hydra*	Tidak ditemukan	Tidak ada kata sandi yang berhasil ditemukan berdasarkan kamus kata sandi.
	Testing for Weak Security Question Answer (WSTG-ATHN-08)	Memeriksa skema pertanyaan keamanan dengan mengumpulkan kemungkinan jawaban dari serangkaian pertanyaan keamanan.	• *Mozilla Firefox*	Ditemukan	*Website* tidak menerapkan pertanyaan keamanan, perlu diterapkan sebagai keamanan tambahan di atas kata sandi utama.
	Testing for Weak Password Change or Reset Functionalities (WSTG-ATHN-09)	Pengujian untuk memeriksa mekanisme perubahan atau pengaturan ulang kata sandi.	• *Mozilla Firefox*	Tidak ditemukan	*Website* telah menerapkan *SSO (Single Sign On)* yang dilakukan oleh sistem terpusat menggunakan kode OTP

Kategori Pengujian	Tahapan Pengujian	Aktivitas	*Tools*	Status	Hasil
					dalam perubahan kata sandi.
	Testing for Weaker Authentication in Alternative Channel (WSTG-ATHN-10)	Mengidentifikasi saluran autentikasi yang lain (alternatif).	• *Mozilla Firefox*	Tidak ditemukan	*Website* menerapkan *SSO (Single Sign On)* sehingga autentikasi melalui sistem terpusat menggunakan *e-mail* universitas.
Authorization Testing	*Testing Directory Traversal File Include (WSTG-ATHZ-01)*	Mengidentifikasi lokasi file *root directory* atau *root* dokumen web.	• *DirbBuster* • *DotDotPwn*	Ditemukan	Terdapat *URL* yang terindikasi terdapat celah keamanan setelah proses *scanning* melalui *HTTPS.*
	Testing for Bypassing Authorization Schema (WSTG-ATHZ-02)	Pengujian untuk melewati skema autorisasi untuk mengakses dan mengoperasikan fungsi pada sumber daya khusus tanpa autentikasi.	• *BurpSuite*	Tidak ditemukan	Skema autorisasi telah diterapkan dengan tepat sehingga tidak ada pesan *error*.
	Testing for Privilege Escalation (WSTG-ATHZ-03)	Pengujian dengan metode injeksi *fuzz* untuk mendapatkan hasil sdengan manipulasi hak istimewa dari *user*.	• *OWASP ZAP*	Tidak ditemukan	Hak istimewa tidak dapat dimanipulasi.
	Testing for Insecure Direct Object References	Pengujian dengan modifikasi *URL website* untuk *direct request*	• *Mozilla Firefox*	Tidak ditemukan	Parameter *URL* pada halaman *log* tidak dapat dimodifikasi.

Kategori Pengujian	Tahapan Pengujian	Aktivitas	*Tools*	Status	Hasil
	(WSTG-ATHZ-04)	ke objek referensi.			

Berdasarkan Tabel 4. 3 dapat dilihat bahwa hasil pengujian *website* Mi-Gateway UII menggunakan metode *OWASP WSTG v4.2* terdapat dua kategori pengujian yaitu *authentication* dan *authorization* dengan total 14 sub kategori yang kemudian dilakukan pengujian untuk memperoleh hasil ada atau tidaknya risiko celah keamanan pada *website*. Hasil dari pengujian yang ditemukan kerentanan dipetakan ke dalam Tabel 4. 4 dimaksudkan untuk langkah analisis memberikan fokus terhadap poin-poin pengujian yang terdapat celah keamanan.

Tabel 4. 4 Pemetaan Hasil Kerentanan Pengujian

Kategori Pengujian	Tahapan Pengujian	Aktivitas	*Tools*	Status	Hasil
Authentication Testing	*Testing for Weak Lock Out Mechanism (WSTG-ATHN-03)*	Melakukan beberapa kali *login* dengan *username* dan *password* yang salah untuk menguji mekanisme penguncian akun	*Mozilla Firefox*	Ditemukan	Tidak ada mekanisme penguncian akun pada *user invalid login*
	Testing for Bypassing Authentication Schema (WSTG-ATHN-04)	Melakukan pengujian skema melewati otentikasi dengan metode *request* secara langsung atau paksa untuk menguji sudah diterapkannya autentikasi disetiap layanan.	• *SQL Map* • *Nikto*	Ditemukan	Belum diterapkan *anti-clikjacking X-Frame-Options.* Serta *X-Content-Type-Options* header juga belum diterapkan.

Kategori Pengujian	Tahapan Pengujian	Aktivitas	*Tools*	Status	Hasil
					Namun, implementasi *WAF (Web Application Firewall)* untuk memproteksi *HTTP* dan sebagai *tools* kontrol *traffic* pada *website*, serta pembaruan dari sisi server.
	Testing for Vulnerable Remember Password (WSTG-ATHN-05)	Melakukan pengujian kredensial pengguna pada sesi yang dihasilkan dengan melihat *log password* yang disimpan.	*Mozilla Firefox*	Ditemukan	Pada halaman *login* tidak diterapkan *remember password.*
	Testing for Weak Security Question Answer (WSTG-ATHN-08)	Memeriksa skema pertanyaan keamanan dengan mengumpulkan kemungkinan jawaban dari serangkaian	*Mozilla Firefox*	Ditemukan	*Website* tidak menerapkan pertanyaan keamanan, perlu diterapkan sebagai keamanan tambahan di

Kategori Pengujian	Tahapan Pengujian	Aktivitas	*Tools*	Status	Hasil
		pertanyaan keamanan.			atas kata sandi utama.
	Testing for Weak Security Question Answer (WSTG-ATHN-08)	Memeriksa skema pertanyaan keamanan dengan mengumpulkan kemungkinan jawaban dari serangkaian pertanyaan keamanan.	*Mozilla Firefox*	Ditemukan	*Website* tidak menerapkan pertanyaan keamanan, perlu diterapkan sebagai keamanan tambahan di atas kata sandi utama.
Authorization Testing	*Testing Directory Traversal File Include (WSTG-ATHZ-01)*	Mengidentifikasi lokasi file *root directory* atau *root* dokumen web.	• *DirbBuster* • *DotDotPwn*	Ditemukan	Terdapat *URL* yang terindikasi terdapat celah keamanan setelah proses *scanning* melalui *HTTPS.*

4.3.2 Hasil Scanning dengan tools OWASP ZAP

Berdasarkan pengujian yang telah dilakukan terhadap *website* Mi-Gateway UII terdapat beberapa kerentanan yang ditemukan pada proses *scanning* menggunakan *tools OWASP ZAP* terdapat pada Tabel 4. 5.

Tabel 4. 5 Kategori ancaman hasil proses *scan* menggunakan *tools OWASP ZAP*

Jenis Ancaman	Level	Jumlah
Directory Browsing	*MEDIUM*	7
Vulnerable JS Library	*MEDIUM*	20
X-Frame-Options Header Not Set	*MEDIUM*	29
Absence of Anti-CSRF Tokens	*LOW*	15

Jenis Ancaman	Level	Jumlah
Cookie Without SameSite Attribute	*LOW*	2
Cross-Domain JavaScript Source File Inclusion	*LOW*	10
Incomplete or No Cache-control and Pragma HTTP Header Set	*LOW*	46
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	*LOW*	25
X-Content-Type-Options Header Missing	*LOW*	134
Information Disclosure - Suspicious Comments	*INFORMATIONAL*	24
Timestamp Disclosure - Unix	*INFORMATIONAL*	25

4.3.3 Hasil

Dari hasil *scanning* yang telah dilakukan pada proses sebelumnya ditemukan bahwa web MI-Gateway UII terdapat beberapa celah keamanan yang dapat membahayakan keamanan web. Perlu segera dilakukan tindakan pencegahan lebih dini dan rata-rata kemungkinan celah keamanan yang ditemukan pada hasil *scanning* menggunakan aplikasi *OWASP ZAP* terdeteksi pada beberapa bagian dari *website.*

Tabel 4. 6 Solusi dari celah keamanan yang ditemukan

Celah Keamanan	Solusi
Directory Browsing	Melakukan *disable directory browsing* melalui CPanel atau dapat melakukan pemblokiran menggunakan file .htacces
Vulnerable JS Library	*Update library* menjadi versi terbaru dari bootsrap
X-Frame-Options Header Not Set	Implementasikan header HTTP X-Frame-Options pada semua konfigurasi header untuk mengembalikan halaman website, cara ini digunakan sebagian besar browser web modern

Dari Tabel 4. 6 terdapat tiga hasil dari proses *scanning* dengan *tools OWASP ZAP* pada *level medium*. *Directory Browsing* mengindikasikan bahwa data berupa informasi penting dapat diakses oleh diluar *user* maupun *administrator*. *Vulnerable JS Library* mengindikasikan bahwa *library* dari *JS* atau *JavaScript* pada kasus ini *version 4.3.0* terindikasi sebagai celah keamanan. *X-Frame-Options Header Not Set* mengindikasikan bahwa *header* ini tidak

mengimplementasikan *HTTP Response* sehingga dapat menjadi celah serangan *clickjacking*. Selain itu, dari hasil *scanning* menggunakan aplikasi *OWASP ZAP* juga terhalang oleh *firewall* pada web MI-Gateway UII yang cukup baik sehingga serangan dapat dihalau dan proses *scanning* dibatalkan.

BAB V
KESIMPULAN

5.1 Kesimpulan

Dalam melakukan uji *penetration testing* menggunakan metode *OWASP WSTG v4.2* yang bertujuan untuk menguji tingkat kerentanan pada web MI-Gateway UII yang dimiliki oleh Program Studi Informatika Program Magister Universitas Islam Indonesia berdasarkan dari seluruh kegiatan penelitian yang dilakukan, dapat diambil kesimpulan yang antara lain sebagai berikut:

a. Metode *OWASP WSTG v4.2* layak dijadikan sebagai acuan dalam melakukan uji *penetration testing* pada web MI-Gateway UII. Hal ini dapat dilihat dari hasil pengujian dan analisa yang menunjukkan bahwa web MI-Gateway UII memiliki kelemahan yang dapat dieksploitasi pada web MI-Gateway UII setelah dilakukan pengujian keamanan berdasarkan dokumen *OWASP WSTG v4.2*. Salah satu kelemahannya adalah belum diterapkan *weak-lock out mechanism* (WSTG-ATHN-03).
b. Keamanan sistem pada web MI-Gateway UII masih belum memenuhi prinsip keamanan *CIA (Confidentiality, Integrity, and Availability)*. Hal ini dapat dilihat dari beberapa celah keamanan setelah dilakukan pengujian penetrasi.
c. Web MI-Gateway UII memiliki *firewall* yang cukup bisa diandalkan dalam menanggulangi serangan-serangan dari luar maupun dari dalam sistem.
d. *Tools OWASP ZAP* sebagai alat otomatisasi diimplementasikan dan menampilkan hasil *scanning* secara rinci mengenai jenis kerentanan, kategori *level* serta diberikan deskripsi dan solusi dalam menangani kerentanan sistem.

5.2 Saran

Berdasarkan penelitian yang telah terlaksana diperoleh beberapa saran yang dapat diterapkan oleh pengembang sistem web MI-Gateway UII sebagai objek penelitian dan dapat diterapkan dalam pengembangan penelitian terkait, sebagai berikut:

a. Perlunya dilakukan pengujian secara menyeluruh berdasarkan metode *OWASP WSTG v*4.2 pada sistem web MI-Gateway UII agar segera dapat diantisipasi jika terdapat celah keamanan.

b. Perlu dilakukan *update* secara berkala terhadap beberapa *plugin* yang terdapat dalam web yang dikelola.
c. Melakukan konfigurasi pada server web agar hanya *user* tertentu yang dapat melakukan *request* terhadap data yang sensitif.
d. Perlu dilakukan *encryption* terhadap data penting untuk mengurangi risiko terjadinya kebocoran data dan informasi.
e. Perlu dilakukan pengujian keamanan sistem secara menyeluruh untuk di lakukan evaluasi terhadap risiko pencurian data dan infromasi dapat diantisipasi.
f. Perlu dilakukan perubahan pada *redirect URI* yang telah menerapkan *307 Temporary Redirect* diubah dengan *302 Found* yang mengutamakan *SEO*.

DAFTAR PUSTAKA

"302 Found - HTTP." 2022. https://developer.mozilla.org/en-US/docs/Web/HTTP/Status/302 (August 29, 2022).

Bangkit Wiguna, Wahyu Adi Prabowo, and Ridho Ananda. 2020. "Implementasi Web Application Firewall Dalam Mencegah Serangan SQL Injection Pada Website." *Digital Zone: Jurnal Teknologi Informasi dan Komunikasi* 11(2): 245–56.

Dirgahayu, Raden Teduh, Yudi Prayudi, and Adi Fajaryanto. 2016. "Penerapan Metode ISSAF Dan OWASP Versi 4 Untuk Uji Kerentanan Web Server." *Network Engineering Research Operation* 1(3). https://nero.trunojoyo.ac.id/index.php/nero/article/view/29 (July 7, 2021).

Drake, Victoria. 2020. "OWASP Web Security Testing Guide v4.2 Released." https://medium.com/@victoriadotdev/owasp-web-security-testing-guide-v4-2-released-7910ea1d7e47.

Guntoro, Guntoro, Loneli Costaner, and Musfawati Musfawati. 2020. "ANALISIS KEAMANAN WEB SERVER OPEN JOURNAL SYSTEM (OJS) MENGGUNAKAN METODE ISSAF DAN OWASP (STUDI KASUS OJS UNIVERSITAS LANCANG KUNING)." *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)* 5(1): 45.

ISBuzz. 2021. "Penetration Testing." https://informationsecuritybuzz.com/education/penetration-testing/.

"ISO - ISO/IEC 17799:2005." https://www.iso.org/standard/39612.html (July 25, 2022).

Kumari, Nidhi. 2020. "Black Box Vs White Box Testing." https://medium.com/@menidhikjha/black-box-vs-white-box-testing-ad589ca0224e (August 29, 2022).

Michael E. Whitman, Herbert J. Mattord - Google Buku. 2020. *Principles of Incident Response and Disaster Recovery*. https://books.google.co.id/books?hl=id&lr=&id=kpQMEAAAQBAJ&oi=fnd&pg=PP1&dq=principles+of+incident+response+and+disaster+recovery&ots=l4X7vClaD-&sig=BiopAmQvzAK67VSxHfrM-InksBE&redir_esc=y#v=onepage&q=principles of incident response and disaster recovery& (May 8, 2022).

Nagendran, K. et al. 2019. "Web Application Penetration Testing." *International Journal of Innovative Technology and Exploring Engineering* 8(10): 1029–35.

Nagpure, Sangeeta, and Sonal Kurkure. 2018. "Vulnerability Assessment and Penetration

Testing of Web Application." In *2017 International Conference on Computing, Communication, Control and Automation, ICCUBEA 2017,*.

Owasp. 2017. "OWASP Foundation | Open Source Foundation for Application Security." https://owasp.org/.

"Penetration Testing of Credential Data over Encrypted Channel." 2022. https://www.hackingloops.com/penetration-testing-of-credential-data-over-encrypted-channel/ (August 29, 2022).

Sharma, Harshit. 2020. "Finding Path Traversal Vulnerability." https://infosecwriteups.com/finding-path-traversal-vulnerability-e2506d390569 (August 29, 2022).

Sunardi, Imam Riadi, and Pradana Ananda Raharja. 2019. "Vulnerability Analysis of E-Voting Application Using Open Web Application Security Project (OWASP) Framework." *International Journal of Advanced Computer Science and Applications* 10(11): 135–43.

LAMPIRAN

A. Report

B. Hasil *scanning* dengan *OWASP ZAP*

REPORT

PENGUJIAN AUTENTIKASI DAN OTORISASI BERDASARKAN DOKUMEN OWASP WSTG v4.2 (STUDI KASUS WEBSITE MI-GATEWAY UII)

Disusun Oleh:

N a m a : Aditya Wibisono Kuncoro

NIM : 18523217

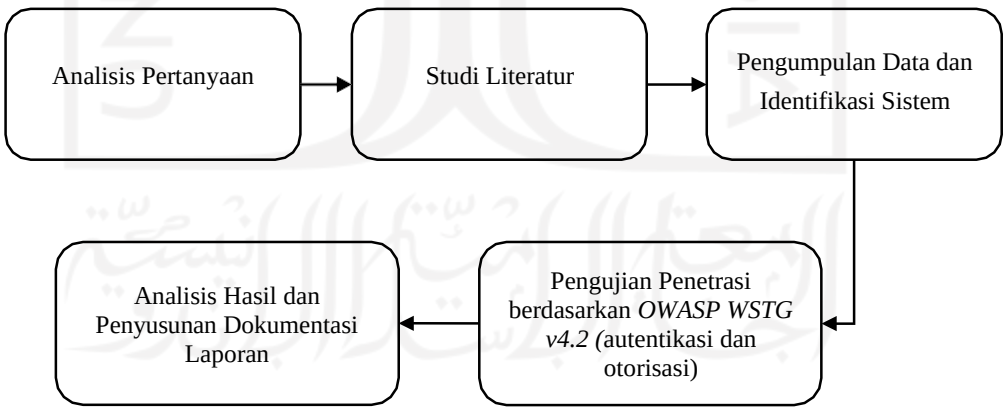
Yogyakarta, 8 Mei 2022

Target:

Website Mi-Gateway UII yang dikelola oleh Program Studi Informatika Program Magister Universitas Islam Indonesia yaitu Mi-Gateway UII. *Website* Mi-Gateway UII menjadi *platform* utama bagi calon mahasiswa, mahasiswa dan dosen Program Magister Informatika. Mi-Gateway memiliki beberapa fitur mulai dari pendaftaran mahasiswa baru, penilaian dosen hingga pendaftaran tesis. Proses pengujian celah keamanan *website* migateway-uii.id bertujuan untuk:

a. Melakukan analisis celah keamanan *website* Mi-Gateway UII menggunakan metode pengujian *OWASP WSTG v4.2*.
b. Mengetahui celah keamanan *website* Mi-Gateway UII sehingga dapat menjadi acuan dalam memperbaiki celah keamanan pada *website* Mi-Gateway UII.
c. Mengetahui kerentanan *website* Mi-Gateway UII dari hasil analisis pengujian celah keamanan menggunakan metode *OWASP WSTG v4.2.*

Tahapan pengujian penetrasi ini diawali dengan diawali dengan identifikasi celah keamanan *(vulnerability identification*) menggunakan *scanning tools* otomatis yaitu *OWASP ZAP*. Pada tahap pengujian ini fokus pada pengujian autentikasi dan otorisasi. Kemudian pembuatan laporan berupa hasil analisis dan penyusunan dokumentasi laporan.

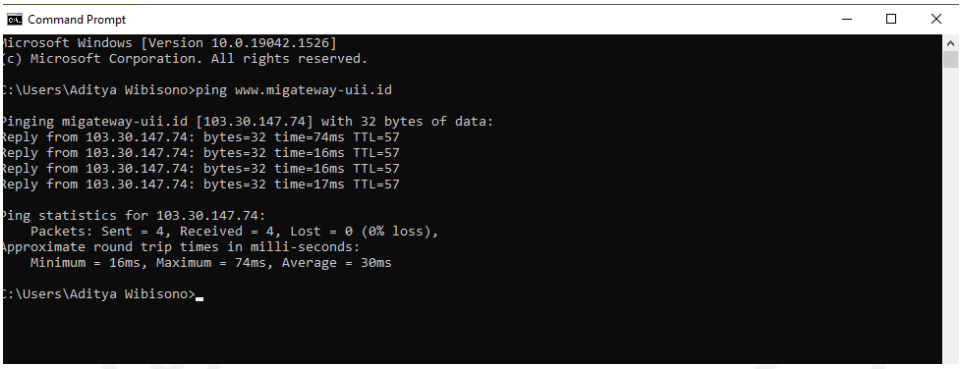


Gambar 1 Proses Pengujian Penetrasi

Pengumpulan Data dan Identifikasi Sistem

Tahap ini dilakukan pengumpulan data menggunakan beberapa *tools* untuk mendapatkan informasi mengenai *website* migateway-uii..id. Identifikasi menggunakan *tools command prompt*

didapatkan *IP adress website* migateway-uii.id menggunakan perintah *ping*. *Command prompt* merupakan suatu aplikasi *CLI (Command Line Interpreter)* yang terdapat pada *OS Windows*.

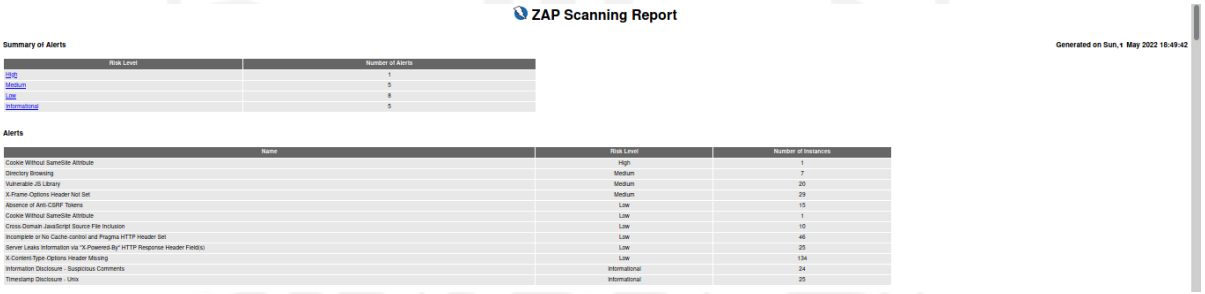


Gambar 2 Hasil *ping website* migateway-uii.id menggunakan *command prompt*

Berdasarkan Gambar 2 dapat dilihat bahwa dengan perintah *ping* dapat diketahui *IP address* dari suatu *website*, dalam penelitian ini menggunakan target *website* migateway-uii.id dengan *IP address* 103.30.147.74.

Scanning Otomatisasi OWASP ZAP

Tahap ini dilakukan pengujian celah keamanan *(Vulnerability Identification)* pada *website* migateway-uii.id menggunakan alat scan otomatis *OWASP ZAP*. *OWASP Zed Attack Proxy (ZAP)* adalah pemindai aplikasi berbasis website yang bersifat *open source*.



Gambar 3 Hasil *automated scan* OWASP ZAP

Berdasarkan Gambar 3 menunjukkan hasil *automated scan* menggunakan *OWASP Zed Attack Proxy (ZAP)* dengan celah keamanan yang terdeteksi pada *website* cek-ejaan.com yaitu sebanyak 18 *alerts* dengan jumlah kasus per *alerts* seperti yang tertera pada gambar, serta terbagi menjadi 3 kategori level risiko *medium*, *low*, dan *informational*, sehingga didapatkan kesimpulan bahwa website target tergolong cukup aman.

Metode OWASP WSTG v4.2

Tahap ini dilakukan pengujian *website* target menggunakan *framework WSTG (Web Security Testing Guide) OWASP* Versi 4 dengan fokus pengujian *authentication*, dan *authorization* menggunakan beberapa kombinasi tools. Hasil yang diperoleh dalam pengujian ini tertuang pada Tabel 1 dibawah ini.

Tabel 1 Hasil Pengujian menggunakan OWASP WSTG v4.2

Kategori Pengujian	**Tahapan Pengujian**	**Aktivitas**	***Tools***	**Status**	**Hasil**
Authentication Testing	*Testing for Credentials Transported over an Encrypted Channel (WSTG-ATHN-01)*	Memastikan data kredensial terenkripsi dengan menggunakan *HTTPS* dalam komunikasi antara klien ke server.	• *Mozilla Firefox*	Tidak ditemukan	Telah menerapkan *HTTPS*
	Testing for Default Credentials (WSTG-ATHN-02)	Memprediksi kredensial secara *default* dan validasi pada halaman *login*	• *BurpSuite* • *THC-Hydra*	Tidak ditemukan	Tidak ditemukan *username default* untuk *login user*
	Testing for Weak Lock Out Mechanism (WSTG-ATHN-03)	Melakukan beberapa kali *login* dengan *username* dan *password* yang salah untuk menguji mekanisme penguncian akun	• *Mozilla Firefox*	Ditemukan	Tidak ada mekanisme penguncian akun pada *user invalid login*
	Testing for Bypassing Authentication Schema (WSTG-ATHN-04)	Melakukan pengujian skema melewati otentikasi dengan metode	• *SQL Map*	Tidak ditemukan	Telah diterapkan *WAF (Web Application Firewall)* untuk

		request secara langsung atau paksa untuk menguji sudah diterapkannya autentikasi disetiap layanan.			memproteksi *HTTP* dan sebagai *tools* kontrol *traffic* pada *website.*
	Testing for Vulnerable Remember Password (WSTG-ATHN-05)	Melakukan pengujian kredensial pengguna pada sesi yang dihasilkan dengan melihat *log password* yang disimpan.	• *Mozilla Firefox*	Ditemukan	Pada halaman *login* tidak diterapkan *remember password.*
	Testing for Browser Cache Weaknesses (WSTG-ATHN-06)	Melakukan pengujian *cache browser* untuk menemukan informasi sensitif yang tersimpan pada sisi klien. Serta pengujian terhadap tombol "kembali" untuk melihat sumber daya yang ditampilkan pada halaman sebelumnya.	• *Mozilla Firefox* • OWASP ZAP	Tidak ditemukan	*Cache browser* tidak ditemukan dan fungsi tombol *back* pada *browser* tidak melakukan simpan *log*.
	Testing for Weak Password Policy (WSTG-ATHN-07)	Pengujian untuk memeriksa ketahanan *website* terhadap beberapa	• *THC-Hydra*	Tidak ditemukan	Tidak ada kata sandi yang berhasil ditemukan berdasarkan

		masukan kata sandi dengan metode *brute force*.			kamus kata sandi.
	Testing for Weak Security Question Answer (WSTG-ATHN-08)	Memeriksa skema pertanyaan keamanan dengan mengumpulkan kemungkinan jawaban dari serangkaian pertanyaan keamanan.	• *Mozilla Firefox*	Ditemukan	*Website* tidak menerapkan pertanyaan keamanan, perlu diterapkan sebagai keamanan tambahan di atas kata sandi utama.
	Testing for Weak Password Change or Reset Functionalities (WSTG-ATHN-09)	Pengujian untuk memeriksa mekanisme perubahan atau pengaturan ulang kata sandi.	• *Mozilla Firefox*	Tidak ditemukan	*Website* telah menerapkan *SSO (Single Sign On)* yang dilakukan oleh sistem terpusat menggunakan kode OTP dalam perubahan kata sandi.
	Testing for Weaker Authentication in Alternative Channel (WSTG-ATHN-10)	Mengidentifikasi saluran autentikasi yang lain (alternatif).	• *Mozilla Firefox*	Tidak ditemukan	*Website* menerapkan *SSO (Single Sign On)* sehingga autentikasi melalui sistem terpusat menggunakan *e-mail* universitas.

Authorization Testing	*Testing Directory Traversal File Include (WSTG-ATHZ-01)*	Mengidentifikasi lokasi file *root directory* atau *root* dokumen web.	• *DirbBuster* • *DotDotPwn*	Ditemukan	Terdapat *URL* yang terindikasi terdapat celah keamanan setelah proses *scanning* melalui *HTTPS.*
	Testing for Bypassing Authorization Schema (WSTG-ATHZ-02)	Pengujian untuk melewati skema autorisasi untuk mengakses dan mengoperasikan fungsi pada sumber daya khusus tanpa autentikasi.	• *BurpSuite*	Tidak ditemukan	Skema autorisasi telah diterapkan dengan tepat sehingga tidak ada pesan *error*.
	Testing for Privilege Escalation (WSTG-ATHZ-03)	Pengujian dengan metode injeksi *fuzz* untuk mendapatkan hasil sdengan manipulasi hak istimewa dari *user*.	• *OWASP ZAP*	Tidak ditemukan	Hak istimewa tidak dapat dimanipulasi.
	Testing for Insecure Direct Object References (WSTG-ATHZ-04)	Pengujian dengan modifikasi *URL website* untuk *direct request* ke objek referensi.	• *Mozilla Firefox*	Tidak ditemukan	Parameter *URL* pada halaman *log* tidak dapat dimodifikasi.

OWASP ZAP Scanning Report

Summary of Alerts

Risk Level	Number of Alerts
Medium	5
Low	8
Informational	5

Alerts

Name	Risk Level	Number of Instances
Directory Browsing	Medium	7
Vulnerable JS Library	Medium	20
X-Frame-Options Header Not Set	Medium	29
Absence of Anti-CSRF Tokens	Low	15
Cookie Without SameSite Attribute	Low	2
Cross-Domain JavaScript Source File Inclusion	Low	10
Incomplete or No Cache-control and Pragma HTTP Header Set	Low	46
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low	25
X-Content-Type-Options Header Missing	Low	134
Information Disclosure - Suspicious Comments	Informational	24
Timestamp Disclosure - Unix	Informational	25