Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi
Jl. Ahmad Yani, K.M. 33,5 - Kampus STMIK Banjarbaru
Loktabat – Banjarbaru (Tlp. 0511 4782881), e-mail: puslit.stmikbjb@gmail.com
e-ISSN: 2685-0893
p-ISSN: 2089-3787

Analisis Keamanan Sistem Informasi Akademik Berbasis *Web* Menggunakan *Framework* ISSAF

Rusydi Umar[1], Imam Riadi[2], Muhammad Ihya Aulia Elfatiha[3*]
[1,3]Program Studi Informatika, Universitas Ahmad Dahlan, Yogyakarta, Indonesia
[2]Program Studi Sistem Informasi, Universitas Ahmad Dahlan, Yogyakarta, Indonesia
*e-mail *Corresponding Author: elfatih2008048045@webmail.uad.ac.id*

Abstract

Utilization of Information Technology (IT) has now become a primary requirement in improving organizational performance in achieving goals. The Muhammadiyah Bekasi Business Institute (IBM Bekasi) in this case has utilized IT in its operational processes known as the Academic Information System which has been integrated with student and lecturer portals. The facilities provided by the system include academic modules, staffing, billing, e-filling and reporting. In this case study, the research aims to analyze system security using the Penetration Testing method using the ISSAF (Information System Security Assessment Framework) to measure system security with specific details at each stage carried out on an academic information system owned by the Muhammadiyah Bekasi Business Institute. As a result of the analysis carried out, it was found that the system is considered less secure from Brute-force Attacks, CSRF Attacks (Cross-Site Request Forgery), Session Hijacking through Cookies, and Insecure Direct Object References (IDOR).

Keywords: *ISSAF; Penetration Testing; Academic Information System.*

Abstrak

Pemanfaatan Teknologi Informasi (*Information Technology*/ IT) saat ini telah menjadi kebutuhan primer dalam meningkatkan kinerja organisasi dalam mencapai tujuan. Institut Bisnis Muhammadiyah Bekasi (IBM Bekasi) dalam hal ini telah memanfaatkan IT dalam proses-proses operasionalnya yang dikenal dengan Sistem Informasi Akademik yang telah terintegrasi dengan portal mahasiswa dan dosen. Fasilitas yang disuguhkan sistem meliputi modul akademik, kepegawaian, *billing*, *e-filling* hingga pelaporan. Dalam studi kasus ini, penelitian dituju untuk menganalisa keamanan sistem dengan metode Penetration Testing menggunakan ISSAF (*Information System Security Assesment Framework*) untuk mengukur keamanan sistem dengan rincian yang spesifik pada setiap tahapan yang dilakukan terhadap sistem informasi akademik yang dimiliki Institut Bisnis Muhammadiyah Bekasi menggunakan. Hasil analisa yang yang dilakukan, didapatkan temuan bahwa sistem dipandang kurang aman terhadap *Brute-force Attack*, CSRF *Attack* (*Cross-Site Request Forgery*), *Session Hijacking* menggunakan *Cookies*, dan *Insecure Direct Object References* (IDOR).

Kata kunci: *ISSAF; Penetration Testing; Sistem Informasi Akademik.*

1. Pendahuluan

Peran teknologi informasi sangat penting bagi perguruan tinggi agar mampu menghadapi persaingan, perguruan tinggi tentunya harus memiliki sistem informasi yang baik dan optimal guna menjadi nilai tambah bagi perguruan tinggi tersebut. Penerapan teknologi informasi sangatlah penting untuk menunjang keberlangsungan aktivitas sebagai media informasi dan komunikasi yang mudah, cepat dan mendukung tujuan perguruan tinggi untuk berkembang dan berdaya saing.

Institut Bisnis Muhammadiyah Bekasi (IBM Bekasi) dalam hal ini telah mengembangkan aplikasi AIS *(Academic Information System)* yang telah terintegrasi dengan modul keuangan, kepegawaian, *e-learning,* hingga modul Pelaporan *feeder* yang diharapkan mampu menjadi *role* model untuk perguruan tinggi khususnya dilingkup Muhammadiyah. Namun masalah yang berkaitan dengan keamanan informasi acap kali kurang diperhatikan, didukung dengan masih banyak ditemukannya teknologi informasi yang tidak memberikan efek kemanfaatan dan timbal

balik kepada organisasi *"IT Productivity Paradox*" [4] dan [5], padahal ini merupakan hal penting dari sistem informasi yang diterapkan manajemen [1]. IBM Bekasi belum pernah melakukan audit yang terukur terhadap sistem terutama pada aspek keamanannya. Sehingga sampai saat ini belum dapat mengetahui seaman apa data pada AIS diterapkan. Maka berdasarkan permasalahan yang ada, pada penelitian ini dilakukan analisis keamanan AIS yang dimiliki IBM Bekasi menggunakan metode *penetration testing* dengan kerangka kerja ISSAF. Kurang lebih dua tahun penggunaan AIS, acap kali terdapat keluhan-keluhan dari sisi pengguna siantaranya yaitu Aktifitas Kuliah Mahasiswa (AKM) yang akan dilaporkan tidak sesuai dengan Daftar Kumpulan Nilai (DKN) Mahasiswa, nilai yang diinput oleh dosen tidak tertampil di portal bahkan acap kali hilang, sehingga operator harus manual merekap data untuk pelaporan. Kemudian permasalahan lain adalah terjadinya waktu tunggu *login* yang mengakibatkan mahasiswa terlambat mengikuti kelas perkuliahan dan terkadang tautan pada portal tidak merespon atau menolak untuk terhubung. Permasalahan ini sejalan dengan pernyataan [2] bahwa semakin banyak informasi yang disimpan, dan dikelola semakin besar risiko terjadi kerusakan, kehilangan atau terbukanya data ke pihak-pihak yang tidak diinginkan.

Berdasarkan kondisi yang ada, terdapat indikator-indikator yang menyatakan bahwa tata kelola sistem informasi yang diterapkan di IBM Bekasi terbilang kurang dilakukan secara baik, sehingga dipandang perlu adanya mekanisme berupa audit atau analisis terhadap Tata Kelola Teknologi Informasi (IT Governance) terkait security system yang berjalan saat ini untuk dapat diperbaiki sehingga kedepannya didapat sebuah sistem informasi yang baik dan sesuai dengan standar yang berlaku [3].

Dalam studi ini, penelitian akan membahas mengenai pengukuran tingkat keamanan dari sistem AIS IBM Bekasi menggunakan kerangka kerja ISSAF kemudian dilakukan Penetration Testing (Pentest) berupa SQL Injection, DDOS dan Metaspolit. penelitian bertujuan untuk mendapat hasil evaluasi *security system* pada AIS IBM Bekasi yang tepat dan akurat. kemudian dari hasil pengujian yang dilakukan dibuatkan kesimpulan, pada bagian mana dari sistem yang sekiranya rentan terhadap serangan untuk menjadi bahan evaluasi sisitem diterapkan di IBM Bekasi sebagai rujukan perbaikan sistem kedepan [6] agar sistem semakin mapan.

2. Tinjauan Pustaka

Berikut adalah beberapa penelitian terdahulu yang telah dilakukan oleh banyak peneliti dalam Evaluasi keamanan sistem atau audit keamanan sistem informasi.

Penelitian dengan judul *Analisis Keamanan Web Server Open Journal System Menggunakan Metode ISSAF dan OWASP (Studi Kasus OJS Universitas Lancang Kuning)* oleh [8]. Penelitian bertujuan untuk menguji keamanan web server OJS dari tindak kejahatan peretas. Hasil penelitian didapat bahwa sistem tegolong aman, karena tidak mampu ditembus. Walaupun begitu, serangan terhadap sistem sewaktu-waktu dapat terjadi maka untuk menghidari jebolnya data mesti dilakukan monitoring untuk melindungi server semisal menerapkkan *firewall* maupun *Instruction Detection System* (IDS) dan melakukan perawatan serta pembaharuan sistem keamanan pada OJS secara berkala.

Penelitian dengan judul *Analisis Keamanan Jaringan Sistem Informasi Sekolah Menggunakan Penetration Test dan ISSAF* oleh [10]. Penelitian bertujuan untuk menemukan kelemahan dalam keamanan jaringan dan menilai tingkat keamanannya guna mencegah tindakan pencurian data atau penyalahgunaan hak akses. Hasil penelitian yang dilakukan 10 kali pada 11 data di setiap percobaan. Dihitung menggunakan Algoritma Naive Bayes menghasilkan nilai akurasi sebesar 72,72%, yang telah memenuhi Threshold Limit Value sebesar 70%. Hasil akurasi ini menunjukkan bahwa jaringan sistem informasi sekolah MTsN 8 Bantul tidak memiliki celah keamanan yang signifikan.

Penelitian dengan judul *Analisis Perbandingan Metode Web Security PTES, ISSAF dan OWASP di Dinas Komunikasi dan Informasi Kota Bandung* oleh [11]. Hasil penelitian Menggunakan PTES, domain diskominfo.bandung.go.id memiliki kerentanan berlevel *Medium*, dengan *Risk Rating* 6, domain memiliki 3 celah kerentanan, yaitu *Error Message on Page*, *Vulnerable Javascript Library*, dan *Application Error Message*. Hasil penelitian menggunakan ISSAF, domain memiliki 3 kerentanan yaitu *Application Error Message*, *Error Message on Page*, *Vulnerable Javascript Library*. penelitian menggunakan OWASP, domain memiliki kerentanan dilevel *Medium*, berdasarkan perhitungan menggunakan *Risk Rating Methodology*. Dari hasil pengujian ketiga *framework*, menyarankan *framework* yang tepat untuk digunakan ialah PTES

dan OWASP dikarenakan penilaiannya memakai level-level yang mudah untuk difahami oleh pengguna yang belum memahami atau berpengalaman dengan *Penetration Testing*.

Penelitian dengan judul *Evaluasi Keamanan Website Lembaga X Melalui Penetration Testing Menggunakan Framework ISSAF* oleh [12]. Penelitian bertujuan untuk mengetahui celah keamanan website dengan menggunakan metode *penetration testing* berdasarkan kerangka kerja ISSAF. Hasil penelitian didapat 18 celah keamanan pada website, diantaranya Injeksi SQL dan serangan XSS. Celah lainnya terdapat pada *port* TCP yang terbuka yang berisiko terhadap serangan. Pemberian rekomendasi pada *website* adalah dengan diadakannya validasi pada level php yang tujuannya agar ada pencegahan terhadap injeksi SQL dan serangan XSS yang mana adalah sumber terbanyak dari celah keamanan yang ada, terbukanya *port*-TCP yang harus segera dilakukan penutupan, dan pemulihan *bug* yang terdapat pada sistem yang mana mampu dimanfaatkan *attacker* sebagai celah keamanan.

Penelitian ini mempunyai perbedaan dengan penelitian sebelumnya, yaitu pada objek penelitian di perguruan tinggi swasta Institut Bisnis Muhammadiyah Bekasi, penelitian menerapkan kerangka kerja ISSAF untuk mengukur dan menganalisa keamanan informasi. Penelitian ini bertujuan untuk mencari kerentanan yang ada pada sistem, pada bagian mana dari sistem yang sekiranya rentan terhadap serangan untuk menjadi bahan evaluasi dan rujukan perbaikan sistem kedepan.

3. Metodologi

Tahapan proses pada penelitian ini merujuk pada kerangka kerja ISSAF (*Information System Security Assesment Framework)* adalah sebuah *framework* terstruktur yang membagi keamanan sistem informasi ke dalam berbagai kategori dan evaluasi yang spesifik. Tujuannya adalah untuk memberikan saran dan masukan berupa penilaian keamanan berdasarkan alur kerja yang sebenarnya dan dapat dijadikan sebagai acuan untuk memastikan keamanan sistem informasi. Penggunaan ISSAF sebagai kerangka kerja dapat membantu mengidentifikasi secara rinci potensi kerentanan yang mungkin ada pada sistem. Berikut tahapan-tahapan metode ISSAF yang ditunjukan pada Gambar 1.

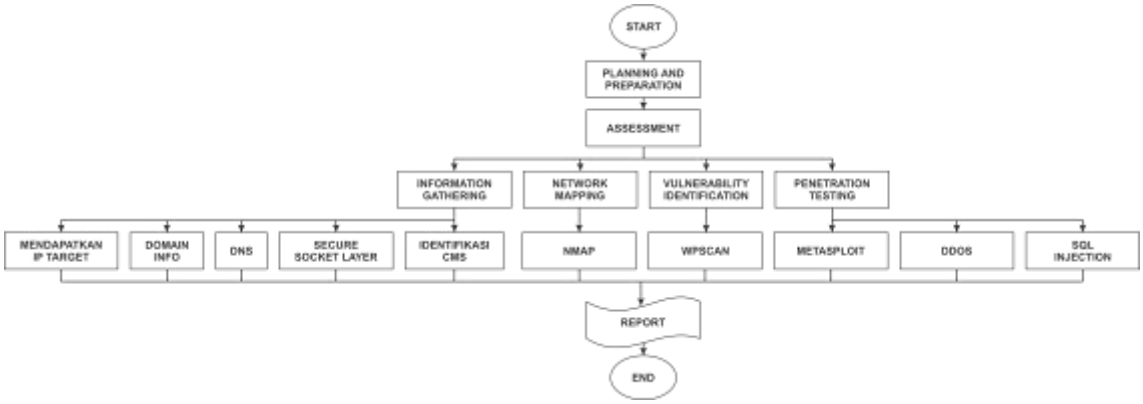
Gambar 1. Tahapan ISSAF

Penjelasan dari masing-masing tahapan pada metode Information System Security Assesment Framework adalah sebagi berikut:

1) *Information Gathering*, tahap pengumpulan informasi terkait sistem dan sebagai tahap persiapan *penetration testing*. Pengujian pada tahap ini antara lain untuk mengetahui SSL, DNS, info domain, dan identifikasi CMS.
2) *Network Mapping*, tahap pendekatan teknis dilakukan dengan menyisipkan "*Footprint*" kepada sistem.
3) *Vulnerability Identification*, tahap identifikasi kerentanan dilakukan dengan menguji beberapa aktifitas agar kerentanan dapat ditemukan pada sistem.
4) *Penetration*, tahap penyerangan dengan mencoba mendapatkan akses ke sistem secara ilegal dengan mensiasati sistem keamanannya dan mencoba untuk mencapai akses pada setiap level seluas-luasnya.
5) *Gaining Access and Privilege Escalation*, tahap lanjutan berupa pengujian yang dilakukan dengan menyerang sistem menggunakan akses yang telah didapat pada tahap penetrasi.
6) *Enumerating Further*, tahap pengumpulan data dan informasi lebih spesifik yang didapat setelah tahap penetrasi dan *gainning access*, hal ini dimaksudkan agar dapat membantu mengidentifikasi celah keamanan pada sistem yang ditargetkan.

7) *Compromise Remote User/Sites*, tahap memanfaatkan celah menggunakan *remote* jarak jauh untuk memperoleh hak akses yang lebih luas kedalam sistem agar memudahkan proses penelitian untuk meretas sistem dengan lebih dalam.
8) *Maintaining Access*, tahap pemeliharaan akses menggunakan *backdoor (root-kit)* agar mempertahankan akses ke dalam sistem bahkan setelah sistem tersebut telah dihapus.
9) *Covering the Track*, tahap pentester menyembunyikan berkas dan menghapus *log file* untuk menghapus jejak-jejak yang ada. Hal ini dilakukan dengan tujuan untuk menutupi aktivitas pengguna tes.
10) *Reporting*, tahap menulis laporan hasil pengujian beserta rekomendasi. Tujuannya untuk memberi informasi mengenai hasil pengujian yang telah dilakukan.
11) *Clean And Destroy Artifacts*, tahap penghapusan seluruh informasi yang sudah didapat atau diletakkan kedalam sistem. Tujuannya untuk memastikan tidak ada jejak yang tersisa setelah pengujian selesai

Pada penelitian ini tahapan *penetration testing* dilakukan sebagaimana dijelaskan pada gambar 1 kemudian diperinci dengan ilustrasi yang ditunjukan pada Gambar 2.



Gambar 2. Tahap Pengujian ISSAF

Setelah ilustrasi tahapan ISSAF yang ditunjukan pada gambar 2, kemudian dilakukan pemetaan terperinci tentang apa saja yang digali, dan apa saja *tools* yang dipakai dapat dilihat pada Tabel 1.

Tabel 1. Ringkasan tahapan implementasi ISSAF

Tahap	Source	Tools
Information Gathering	Domain Info	Nikto, Whois.
	SSL	SSL Scan, DNS Lookup.
Network Mapping	Network Info	N Map.
Vulnerability Identification	Web Scanner	ZAP.
	Vulnerability	
Exploitation	DoS Attack	Low Orbit Ion Canon.
	SQL Inject	Wireshark, SQL Map
	Metasploit	Metasploit
Gaining Access & Privilages Escalation	Backdoor	Php Rootkits
Enumerating Further	Backdoor	Php Rootkits
Compromise Remote User/Site	Backdoor	Php Rootkits
Maintaining Access	Backdoor	Php Rootkits
Covering Tracks	Backdoor	Php Rootkits
Reporting		Manual
Clean and Destroy		Manual

Tahap terakhir yaitu melakukan analisis dan membuat pelaporan dari hasil pengujian penetrasi yang sudah dilakukan berdasarkan *framework* ISSAF.

4. Hasil dan Pembahasan

Pada bagian ini, dijelaskan pembahasan terkait tahapan yang dilakukan beserta hasil penelitian terhadap objek, kemudian dari hasil yang didapat dibuatkan laporan serta rekomendasi yang berdasar hasil pengujian untuk perbaikansistem kedepan.

4.1. *Information Gathering*

Pada tahapan ini, peneliti menggunakan sumber dari internet untuk memperoleh sebanyak mungkin informasi dari target dengan memanfaatkan teknik teknis seperti DNS/WHOIS dan juga teknik non-teknis seperti mesin pencari, daftar alamat email, serta sumber publik lainnya. Kegiatan pengumpulan informasi ini diperlukan oleh peneliti untuk membangun hubungan dengan sistem target dan menggali informasi sebanyak mungkin tentang sistem tersebut.

4.1.1. Mendapatkan IP *Address* Target

Langkah untuk mendapatkan IP *Address* target dilakukan dengan menggunakan *tool* Nikto, yang mana *tool* ini merupakan salah satu *tool* yang di sediakan Kali Linux untuk melakukan pencarian celah keamanan pada suatu *website* yang hanya dengan *scanning* nama domain dari pada *website* itu sendiri. Untuk melakukan *scanning*, pada terminal kali Linux masukan perintah nikto -h ais.ibm.ac.id yang ditampilkan pada Gambar 3.

Gambar 3. Hasil pemindaian menggunakan nikto

Hasil yang diperoleh dari pemindaian *website* didapat informasi bahwa IP *Address* dari pada target adalah 54.179.125.207

4.1.2. Info Domain

Setelah mendapat IP target, kemudian mencari info domain dengan menggunakan *tool* whois pada terminal kali linux pada Gambar 4.

Gambar 4. Domain info menggunakan Whois

Setelah melakukan pemindaian menggunakan Hasil yang diperoleh pada *Information Gathering* disimpulkan pada Tabel 2.

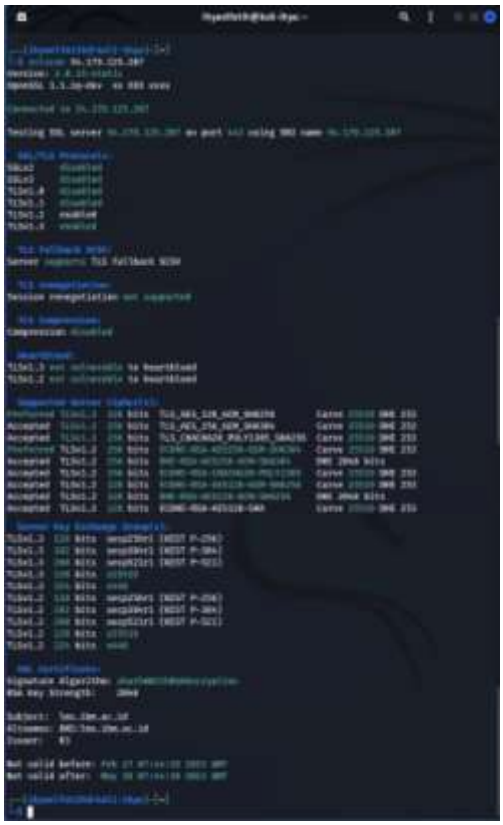
Tabel 2. Hasil pemindaian Whois

Information Gathering Results	
Domain ID	PANDI-DO1241026
Domain Name	ibm.ac.id
Create On	2019-02-18 02:09:02
Expiration Date	2024-02-18 00:09:02
Registrar Organization	PT. Registrasi Nama Domain

Registrar URL	https://daftarnama.id
Registrar Street	Cyber 2 Tower, Lt. 29 Jl. HR. Rasuna Said X5 No. 13, RT07/RW02
Registrar City	Jakarta Selatan
Registrar Province/State	Jakarta
Registrar Postal Code	12950
Registrar Country	ID
Registrar Phone	02180625578
Registrar E-Mail	info@daftarnama.id
Name Server	dns1.masterweb.com dns2.masterweb.com dns3.masterweb.com dns4.masterwebnet.com
Admin ID	-
Admin Name	-
Admin Organization	-

4.1.3. SSL *(Secure Socket Layer)*

Untuk mengetahui SSL pada domain ais.ibm.ac.id dilakukan pemindaian terhadap sistem dengan membuka terminal pada kali linux memasukan perintah sslscan 54.179.125.207 yang ditunjukan pada Gambar 5.



Gambar 5. Pemindaian SSL

Hasil yang diperoleh pada SSL didapat informasi berdasarkan kode warna yang menunjukan tentang tingkat keparahan dalam hal keamanan, dari hasil yang ditampilkan tidak ada kode yang berwarna merah artinya sistem tidak menunjukkan konfigurasi yang tidak aman, sedangkan pada TLSv1.2 pada sistem menunjukan kode berwarna putih dapat diartikan bahwa sistem masih tergolong aman atau disarankan. Kemudian pada TLSv1.3 menunjukan kode berwarna hijau yang diartikan konfigurasi dikategorikan aman. Hasil disimpulkan pada tabel 3.

Tabel 3. Hasil pemindaian SSL

SSL/TLS	Status	Vulnerable
SSLv2	Disabled	Not Identified
SSLv3	Disabled	Not Identified
TLSv1.0	Disabled	Not Identified
TLSv1.1	Disabled	Not Identified
TLSv1.2	Enabled	Not Vulnerable
TLSv1.3	Enabled	Not Vulnerable

4.1.4. DNS *(Domain Name Server)*

Pada pengujian *Domain Name Server* (DNS), penelitian menggunakan *tool* nslookup pada terminal kali linux yang diperlihatkan pada Gambar 6.

Gambar 6. Uji DNS dengan nslookup

Hasil dari pemindaian *Domain Name Server* menggunakan nslookup didapat bahwa server DNS merespon permintaan yang dimasukan dan memunculkan hasil yang sesuai antara nama domain dan IP, artinya tidak mengalami masalah atau tidak mengalami *error* yang berarti.

4.1.5. Identifikasi CMS

Identifikasi CMS dilakukan untuk mengetahui detail informasi dari CMS yang dipakai dengan menggunakan *tool* whatweb, pada terminal kali linux masukan perintah whatweb pada yang dapat dilihat pada Gambar 7.

Gambar 7. Identifikasi CMS

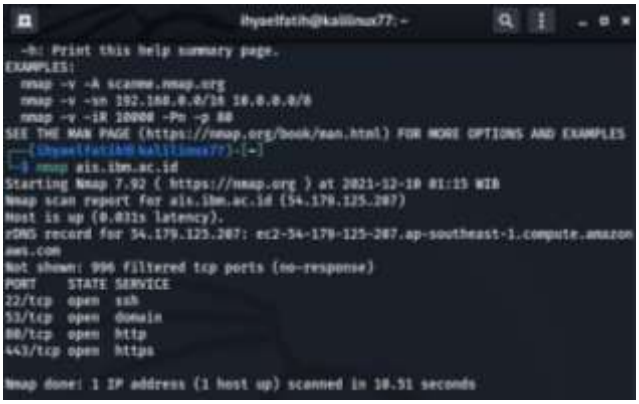
Hasil dari identifikasi yang dilakukan pada gambar 7 didapat informasi bahwa ais.ibm.ac.id menggunakan Bootstrap, CodeIgniter-PHP-Framework, HTML5, HTTPServer menggunakan Nginx, dan Jquery versi 1.11.1.3.2.1.

4.2. *Network Mapping*

Pengujian pada tahap *network mapping* dilakukan dengan memindai *port* pada *website*, yang kemudian dilanjutkan dengan pemindaian sistem operasi yang digunakan, dan pemindaian layanan yang dipakai pada sistem.

4.2.1. *Port Scanning*

Pemindaian *port* dilakukan pada tahap pertama *network mapping*, dengan menggunakan tool nmap pada teminal kali linux dengan mengetikan perintah nmap ais.ibm.ac.id, ditunjukan pada Gambar 8.



Gambar 8. Hasil pemindaian port

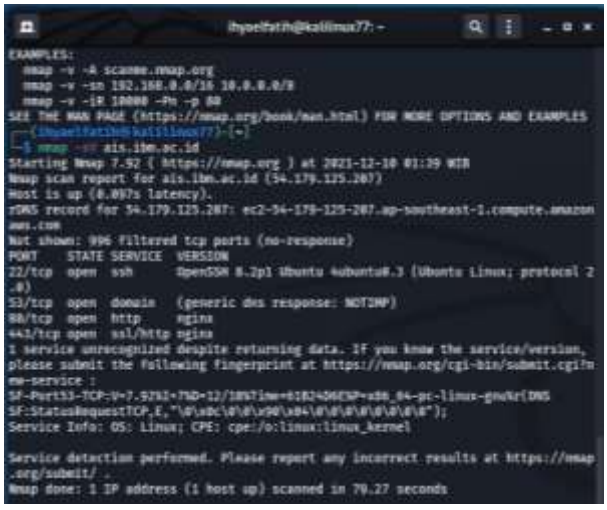
Hasil pemindaian *port* yang dilakukan menggunakan tool nmap pada gambar 8, didapatkan hasil ada beberapa *port* yang terbuka pada website ais.ibm.ac.id yang ditunjukan pada tabel 4.

Tabel 4. Kesimpulan hasil pemindaian port

Port	State	Services
22/tcp	Open	ssh
53/tcp	Open	domain
80/tcp	Open	http
443/tcp	Open	https

4.2.2. *Services and Operation System Scanning*

Pemindaian layanan dan sistem operasi dilakukan untuk mengetahui lebih detail tentang sistem berupa layanan yang dijalankan dan sistem operasi yang digunakan terkait website. Pemindaian menggunakan *tool* nmap dengan mengetikan perintah nmap -sV ais.ibm.ac.id pada terminal kali linux, ditunjukan pada Gambar 9.



Gambar 9. Pemindaian layanan dan sistem operasi

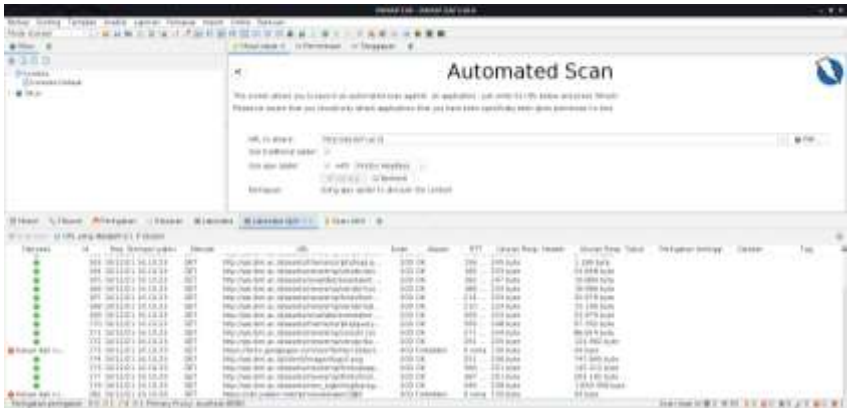
Hasil pemindaian layanan dan sistem operasi diperoleh bahwa sistem menggunkan OS Linux untuk lebih detailnya disimpulkan pada Tabel 5.

Tabel 5. Hasil pemindaian nmap

Port	State	Service	Version
22/tcp	open	ssh	UbuntuSSH 8.2p1 4ubuntu0.3 (ubuntu linux 2.0)
53/tcp	open	domain	ISC BIND 9.10.3-P4 (Ubuntu Linux)
80/tcp	open	http	nginx
443/tcp	open	https	ssl/http nginx

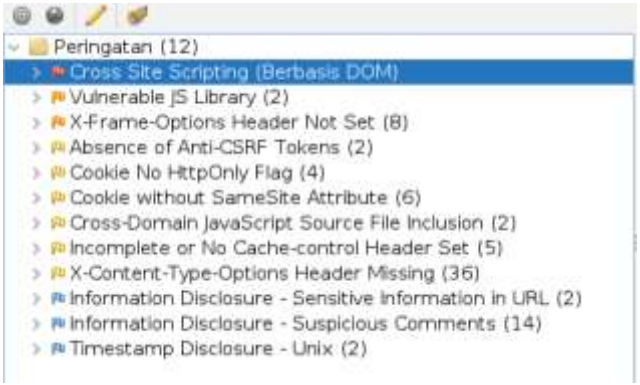
4.3. *Vulnerability Identification*

Pengidentifikasian vulnerability dilakukan dengan menggunakan tool ZAP, yang mana tool ini diperuntukan scanning URL target untuk dilakukan attacking yang digambarkan pada gambar 10.



Gambar 10. Proses pemindaian menggunakan tool ZAP

Setelah *scanning* yang dilakukan pada ais.ibm.ac.id ditemukan kerentanan pada sistem yang dijelaskan pada Gambar 11.



Gambar 11. Hasil pemindaian menggunakan tool ZAP

Dari hasil pemindaian yang diperlihatkan pada gambar 11, ditemukan bahwa sistem memiliki celah kerentanan yang dijelaskan pada Tabel 6.

Tabel 6. Hasil pengujian vulnerability identification

Domain	Kerentanan	Level
ais.ibm.ac.id	Cross Site Scripting (Berbasis DOM)	High
	Vulnerable JS Library	Medium
	X-Frame-Options Header Not Set	Medium
	Anti-CSRF Tokens	Low
	Cookie No HttpOnly Flag	Low
	Cookie without SameSite Attribute	Low
	Cross-Domain JS Source File inclusion	Low
	Incomplete or No Cache-control Header Set	Low
	X-Content-Type-Options Header Missing	Low
	Information Disclosure - Sensitive Information in URL	Informational
	Information Disclosure - Suspicious Comments	Informational
	Timestamp Disclosure - Unix	Informational

Pada tabel hasil pengujian *vulnerability identification* bahwa dari pemindaian yang dilakukan oleh tool ZAP menunjukan satu kerentanan pada level *high*, dua kerentanan pada level *medium*, enam kerentanan pada level *low*, serta tiga kerentanan ditemukan pada level *informational*.

4.4. *Exploitation*

Pada tahap eksploitasi, dilakukan penetrasi terhadap sistem dengan memulai simulasi serangan pada sistem yang menjadi target untuk mencari celah dalam keamanannya. Dalam tahap ini, pengujian dilakukan dengan menggunakan teknik SQL *Injection*, *Cross-Site Scripting* (XSS), dan *Broken Access Control* pada sistem target. Tujuannya adalah untuk menguji keamanan sistem dan menemukan kerentanan yang bisa dimanfaatkan oleh peneliti atau penyerang lainnya. Kesimpulan yang didapat dari hasil pengujian dapat dilihat pada tabel 7.

Tabel 7. Kesimpulan eksploitasi

Pengujian	Domain	Hasil
SQL Injection	http://ais.ibm.ac.id/assets/newtemp/vendor/countdowntime/countdowntime.js/	Gagal
	http://ais.ibm.ac.id/assets/newtemp/vendor/bootstrap/js/bootstrap.min.js/	Gagal
	https://cdn.jsdelivr.net/npm/sweetalert2@9/	Gagal
	http://ais.ibm.ac.id/assets/newtemp/vendor/jquery/jquery-3.2.1.min.js/	Gagal
	http://ais.ibm.ac.id/assets/theme/script/load.js?1359456394/	Gagal
	http://ais.ibm.ac.id/assets/newtemp/js/main.js	Gagal
	http://ais.ibm.ac.id/assets/newtemp/vendor/animation/js/animation.js/	Gagal
	http://ais.ibm.ac.id/assets/newtemp/vendor/bootstrap/js/popper.js/	Gagal
	http://ais.ibm.ac.id/assets/newtemp/vendor/jquery/jquery-1.11.1.js/	Gagal
	http://ais.ibm.ac.id/assets/theme/script/jquery-1.11.1.js	Gagal
XSS Cross Scripting	http://ais.ibm.ac.id/assets/swal#jaVasCript:/	Gagal
	http://ais.ibm.ac.id/index.php/welcome/login/	Gagal
	http://ais.ac.id/assets/theme/scripts/jquery-1.11.1.js	Gagal
Broken Access Control	http://ais.ibm.ac.id/login/index.php/	Gagal
	http://ais.ibm.ac.id/#c_mahasiswa/view/2366	Gagal
	http://ais.ibm.ac.id/search/index.php/	Berhasil

Berdasarkan Tabel 7, didapat bahwa pengujian terhadap SQL *Injection* tidak berhasil, dan pengujian pada XSS *Cross-Scripting* gagal dilakukan. Demikian juga, pengujian *Broken Access Control* melalui *Bypass Login* juga tidak berhasil. Akan tetapi, pada saat menguji kolom pencarian website ais.ibm.ac.id, ditemukan adanya celah pada parameter id yang memungkinkan penyerang untuk memperoleh informasi mengenai pengguna yang terdaftar dalam sistem.

4.5. *Gaining Access and Privilage Escalation*

Tahap ini bertujuan untuk memperoleh hak akses istimewa dengan cara memperoleh akses ke akun menggunakan berbagai teknik, seperti melakukan *brute-force attack* dengan mencoba kombinasi nama pengguna dan kata sandi, atau menggunakan *dictionary attacks* dengan mencoba kata sandi kosong atau kata sandi default. Pada penelitian ini, pengujian dilakukan dengan tool Hydra menggunakan metode *brute-force attack*. Kesimpulan yang didapat dapat dilihat pada tabel 8.

Tabel 8. Kesimpulan gainning access and privilage escalation

Pengujian	Status	Hasil
Brute-force Attack	Terdapat celah pada login page, namun tidak ditemukan kata sandi dan username yang sesuai	Gagal

Berdasarkan Tabel 8, Ditemukan kelemahan pada halaman login website ais.ibm.ac.id secara manual yaitu celah untuk disisipi *Brute-force attacks*, karena halaman *login* tidak memiliki perlindungan *limit login attempt*. Meskipun begitu, pengujian pada tahap ini tidak berhasil karena tidak ditemukan kombinasi nama pengguna dan kata sandi yang tepat.

4.6. *Enumerating Further*

Tahap ini adalah lanjutan dari tahap *gaining access and privilage escalation*, pada tahap ini dilakukan *sniff traffic* menggunakan Wireshark sebagai *tool* dan untuk sesi eksploitasi mengambil *cookie* yang didapat dari *website* ais.ibm.ac.id. Kesimpulan terhadap hasil pengujian *Enumerating Further* dapat dilihat pada Tabel 9.

Tabel 9. Hasil enumerating further

Pengujian	Keterangan	Hasil
Session Hijacking dengan Cookie	Login menggunakan cookie	Berhasil
Sniffing Traffic	Paket berisi data (nama pengguna dan kata sandi) dalam jaringan terenkripsi	Gagal

Berdasarkan hasil pada Tabel 9 menunjukkan bahwa pengujian *Session Hijacking* menggunakan *Cookies* berhasil dilakukan, yang menandakan adanya kerentanan dalam sistem karena seorang peretas dapat melakukan *login* pada *website* tanpa menggunakan nama pengguna dan kata sandi, dengan memanfaatkan *session login* yang tersimpan pada *cookies* di *browser*. Namun, pengujian *Sniffing Traffic* menunjukkan bahwa *website* ais.ibm.ac.id sudah aman karena data yang dikirim sudah dienkripsi dengan TLS 1.2.

4.7. *Compromise Remote User/Site*

Tahap ini pengujian dilakukan dengan cara eksploitasi untuk memperoleh akses *root (user)* pada *website* melalui *remote*. Untuk melakukan pengujian pada tahap ini, Metasploit digunakan dengan metode menyerang *port-port* terbuka yang telah didapatkan melalui tahap *Scanning* menggunakan *Network Mapping*. Hasil pengujian dapat dilihat pada Tabel 10.

Tabel 10. Hasil pengujian compromise remote user/site

Port	Metode Serangan	Status	Hasil
ssh 22/tcp	Brute-force key attack	Berhasil dilakukan, namun tidak menemukan kunci yang sesuai	Gagal
	Brute-force wordlist attack	Berhasil dilakukan, namun password dan username yang tidak ada yang sesuai	Gagal
http 80/tcp	Brute-force wordlist attack	Tidak dapat dilakukan brute-force attack	Gagal
https 443/tcp	HeartBleed Attack	Tidak ditemukan celah kerentanan	Gagal

4.8. *Maintaining Access*

Tahapan ini dilakukan untuk mendapatkan akses ke sistem dan menanamkan beberapa *backdoor* yang bisa digunakan untuk melakukan akses kedalam sistem target. Hasil pengujian dapat dilihat pada tabel 11.

Tabel 11. Hasil maintaining access

Backdoor	Keterangan	Hasil
Marijuana Backdoor	Terunggah, namun tidak dapat dieksekusi karena .htaccess di server terkonfigurasi untuk force download	Gagal
Weevely Backdoor	Terunggah, namun tidak dapat dieksekusi pada server karena script dilakukan filtering	Gagal
Remote Control Execution (RCE)	Terunggah, tetapi tidak dapat dieksekusi, karena server mematikan fitur execute shell comand	Gagal

Hasil pada Tabel 11 menunjukkan bahwa pengujian tidak berhasil dilakukan. Meskipun semua *backdoor* dapat diunggah ke dalam server, file tidak ada yang dapat dieksekusi karena server dilengkapi dengan sistem keamanan yang meliputi *filtering script*, *force download*, dan *disable execute shell command*.

4.9. *Covering the Track*

Tahap ini bertujuan untuk menghilangkan atau menutupi jejak yang dapat terdeteksi oleh administrator sistem bahwa terjadi aktivitas mencurigakan pada *website*. Caranya adalah dengan menghapus *log* pada sistem. Tindakan penghapusan *log* pada sistem dilakukan agar serangan yang terjadi pada tahap-tahap sebelumnya tidak terdeteksi oleh administrator. Namun, pada penelitian ini, tahap ini tidak berhasil dilakukan karena tidak berhasil diperoleh akses menuju server *(root).* Sehingga *log* dari serangan sebelumnya tidak dapat dihapus atau dihilangkan.

4.10. *Report and Result*

Tahapan *result and report* merupakan tahapan akhir yang merupakan laporan dari hasil uji penetrasi yang dilakukan pada website ais.ibm.ac.id, yaitu mengenai celah dan kerentanan apa saja yang ditemukan pada website target dan berapa nilai kerentanan yang didapat (CVSS) *Common Vulnerability Scoring System*. Kesimpulan dapat di lihat pada Tabel 12.

Tabel 12. Report and result

Vulnerability	Status	CVSS
Session Hijacking melalui Cookies	Ada penambahan cookie berupa auth_token setelah melakukan login Menerapkan regenerated cookie value pada setiap permintaan	9.0
Brute-force Attack	Terdapat limit login attempt Menerapkan captcha pada setiap kali login	7.8
CSRF (Cross-Site Request Forgery)	CSRF token diterapkan pada kolom search	7.6
IDOR (Insecure Direct Object Reference)	Parameter URL disembunyikan	6.8
Unrestrictied File Upload	Penerapan batasan file yang dapat diunggah: tidak lebih dari 10mb dan ekstensi diperbolehkan hanya docx, xlsx, pptx, pdf.	6.5

Dapat dilihat pada Tabel 10 dari seluruh tahapan pengujian menggunakan *framework* ISSAF yang dilakukan, terdapat 5 celah kerentanan yang di urutkan dari yang paling rentan (*critical/high*) hingga *medium* pada ais.ibm.ac.id. dari hasil yang telah di dapat, kerentanan yang ditemukan menunjukan bahwa kerentanan yang paling tinggi terdapat pada session hijacking melalui cookies dengan nilai 9.0, kemudian setelah dilakukan *brute-force attack* sistem menerapkan limit login yang menempatkan kerentanan dengan brute-force pada level *medium* dengan nilai 7.8, begitu pun dengan CSRF setelah dilakukan penyerangan terdapat celah kerentanan berada pada level *medium* dengan nilai 7.6. Hasil pengukuran CVSS pada sistem setelah dilakukan penetrasi menunjukan bahwa kerentanan terendah ada pada *unrestrictied file upload* dengan nilan 6.5 karna sistem menerapkan batasan file yang dapat diunggah yaitu tidak lebih dari 10mb dan hanya memperbolehkan ekstensi docx, xlsx, pptx dan pdf.

Hasil temuan pada penelitian mendukung temuan pada penelitian oleh [13] dimana celah keamanan pada website yang paling rentan adalah SQL injection dan CSRF yang mana merupakan sumber celah terbanyak, celah lainnya juga terdapat pada port-port yang terbuka yang berisiko terhadap serangan. Pemberian rekomendasi pada website adalah dengan diadakannya validasi pada level php yang bertujuan untuk pencegahan terhadap SQL Injection dan serangan XSS atau CSRF kemudian penutupan port-port TCP dan pemulihan bug yang terdapat pada sistem. Penelitian juga mendukung temuan pada penelitian [8] dimana sistem harus dilakukan monitoring, perawatan dan pembaharuan secara berkala untuk melindungi sistem dan menghidari peretasan terhadap sistem dikarenakan serangan terhadap sistem sewaktu-waktu dapat terjadi.

5. Simpulan

Dari penelitian yang telah dilakukan, didapat temuan terkait *website* yang dianalisis diantaranya yaitu tidak adanya anti-clickjacking X-Frame-Option, clickjacking merupakan teknik serangan agar target mau melakukan klik didaerah tertentu, yang artinya ini dapat sangat menguntungkan seorang *attacker*. Kemudian Tidak adanya XSS *Protection Header*, yang artinya seorang yang akan menyerang *website* tersebut dapat melakukan eksploitasi menggunakan XSS. *Header* yang terdapat pada *webiste* tidak umum, biasanya *header* ini dipakai *attacker* untuk menyembunyikan seluruh aktivitas *attacking* nya agar tidak mudah diketahui.

Metode ISSAF yang dipakai mendapatkan hasil bahwa dari sembilan tahap yang dilaksanakan, diperoleh bahwa ais.ibm.ac.id dipandang kurang aman dari serangan *Brute-force Attack* dengan nilai 7.8, kemudian pada CSRF *Attack (Cross-Site Request Forgery)* dengan nilai 7.6, *Session Hijacking* melalui *Cookies* dengan nilai 9.0, dan *Insecure Direct Object References* (IDOR) dengan nilai 6.8. Dari hasil yang diperoleh diharapkan dapat ditindak lanjuti pengembang *website* supaya dapat ditingkatkan lagi segi keamanannya dan menutup celah kerentanan yang ada.

Daftar Referensi

[1] A. R. Tanaamah dan F. J. Indira, “Analysis of Information Technology Security Management SWCU SIASAT Using ISO/IEC 27001:2013”, IJITEE, vol. 5, no. 2, pp. 68-74, Juni 2021. https://doi.org/10.22146/ijitee.65670

[2] R. Sarno dan I. Iffano. Sistem Manajemen Keamanan Informasi Berbasis ISO 27001, Surabaya: ITS Press, 2009.

[3] H. F. Tipton and M. Krause. Information Security Management Handbook, London: CRC Press, 2010.

[4] M. H. Murdani, M. U. Sari, and Muharom, “IT Productivity Paradox pada Perguruan Tinggi Swasta,” Jurnal Ilmiah Teknologi Informasi Asia, vol. 12, no. 2, pp. 81-90, 2018. doi:10.32815/jitika.v12i2.216

[5] R. Gilang Jodi Putra, “Paradoks Produktivitas Teknologi Informasi: Analisis Investasi Sistem Aplikasi CRM,” Tugas Akhir, Jurusan Sistem Informasi, ITS, Surabaya.

[6] H. F. Tipton and M. Krause. Information Security Management Handbook, London: CRC Press, 2010.

[7] I. Riadi, R. Umar, and D. Bernadisman, "Analisis Forensik Database Menggunakan Metode Forensik Statis," JSINBIS (Jurnal Sistem Informasi Bisnis), vol. 9, no. 1, pp.9-17, Mei 2019. https://doi.org/10.21456/vol9iss1pp9-17.

[8] G. L. Costaner, and Musfawati. “Analisis Keamanan Web Server Open Journal System (OJS) Menggunakan Metode ISSAF dan OWASP (Studi Kasus OJS Universitas Lancang Kuning),” JIPI, vol. 05, no. 01, pp.45-55, Juni 2020. https://doi.org/10.29100/jipi.v5i1.1565.

[9] E. P. Silmina, A. Firdonsyah, and R. A. Amanda, "Analisis Keamanan Jaringan Sistem Informasi Sekolah Menggunakan Penetration Test dan ISSAF," Jurnal Transmisi, vol. 24, no. 3, pp.83-91, Agustus 2022. https://doi.org/10.14710/transmisi.24.3.83-91.

[10] T. R. Syarif, “Analisis Perbandingan Metode Web Security PTES, ISSAF, dan OWASP di Dinas Komunikasi dan Informasi Kota,” Skripsi, Program Studi Teknik Informatika, Universitas Komputer Indonesia, Kota Bandung.

[11] S. E. Prasetyo and N. Hassanah, “Analisis Keamanan Website Universitas Internasional Batam Menggunakan Metode ISSAF”, oai, vol. 9, no. 02, pp.82–86, September 2021. https://doi.org/10.33884/jif.v9i02.3758.

[12] I G. Ary, G. M. A. Sasmita, and D. M. Sri Arsa, “Evaluasi Keamanan Website Lembaga X Melalui Penetration Testing Menggunakan Framework ISSAF,” Jurnal Ilmiah Merpati, vol. 8, no. 2, pp.113-124, Agustus 2020. https://doi.org/10.24843/JIM.2020.v08.i02.p05.

[13] A. Wisnu Wardhana, and H. Bayu Seta, “Analisis Keamanan Sistem Pembelajaran Online Menggunakan Metode ISSAF pada Website Universitas XYZ,” Jurnal Informatik IFTK, vol. 17, no. 3, pp.226-237, Desember 2021. https://doi.org/10.52958/iftk.v17i3.3653.

[14] A. Fadlil, I. Riadi, and A. Nugrahantoro, “Data Security for School Service Top-Up Transactions Based on AES Combination Blockchain Technology,” Lontar Komputer, vol. 11, no. 3, pp.155-166, Desember 2020. https://doi.org/10.24843/lkjiti.2020.v11.i03.p04.

[15] S. Hidayatulloh dan D. Saptadiaji, “Penetration Testing pada Website Universitas ARS Menggunakan Open Web Application Security Project (OWASP)”, Jurnal Algoritma, vol. 18, no. 1, pp.77-86, Agustus 2021. https://doi.org/10.33364/algoritma/v.18-1.827.

[16] I. Riadi, A. Yudhana, and Yunanri W. “Analisis Keamanan Website Open Journal System Menggunakan Metode Vulnerability Assesment,” Jurnal JTIIK, vol. 7, no. 4, pp.853-860, Agustus 2020. http://dx.doi.org/10.25126/jtiik.2020701928.

[17] H. Tanuwijaya, “Analisis Keamanan Sistem Informasi Perdaganagan Terintegrasi Menggunakan Standar ISO 27002,” JUTISI Banjarbaru, vol. 11, no. 3, pp. 571-582, Desember 2022. https://dx.doi.org/10.35889/jutisi.v11i3.993.

[18] F. Fachri, A. Fadlil, and I. Riadi, “Analisis Keamanan Webserver Menggunakan Penetration Test”, Junal Informatika BSI, vol. 8, no. 2, pp. 183-190, September 2021. https://doi.org/10.31294/ji.v8i2.10854.