

SKRIPSI

ANALISIS *VULNERABILITY* PADA *WEBSITE* SMPK BINTANG LAUT KOTA TERNATE MENGGUNAKAN METODE *VULNERABILITY* *ASESSMENT* (Berdasarkan Panduan OWASP 10)



OLEH
Mahani Albaar
07351711012

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KHAIRUN
TERNATE
2024**

LEMBAR PENGESAHAN

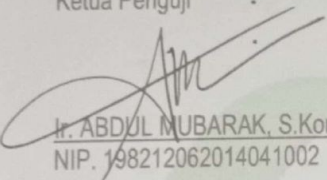
ANALISIS VULNERABILITY PADA WEBSITE SMPK BINTANG LAUT KOTA TERNATE MENGUNAKAN METODE VULNERABILITY ASSESMENT (BERDASARKAN PANDUAN OWASP 10)

Oleh
Mahani Albaar
07351711012

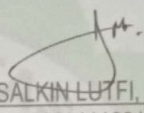
Skripsi ini telah disahkan
Tanggal 07 Juni 2024

Menyetujui
Tim Penguji

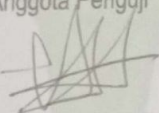
Ketua Penguji


Ir. ABDUL MUBARAK, S.Kom., M.T., IPM.
NIP. 198212062014041002

Pembimbing I


Ir. SALKIN LUTFI, S.Kom., M.T.
NIP. 198601112014041002

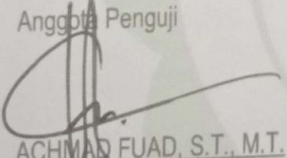
Anggota Penguji


MUHAMMAD FHADLI, S.kom., M.Sc.
NIP. 199611232023211012

Pembimbing II

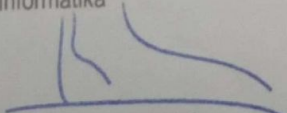

Ir. AMAL KHAIRAN, S.T., M.Eng., IPM.
NIP. 197401112003121003

Anggota Penguji


ACHMAD FUAD, S.T., M.T.
NIP. 197606182005011001

Mengetahui/Menyetujui

Koordinator Program Studi
Informatika


ROSIHAN, S.T., M.Cs.
NIP. 197607192010121001

Dekan Fakultas Teknik
Universitas Khairun


Ir. ENDAH HARISUN, S.T., M.T., CRP.
NIP. 197511302005011013



LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Mahani Albaar
Npm : 07351711012
Fakultas : Teknik
Jurusan/Program Studi : Informatika
Judul : Analisis *Vulnerability* pada Website SMPK Bintang Laut Kota Temate Menggunakan Metode *Vulnerability Assessment* (Berdasarkan Panduan OWASP 10)

Dengan ini menyatakan bahwa penulisan Skripsi yang telah saya buat ini merupakan hasil karya saya sendiri dan benar keasliannya. Apabila ternyata di kemudian hari penulisan Skripsi ini merupakan hasil plagiat atau penjiplakan terhadap karya orang lain, maka saya bersedia mempertanggung jawabkan sekaligus bersedia menerima sanksi berdasarkan aturan tata tertib di Universitas Khairun.

Demikian pernyataan ini saya buat dalam keadaan sadar dan tidak dipaksakan.

Penulis



Mahani Albaar

LEMBAR PERSEMBAHAN

Bismillahirrahmanirrahim

Dengan rasa Syukur yang mendalam, dengan telah selesainya skripsi ini penulis mempersembahkannya kepada:

1. Allah SWT yang telah memudahkan dan melancarkan sehingga saya dapat menyelesaikan skripsi ini.
2. Kepada kedua orang tua saya, penyemangat ku ayahanda Ali Albar dan ibunda tercinta Nurhaida Esa. Dua orang yang paling amat berjasa dan berperan penting dalam membantu saya untuk menyelesaikan skripsi ini.
3. Kepada kaka saya tercinta yang selalu memberikan semangat dan dorongan kepada saya
4. Kepada Angkatan tercinta 2017 yang selalu memberikan semangat dan bantuan dalam mas aperkuliah sampai saat ini.

MOTTO

“SELESAIKAN APA YANG KAMU MULAI”

KATA PENGANTAR

Puji dan syukur penulis persembahkan kehadiran Tuhan Yang Esa, karena berkat rahmat dan karunia-Nya semata sehingga penulis mampu menyelesaikan penyusunan Skripsi dengan judul “Analisis *Vulnerability* Pada *Website* SMPK Bintang Laut Menggunakan Metode *Vulnerability Assessment* (Berdasarkan Panduan Owasp 10).

Penyusunan skripsi ini adalah untuk memenuhi salah satu persyaratan kelulusan pada Universitas Khairun Ternate Fakultas Teknik Program Studi Informatika. Penyusunan laporan ini dapat terlaksana dengan baik berkat dukungan dari banyak pihak, untuk itu pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Bapak Dr. M. Ridha Ajam, S.H., M.Hum., Selaku rektor Universitas Khairun Ternate.
2. Bapak Ir., Endah Harisun, S.T., M.T., CRP., Selaku Dekan Fakultas Teknik Universitas Khairun Ternate.
3. Bapak Rosihan, ST., M.Cs., Selaku Ketua Program Studi Teknik Informatika.
4. Bapak Ir. Salkin Lutfi, S.Kom., M.T., selaku pembimbing I, terima kasih telah memberikan arahan kepada penulis dan bimbingannya serta dorongan dalam menyelesaikan Skripsi ini.
5. Bapak Ir. Amal Khairan, S.T., M.Eng., IPM., selaku pembimbing II, terima kasih telah membimbing peneliti dengan sabar serta memberikan pembelajaran dan saran yang sangat bermanfaat dalam penyusunan skripsi ini bimbingannya serta dorongan dalam menyelesaikan Skripsi ini.
6. Bapak Hairil Kurniadi Sirajuddin, S.Kom., M.Kom., selaku penasehat akademik, terima kasih atas bimbingannya serta dorongan saya selama kuliah.
7. Bapak Ir. Abdul Mubarak, S.Kom., M.T., IPM., selaku Dosen Penguji I Tugas Akhir yang telah bersedia memeberikan saran serta ilmu yang bermanfaat kepada penulis dalam menyelesaika Skripsi ini.
8. Bapak Muhammad Fhadli, S.Kom., M.Sc., selaku Dosen, penguji II yang telah bersedia memberikan saran serta ilmu yang bermanfaat kepada penulis dalam menyelesaikan Skripsi ini.
9. Bapak Achmad Fuad, S.T., M.T., selaku Dosen Penguji III yang telah bersedia memberikan saran serta ilmu yang bermanfaat kepada pennulis dalam

menyelesaikan skripsi ini.

10. Kedua orang tua, kakak, adik, serta seluruh keluarga yang selalu memberikan dukungan dan perhatian penuh hingga penulis dapat menyelesaikan Skripsi ini dengan baik.
11. Terima kasih kepada kerabat dan semua pihak yang tidak bisa penulis sebutkan satu persatu yang telah membantu penulis baik langsung maupun tidak langsung dalam menyelesaikan Skripsi.

Semua pihak terkait sangat penulis harapkan. Akhir kata, semoga penyusunan Skripsi ini dapat bermanfaat bagi kita semua.

Ternate, 10 Januari 2024

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN.....	ii
HALAMAN PERNYATAAN KEASLIAN.....	iii
HALAMAN PERSEMBAHAN.....	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	xi
ABSTRAK.....	xii
 BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan masalah.....	2
1.3. Batasan Masalah.....	2
1.4. Tujuan Penelitian.....	2
1.5. Manfaat Penelitian	3
1.6. Sistematika Penulisan	3
 BAB II TINJAUAN PUSTAKA	
2.1. Penelitian Terkait.....	4
2.2. Pengertian Analisis	6
2.3. Keamanan Sistem informasi	7
2.4. Website	7
2.5. Bentuk-bentuk Ancaman dari Sistem Komputer.....	8
2.6. <i>Vulnerability Assessment (VA)</i>	9
2.6.1. <i>Reconnaissance</i>	9
2.6.2. <i>Scanning</i>	10
2.7. <i>Penetration Testing</i>	10
2.8. <i>Open Web Application Security Project (OWASP)</i>	12
2.9. SMPK Bintang Laut Kota Ternate	14

2.10. <i>Flowchart</i>	14
2.11. <i>Owaps Zap</i>	15

BAB III METODE PENELITIAN

3.1. Tempat dan Waktu Penelitian.....	17
3.2. Tahapan Penelitian.....	17
3.3. Alat Penelitian	18
3.1.1. Perangkat keras (<i>Hardware</i>)	19
3.1.2. Perangkat Lunak (<i>Software</i>).....	19
3.4. Metode Pengumpulan Data	19
3.5. Tahapan <i>Vulnerability Assessment</i>	20
3.6. Analisa Hasil dan Pelaporan Hasil	22

BAB IV HASIL DAN PEMBAHASAN

4.1. <i>Reconnaissance</i>	23
4.2. <i>Scanning</i>	24
4.3. <i>Penetration Testing</i>	33
4.3.1. <i>Sql Injection</i>	33
4.3.2. <i>Cross Site Scripting</i>	35
4.3.3. <i>ClickJacking</i>	37
4.3.4. <i>Hidden File Found</i>	39
4.4. Analisis.....	39
4.5. Hasil.....	43

BAB V PENUTUP

5.1. Kesimpulan.....	44
5.2. Saran.....	44

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

	Halaman
Gambar 2.1. <i>SQL Injection</i>	11
Gambar 2.2. <i>XSS</i>	12
Gambar 2.3. <i>Profil SMPK Bintang Laut Kota Ternate</i>	14
Gambar 3.1. <i>Tahapan Penelitian</i>	17
Gambar 3.2. <i>Tahapan Vulnerability Asessment</i>	20
Gambar 4.1. <i>Website SMPK Bintang Laut Ternate</i>	23
Gambar 4.2. <i>Tampilan OwapsZap</i>	25
Gambar 4.3. <i>Tampilan Hasil Scan Vulnerrability</i>	25
Gambar 4.4. <i>Tampilan Hasil Scan Cross Site Scripting</i>	26
Gambar 4.5. <i>Tampilan Hasil Scan SQL Injection</i>	26
Gambar 4.6. <i>Tampilan Hasil Scan Absence Of Anti-CSRF Token</i>	27
Gambar 4.7. <i>Tampilan Hasil Scan Cross-Domain Misconfiguration</i>	28
Gambar 4.8. <i>Tampilan Hasil Hidden File Found</i>	28
Gambar 4.9. <i>Tampilan Hasil Missing Anti ~click Jacking Header</i>	29
Gambar 4.10. <i>Tampilan Vulneable JS Library</i>	30
Gambar 4.11. <i>Tampilan Hasil Application Error Disclosure</i>	30
Gambar 4.12. <i>Tampilan Hasil Scan Big Redirect Detected</i>	31
Gambar 4. 13. <i>Hasil Scan Cross-Domain JavaScript Source File Inclusion</i>	31
Gambar 4.14. <i>Hasil Scan Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)</i>	32
Gambar 4.15. <i>Hasil Scan Strict-Transport-Security Header Not Set</i>	33
Gambar 4.16. <i>Tampilan Payload Pada Terminal Linux</i>	34
Gambar 4.17. <i>Tampilan awal Exploit Sql Injection</i>	34
Gambar 4.18. <i>Tampilan Hasil Payload</i>	35
Gambar 4.19. <i>Tampilah Payload XSS</i>	36
Gambar 4.20. <i>Tampilan Hasil Payload Variasi 1</i>	36
Gambar 4. 21. <i>Tampilan Payload XSS</i>	37
Gambar 4.22. <i>Tampilan Hasil Payload Variasi 2</i>	37

Gambar 4.23. Tampilan Implementasi Kode Script	38
Gambar 4.24. Tampilan Implementasi Link pada <i>Browser</i>	39
Gambar 4.25. Grafik <i>Scoring</i>	43

DAFTAR TABEL

	Halaman
Tabel 2.1. Penelitian Terkait	4
Tabel 2.2. Dokumen OWASP	13
Tabel 2.3. Simbol-Simbol Pada <i>Flowchart</i> (Munawar, 2005)	14
Tabel 3.1. Spesifikasi Perangkat Keras (<i>Hardware</i>)	19
Tabel 3.2. Spesifikasi Perangkat Lunak (<i>Software</i>)	19
Tabel 4.1. Informasi <i>Netcraft</i>	24
Tabel 4.2. Hasil Analisis	40
Tabel 4.3. <i>Vulnerability Assessment</i>	43

ABSTRAK

ANALISIS *VULNERABILITY* PADA *WEBSITE* SMPK BINTANG LAUT KOTA TERNATE MENGGUNAKAN METODE *VULNERABILITY ASESSMENT* (Berdasarkan Panduan OWASP 10)

Mahani Albaar¹, Salkin Lutfi², Amal Khairan³

Program Studi Informatika, Fakultas Teknik, Universitas Khairun

Jl. Jati Metro, Kota Ternate Selatan

Email: mahanielbaar@gmail.com¹, salkin.lutfi@unkhair.ac.id², 4malkhairan@unkhair.ac.id³

Perkembangan teknologi sekarang ini membuat sebagian besar orang maupun organisasi mengandalkan *website* untuk menyebarkan dan memperoleh informasi secara cepat. Pentingnya peranan *website* dalam penyebaran informasi membuat perlunya keamanan tinggi agar terhindar dari aksi serangan *hacking*. Saat ini *hacking* lebih mudah dilakukan dikarenakan telah banyak tersedia alat yang memudahkan kegiatan tersebut. Siapapun dengan wawasan dengan teknologi informasi terbatas dapat melakukan *hacking*. *Vulnerability Assessment* merupakan kerangka kerja konseptual menyeluruh yang pilih, termasuk definisi kerentanan yang menentukan resiko untuk pengukuran. Hal ini juga tergantung pada tujuan pengguna hasil penilaian, yang dapat berkisar dari niat untuk menginformasikan kebijakan internasional atau untuk mengacu tindakan ditingkat masyarakat. Berdasarkan hasil *scanning* dan *pentest* yang dilakukan didapatkan beberapa celah keamanan yang ada akan di analisa yaitu *Sql Injection*, *Croos Site Scripting*, *click jacking* dan *Hidden file found* pada *website* smp katolik bintang laut Penelitian yang telah dilakukan ini membuktikan bahwa dengan menerapkan metode *Vulnerability Assessment* dapat menemukan celah keamanan pada *website* sekolah Smp Bintang Laut Kota Ternate, celah yang ditemukan pada *website* SMP Bintang Laut Kota Ternate yang perlu segera diperbaiki yaitu pada kerentanan *SQL Injection*, *Cross Site Scripting*, *Click Jacking* dan *Hidden file found*.

Kata kunci: *Vulnerability Assessment*, SMPK Bintang Laut, Analisis.

Abstract

ANALYSIS OF VULNERABILITIES ON SMPK BINTANG LAUT TERNATE WEBSITE USING THE VULNERABILITY ASSESSMENT METHOD (BASED ON OWASP TOP 10 GUIDELINES)

The rapid development of technology today has led most individuals and organizations to rely on websites for the swift dissemination and acquisition of information. The crucial role of websites in information dissemination necessitates high security to avoid hacking attacks. Currently, hacking has become easier due to the widespread availability of tools that facilitate such activities. Anyone with limited knowledge of information technology can engage in hacking. Vulnerability Assessment is a comprehensive conceptual framework that includes the definition of vulnerabilities that determine risks for measurement. This also depends on the objectives of the assessment users, which can range from intentions to inform international policy to actions at the community level. Based on the results of scanning and penetration testing, several security vulnerabilities were identified and will be analyzed, including SQL Injection, Cross-Site Scripting, Clickjacking, and Hidden files found on the SMP Katolik Bintang Laut website. The research conducted proves that by applying the Vulnerability Assessment method, security vulnerabilities on the SMP Bintang Laut Kota Ternate school website

can be identified. The vulnerabilities found on the SMP Bintang Laut Kota Ternate website that need immediate attention are SQL Injection, Cross-Site Scripting, Clickjacking, and Hidden file found.

Keywords: *Vulnerability Assessment, SMPK Bintang Laut, Analysis.*

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi sekarang ini membuat sebagian besar orang maupun organisasi mengandalkan *website* untuk menyebarkan dan memperoleh informasi secara cepat. Pentingnya peranan *website* dalam penyebaran informasi membuat perlunya keamanan tinggi agar terhindar dari aksi serangan *hacking*. saat ini *hacking* lebih mudah dilakukan dikarenakan telah banyak tersedia alat yang memudahkan kegiatan tersebut. Siapapun dengan wawasan dengan teknologi informasi terbatas dapat melakukan *hacking*.

Indonesia menjadi salah satu negara yang paling sering menjadi target serangan siber pada 2017 yang lalu. Menurut data yang dihimpun Kasprey, dalam periode 1-7 Juli 2017 Indonesia mendapatkan 902.559 serangan *cyber* jaringan. Hal ini menunjukkan bahwa *cyber security* (keamanan siber) masih menjadi tantangan yang cukup besar, jika Indonesia ingin menjadi negara dengan ekonomi digital terbesar di Asia Tenggara pada tahun 2025 nanti (Budi 2021).

SMPK Bintang Laut Kota Ternate merupakan salah satu Sekolah Menengah Pertama Negeri yang berada di Jl. Salim Fabanyo, Muhajirin, Ternate Tengah, Kota Ternate, Maluku Utara. Kabupaten Ternate, Provinsi Maluku Utara. SMPK Bintang Laut Kota Ternate merupakan sekolah yang menyediakan informasi kepada siswa-siswi melalui sistem informasi berbasis web. Adapun beberapa tampilan halaman yang ada pada website SMPK Bintang Laut Kota Ternate yaitu profil, berita, informasi, galeri, infografis dan suara anda.

Dalam wawancara yang dilakukan dengan Ibu Getrudis Jaja William selaku pengelola *website* smpk Bintang Laut sekaligus Kepala sekolah mengatakan bahwa *website* smp

bintang laut Ternate sudah pernah terjadi peretasan yang disebabkan beberapa *bug* pada celah keamanan *website* sekolah tersebut. dengan demikian perlu dilakukan pengujian keamanan sistem mengingat pentingnya menjaga data-data dari ancaman peretasan. Hasil wawancara dalam pelayanan SMPK Bintang Laut Kota Ternate yang menyediakan.

Maka dari permasalahan tersebut penulis menawarkan solusi yaitu dengan menganalisa kerentanan dari *website* menggunakan *tools* *Owapszap*, dalam analisa tersebut akan diperoleh berbagai macam kerentanan yang memungkinkan penyerang masuk ke dalam *website* SMPK Bintang Laut Kota Ternate kemudian peneliti akan merekomendasikan dari hasil analisa kewanaman *website* tersebut berdasarkan panduan *Owasp* 10.

Berdasarkan uraian di atas maka penulis melakukan penelitian dengan judul "*Analysis Vulnerability* pada *Website* SMPK Bintang Laut Kota Ternate menggunakan metode *Vulnerability Assessment* berdasarkan panduan *owasp* 10).

1.2. Rumusan masalah

Dari penjelasan latar belakang di atas, dapat di ambil rumusan masalah adalah bagaimana hasil analisis kerentanan *website* SMPK Bintang Laut dengan menggunakan *vulnerability assessment*?

1.3. Batasan Masalah

Adapun batasan masalah yang dapat diklasifikasikan sebagai berikut:

1. Tempat penelitian di SMPK Bintang Laut Kota Ternate.
2. Tidak melakukan perbaikan terhadap *website*, hanya merekomendasikan perbaikan pada *website* tersebut.

1.4. Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah untuk mengetahui hasil analisis

kerentanan *website* SMPK Bintang Laut dengan menggunakan *vulnerability assessment*.

1.5. Manfaat Penelitian

Manfaat yang diharapkan dari penelitian ini adalah:

1. Meningkatkan keamanan pada layanan *website*.
2. Dapat menambah pengetahuan dan pemahaman tentang keamanan sistem.
3. Dapat membantu suatu instansi dalam keamanan sistem.

1.6. Sistematika Penulisan

Untuk memudahkan pembahasan dalam skripsi ini, sistematika penulisan dibagi menjadi 5 (lima) bab terdiri dari:

BAB I PENDAHULUAN

Terdiri dari latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Pada bab ini akan menjelaskan mengenai teori yang menjadi dasar dan mendukung penulisan.

BAB III METODE PENELITIAN

Bab ini membahas tentang metode penelitian yang telah dilakukan oleh penulis menggunakan metode *Vulnerabilty Assesment*.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini membahas hasil dari penelitian pada *Website* SMP Bintang Laut Kota Ternate yang dilakukan oleh peneliti dengan menerapkan metode *Vulnerability Asessment*.

BAB V PENUTUP

Pada bab ini menejelaskan hasil dari penelitian dilakukan berupa kesimpulan dan saran.

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Penelitian Berikut ini adalah ulasan dari penelitian yang berkaitan dengan penelitian yang dilakukan, dapat dilihat pada tabel 2.1.

Tabel 2.1 Penelitian Terkait

No	Peneliti	Judul Penelitian dan Tahun	Metode Penelitian	Kelebihan dan Kekurangan
1	Harry Purmanta Siagian, M. Akbar, Andri	Pada tahun 2017 dengan judul " <i>Vulnerability Assessment</i> pada Wwww.Binadarma.Ac.Id	Penelitian menggunakan: 1. Metode deskriptif dimana metode ini bertujuan untuk deskripsi, gambaran secara sistematis dan <i>actual</i> . 2. Metode <i>white bos</i> yang akan menguji secara <i>structural</i> . 3. Metode <i>penetration testing</i> yang menggunakan sebagai pengukuran kerentangan.	Kelebihan dapat memberikan kontribusi saran perbaikan pada system web server kekurangan penelitian ini hanya sebatas pengujian saja.
2	Aufan Imron Rosadi	Analisis Keamanan sistem Informasi Akademik dengan Web <i>Penetration Testing</i> pada tahun 2016	Penelitian menggunakan; 1. Bagaimana menganalisa tingkat keamanan system informasi akademik di Universitas XYZ dengan menggunakan metode Web <i>Penetration Testing</i> . 2. Mencari pengetahuan dasar yang didapat dari buku, jurnal dan lainnya sebagai acuan dasar dalam menyelesaikan penelitian ini. 3. Penetrasi menggunakan teknik serangan yang hanya untuk mengetahui informasi database yang dimiliki oleh system tersebut. Jenis serangann	Kelebihan dari penelitian ini adalah sistem ini telah aman untuk mendeteksi SQL <i>Injection</i> kekurangannya untuk jenis serangan lain penelitian ini belum disimpulkan aman.

			ini biasa disebut SQL Injection.	
3	Fauzan Masykur	Analisa <i>Vulnerability Web Based Appication</i> Menggunakan Nessus pada Tahun 2015	Penelitian menggunakan: 1. Analisa <i>Vulnerability</i> dengan cara menscan <i>Web Application</i> yang telah di unggah di web server kemudian di scan menggunakan <i>vulnerability scanner</i> yakni Nessus. 2. <i>Web application</i> yang akan di analisa adalah sebuah <i>web application system online</i> yang di rancang menggunakan bahasa pemograman PHP dan database MySQL.	Kelebihannya <i>web application</i> yang dievaluasi menggunakan <i>Nessus Scanner</i> terdapat beberapa celah bagi <i>attacker</i> untuk bisa menguasai <i>web application</i> yang telah dibuat oleh <i>administrator</i>. sebuah <i>web application</i> bisa di evaluasi mengenai kelemahan-kelemahan yang terjadi sebelum sebuah <i>web application</i> tersebut di <i>upload</i> pada sebuah web server. Kekurangannya <i>web application</i> ujian <i>online</i> ini sudah digunakan namun hanya bisa diakses di jaringan intranet.
4	Yunanri. W, Imam Riadi dan Yudhana	Pada tahun 2018 Analisis Deteksi <i>Vulnerability</i> pada <i>Web Server Open Jurnal System</i> Menggunakan OWAPS Scanner	Penelitian ini menggunakan: 1. Metode pengumpulan data dilakukan dengan mengdopsi teknik Penetrasi Testing. 2. Alur sistematika ini terdiri dari studi <i>literature</i>, analisis dan ujicoba pada <i>webserver Open Jurnal System</i>, secara <i>localnetwork</i>.	Pada pengujian di lab ditemukan beberapa kerentanan dalam <i>open Jurnal System</i> yang dapat memanipulasi <i>file local</i>, mengunggah <i>file</i> dengan melakukan serangan <i>Cross-Site Scripting (XSS)</i>.

5	Yunanri. W, Imam Riadi dan Yudhana	Analisis Keamanan Website Open Journal System Menggunakan Metode Vulnerability Assessment	1. Metode yang digunakan adalah metode <i>penetration testing</i> yang berfokus pada <i>vulnerability assessment</i>. 2. <i>BlackBox</i> adalah merupakan jenis pengujian sistem tanpa mengetahui struktur rancang bangun pada sebuah sistem. Pengujian dilakukan pada <i>open journal system</i> (OJS) dan informasi mengenai jaringan maupun informasi lainnya harus dicari sendiri oleh penguji.	Hasil pengujian pada OJS diperoleh <i>tool</i> OWASP kerentanan <i>high</i> 70, kerentanan <i>medium</i> 1919 dan 4050 kerentanan <i>low</i>. Total celah atau <i>vulnerability</i> yang ditemukan berjumlah 6049. Hasil pengujian yang dilakukan Menunjukkan bahwa pada OJS versi 1.4.7 memiliki banyak celah atau kerentanan tidak di rekomendasikan untuk di gunakan, gunakanlah versi terbaru yang dikeluarkan oleh pihak OJS <i>Public knowledge project</i> (PKP)
---	------------------------------------	---	--	--

2.2. Pengertian Analisis

Panjang Kata analisis diadaptasi dari bahasa Inggris "*analysis*" yang secara etimologis berasal dari bahasa Yunani kuno yaitu "*Analusis*". Kata 'analisis' memiliki arti menguraikan kembali. Berdasarkan Kamus Besar Bahasa Indonesia (KBBI), analisis merupakan penyelidikan terhadap suatu peristiwa (karangan, perbuatan, dan sebagainya) untuk mengetahui keadaan yang sebenarnya (sebab-musabab, duduk perkaranya, dan sebagainya). Analisis juga bisa diartikan sebagai penjabaran sesudah dikaji sebaik-baiknya, ataupun pemecahan persoalan yang dimulai dengan dugaan akan kebenarannya Kata analisis sering sekali ditemukan dalam berbagai bidang ilmu, salah satunya adalah analisis

pada sistem informasi. Dalam bidang sistem informasi, analisis diperlukan sebelum dirancangnya sebuah sistem. Analisis pada sistem dilakukan untuk mengidentifikasi permasalahan pada rancangan sistem agar dapat ditemukan solusi untuk membangun sistem yang lebih efektif dan efisien. Langkah-langkah yang harus dilakukan dalam melakukan analisis sistem informasi yaitu mengidentifikasi permasalahan, menentukan dan memahami pola kerja sistem, dan menganalisis kesalahan yang terjadi (Prasetya 2023).

2.3. Keamanan Sistem informasi

Sistem keamanan informasi (*information security*) memiliki empat tujuan sangat mendasar adalah: (Nabila 2023).

1. Kerahasiaan (*confidentiality*): informasi pada sistem komputer terjamin kerahasiaannya, hanya dapat diakses oleh pihak-pihak yang diotorisasi, keutuhan serta konsistensi data pada sistem tersebut tetap terjaga. Sehingga upaya orang-orang yang ingin mencuri informasi tersebut akan sia-sia.
2. Ketersediaan (*Availability*): menjamin pengguna yang sah untuk selalu dapat mengakses informasi dan sumber daya yang diotorisasi. Untuk memastikan bahwa orang-orang yang memang berhak untuk mengakses informasi.
3. Integritas (*Integrity*): menjamin konsistensi dan menjamin data tersebut sesuai dengan aslinya, sehingga upaya orang lain yang berusaha merubah data akan segera dapat diketahui.
4. Penggunaan yang sah (*Legitimate Use*): menjamin kepastian bahwa sumber daya tidak dapat digunakan oleh orang yang tidak berhak.

2.4. Website

Website merupakan sekumpulan halaman yang berisi data dan informasi yang

disediakan melalui internet yang dapat diakses tanpa adanya batasan waktu dan tempat serta dipublikasikan baik oleh perorangan maupun organisasi. *Website* dapat menyediakan banyak informasi seperti berita, politik, hukum, budaya, hiburan, ekonomi, dan masih banyak informasi lainnya yang dapat disediakan, *website* menampilkan informasi dalam tulisan *text*, gambar, *audio*, bahkan *video*.

2.5. Bentuk-bentuk Ancaman dari Sistem Komputer

Adapun bentuk-bentuk ancaman dari sistem komputer diantaranya yaitu :(Rendro 2020)

1. Interupsi (*interruption*), Interupsi merupakan bentuk ancaman terhadap ketersediaan, yang ana data rusk sehingga tidak dapat diakses bahkan digunakan lagi. Perusakan fisik, contohnya: perusakan pada *Hardisk*, perusakan pada media penyimpanan yang lainnya, serta pemotongan kabel jaringan. Perusakan non fisik, contohnya: pengapusan suatu file-file tertentu dari sistem komputer.
2. Intersepsi (*Interception*), interepsi merupakan bentuk sebuah ancaman terhadap kerahasiaan atau *secrecy*, dimana pihak yang tidak berhak berhasil mendapatkan hak akses untuk membaca suatu data atau informasi dari suatu sistem komputer tindakan yang dilakukan dapat berupa melalui penyadapan data yang ditransmisikan melalui jalur *public* atau umum yang dikenal dengan istilah *Wrietapping* dalam *Wired Networking*, yang menggunakan kabel sebagai media dari trasmisi data.
3. Modifikasi (*Modification*), modifikasi merupakan sebuah bentuk dari ancaman terhadap integritas (*integrity*), dimana pihak yang tidak berhak berhasil mendapatkan hak akses dalam mengubah suatu data ataupun informasi dari suatu sistem komputer. Data atau informasi yang diubah tersebut berupa *record* dar suatu tabel

yang terdapat *pada file database*.

4. Pabrikasi (*fabrication*), pabrikasi adalah suatu bentuk ancaman terhadap integritas. Tindakan yang dilakukan adalah dengan meniru dan juga memasukkan suatu objek ke dalam sistem komputer. Objek yang dimasukkan biasanya berupa suatu *file* ataupun *record* yang disisipkan atau diletakkan pada suatu program aplikasi.

2.6. Vulnerability Assessment (VA)

Vulnerability Assessment merupakan kerangka kerja konseptual menyeluruh yang pilih, termasuk definisi kerentanan yang menentukan resiko untuk pengukuran. Hal ini juga tergantung pada tujuan pengguna hasil penilaian, yang dapat berkisar dari niat untuk menginformasikan kebijakan internasional atau untuk mengacu tindakan ditingkat masyarakat.

Vulnerability assessment adalah melakukan identifikasi vulnerability dari suatu aplikasi system operasi dan infrastruktur jaringan. *Vulnerability assessment* sangat penting karena memberikan informasi kepada perusahaan tentang resiko dan kelemahan keamanan sistem. Sedangkan *vulnerability* adalah suatu kelemahan dalam desain sistem, implementasi sistem atau operasi dan manajemen yang dapat dimanfaatkan untuk melanggar kebijakan keamanan sistem. *Vulnerability assessment* lebih focus untuk menemukan beragam *public vulnerability* pada seluruh system komputer dalam jaringan target (Mulyanto 2021).

2.6.1. Reconnaissance

Reconnaissance adalah tahap kegiatan dimana penyerang mengumpulkan informasi sebanyak mungkin mengenai target. Informasi yang diperoleh dari hasil kegiatan ini berupa informasi dasar yang berguna, seperti : *IP addres, topology network, networking resources*

dan informasi personal tentang *user* yang diperlukan untuk tahap selanjutnya (Wahyudi 2020).

2.6.2. Scanning

Hacker akan mencari berbagai kemungkinan yang bisa digunakan untuk mengambil alih komputer korban. *Hacker* bisa mencari jalan masuk untuk menguasai komputer korban. Berbagai *tool* biasanya digunakan oleh *hacker* dalam membantu proses pencarian ini. Namun, seorang *hacker* profesional tidak hanya mengandalkan sebuah *tool*, mereka juga bisa mencari secara manual untuk hal-hal yang tidak bisa dilakukan oleh sebuah *tools* (Rustianto 2010).

2.7. Penetration Testing

Penetration Testing adalah tindakan mengevaluasi keamanan jaringan dengan cara mensimulasikan serangan *hacking*. *Penetration Testing* berfokus pada *vulnerability* yang mengizinkan eksekusi perintah. Hampir sebagian besar *vulnerability* jenis tersebut adalah *buffer overflows*. *Penetration text* juga menyediakan bukti nyata terhadap masalah yang terjadi pada sistem (Rustianto 2010).

2.7.1. Jenis-Jenis Serangan

Adapun jenis-jenis dari serangan dapat dilihat sebagai berikut:

1. SQL Injection

SQL Injection merupakan sebuah bahasa yang digunakan untuk mengakses suatu basis data, sedangkan kata *injection* jika diterjemahkan memiliki arti menyuntik *SQL Injection* adalah sebuah metode untuk memasukkan perintah *SQL* sebagai *input* melalui sebuah *web* guna mendapatkan akses *database* (Yelvita 2022). Dapat dilihat pada gambar 2.1.



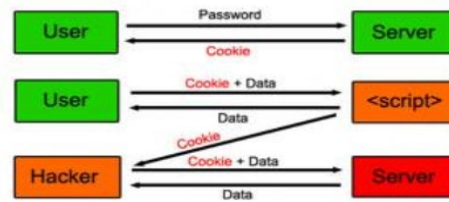
Gambar 2.1 SQL Injection

2. ClickJacking

Clickjacking adalah teknik serangan yang menyamarkan suatu elemen *website* guna mengelabui pengguna. *Hacker* biasanya menggunakan elemen HTML atau beberapa lapisan transparan untuk menampilkan *invisible page* di atas halaman yang dilihat *user*. Jika elemen tak terlihat tersebut diklik, pelaku telah berhasil meretas tombol klik pengguna. Hal ini sangat berbahaya karena ada banyak potensi yang dapat merugikan user. Sebut saja seperti mengunduh *malware* tanpa sepengetahuan pengguna, pencurian data sensitif, dialihkan ke situs berbahaya, pembelian produk *online*, transfer uang, dan lain-lain.

3. Cross site Scripting (XSS)

Cross Site Scripting dapat diartikan sebagai kelemahan yang terjadi akibat ketidakmampuan *server* dalam memvalidasi *input* yang diberikan oleh *user*. Algoritma, yang digunakan untuk pembuatan halaman yang diinginkan, tidak mampu melakukan penyaringan terhadap *input* tersebut. Hal ini memungkinkan halaman yang dihasilkan menyertakan perintah yang sebenarnya tidak diperbolehkan. *Cross Site Scripting* merupakan kelemahan yang populer untuk dieksploitasi. Namun sayangnya, banyak penyedia layanan yang tidak mengakui kelemahan tersebut dan melakukan perubahan pada sistem yang digunakan. Adapun gambar *cross site scripting* (XSS) dapat dilihat pada gambar 2.2.



Gambar 2.2 XSS

4. *Hidden File Found*

Ponsel dan *computer* menyimpan lebih dari yang mungkin anda sadari. *File* yang dapat Anda lihat secara *default* di *Windows*, *macOS*, *Android*, dan *iOS* bukanlah semua yang disimpan di sistem tersebut. File tersembunyi ini biasanya digunakan oleh sistem operasi dan aplikasi yang Anda jalankan untuk menyimpan data yang biasanya tidak Anda perlukan aksesnya bahkan data yang dapat mengganggu kelancaran perangkat Anda jika diedit dengan cara yang salah atau dihapus.

2.8. **Open Web Application Security Project (OWASP)**

OWASP (Open Web Application Security Project) merupakan komunitas terbuka diseluruh dunia yang berfokus pada peningkatan keamanan aplikasi perangkat lunak. Hingga saat ini, panduan pengujian *OWASP* telah mencapai versi 4.0. Panduan pengujian *OWASP* versi 1.0 pertama kali dipublikasikan pada bulan Desember tahun 2004. Pada tanggal 25 Desember 2006, panduan pengujian *OWASP* versi 2.0 pun dirilis. Selang 2 tahun tepatnya pada tanggal 15 September 2008, *OWASP* merilis panduan pengujian *OWASP* versi 3.0 (Yudiana, 2021). *OWASP* kembali merilis panduan pengujian *OWASP* versi 4.0 pada tahun 2014. Pada tahun 2003, untuk pertama kalinya Yayasan *OWASP* merilis 10 daftar resiko keamanan aplikasi website yang paling kritis. Daftar tersebut disebut *OWASP TOP 10*. Pada tahun 2004, untuk kedua kalinya yayasan *OWASP* merilis *OWASP TOP 10–2004*. Seiring dengan perkembangan teknologi dan meningkatnya kejahatan siber,

Yayasan OWASP kembali merilis OWASP TOP 10 pada tahun 2007, 2010, 2013, dan terakhir pada tahun 2017. OWASP TOP 10 tidak hanya berisi daftar resiko keamanan aplikasi *website* yang paling kritis, namun juga berisi tentang penjelasan daftar resiko tersebut, cara mencegah celah keamanan tersebut, dan contoh skenario dari serangan tersebut (Rosaliah, 2021).

Dokumen OWASP adalah suatu pengujian yang mengikuti pedoman pedoman yang telah di susun sedemikian rupa untuk melakukan pengetesan pada website, menurut Eoin Keary panduan pengujian OWASP memiliki peran penting dalam memecahkan masalah serius ini. Sangat penting bahwa pendekatan kami untuk menguji perangkat lunak dari masalah keamanan didasarkan pada prinsip-prinsip teknik dan sains, kami membutuhkan pendekatan yang konsisten dan berulang, dapat dilihat pada tabel 2.2.

Tabel 2.2 Dokumen OWASP

No	Pedoman Dokumen OWASP
1.1	<i>Foreword</i>
2.1	<i>Frontispiece</i>
3.1	<i>Introduction</i>
4.1	<i>The OWASP Testing Framework</i>
4.1.1	<i>The Web Security Testing F</i>
4.1.2	<i>Penetration Testing Methodologies</i>
5.1	<i>Web Application Security Testing</i>
5.1.1	<i>Information Gathering</i>
5.1.2	<i>Identify Management Testing</i>
5.1.3	<i>Authentication Testing</i>
5.1.4	<i>Authorization Testing</i>
5.1.5	<i>Session Management Testing</i>
5.1.6	<i>Input Validation Testing</i>
5.1.7	<i>Testing For Error Handling</i>
5.1.8	<i>Testing For Weak Cryptography</i>
5.1.9	<i>Business Logic Testing</i>
5.1.10	<i>Client-Side-Testing</i>
5.1.11	<i>API Testig</i>
6.1	<i>Reporting</i>

Pada tabel 2.2 dijadikan sebagai pedoman dalam melakukan pengujian keamanan

pada *website* mengikuti *Document OWASP* sehingga nantinya pengujian dapat lebih sistematis dan terstruktur.

2.9. SMPK Bintang Laut Kota Ternate

SMPK Bintang Laut Kota Ternate merupakan salah satu Sekolah Menengah Pertama Katolik Negeri yang berada di Jl. Salim Fabanyo, Muhajirin, Ternate Tengah, Kota Ternate, Maluku Utara. Kabupaten Ternate, Provinsi Maluku Utara. SMPK Bintang Laut Kota Ternate merupakan sekolah yang menyediakan informasi kepada siswa-siswi melalui sistem informasi berbasis web. Dapat dilihat tampilan *website* pada gambar 2.3.

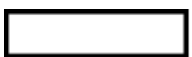


Gambar 2.3 Profil SMPK Bintang Laut Kota Ternate

2.10. Flowchart

Flowchart merupakan sekumpulan simbol-simbol atau skema yang menunjukkan atau menggambarkan rangkaian kegiatan program dari awal sampai akhir. Inti dari pembuatan *flowchart* ini adalah penggambaran dari urutan langkah-langkah pekerjaan dari suatu algoritma (Munawar, 2005). Simbol *flowchart* diagram alir bisa dilihat pada tabel 2.3.

Tabel 2.3 Simbol-Simbol Pada *Flowchart* (Munawar, 2005)

No	Simbol	Fungsi
1		Terminal, untuk memulai dan mengakhiri suatu proses/kegiatan.
2		Proses, suatu yang menunjukkan setiap pengolahan yang dilakukan oleh <i>computer</i> .

3		<i>Input</i> , untuk memasukkan hasil dari suatu proses.
4		<i>Decision</i> , suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan.
5		<i>Display, output</i> yang ditampilkan dilayar terminal.
6		<i>Connector</i> , suatu prosedur akan masuk atau keluar melalui simbol ini dalam lembar yang sama.
7		<i>Off Page Connector</i> , merupakan <i>symbol</i> masuk atau keluarnya suatu prosedur pada kertas lembar lain.
8		<i>Arus Flow</i> , simbol ini digunakan untuk menggambarkan arus proses dari suatu kegiatan lain.
9		<i>Hard Disk Storage</i> , <i>input/output</i> yang menggunakan <i>hardisk</i> .
10		<i>Stored Data</i> , <i>input/output</i> yang menggunakan <i>disket</i> .
11		<i>Printer</i> , simbol ini digunakan untuk menggambarkan suatu dokumen atau kegiatan mencetak suatu informasi dengan mesin printer.

2.11. Owaps Zap

Zed Attack Proxy (ZAP) adalah aplikasi untuk melakukan pentest untuk menemukan *vulnerabilities* dalam suatu *web applications* dengan cara mudah, ZAP menyediakan *scanner automatis* sebaik bila kita menggunakan *tool* untuk menemukan *vulnerabilities* secara manual. Ketika digunakan sebagai *server proxy*, ini memungkinkan pengguna untuk memanipulasi semua lalu lintas yang melewatinya, termasuk lalu lintas menggunakan *https*, itu juga dapat berjalan dalam mode daemon yang kemudian dikontrol melalui Rest API. ZAP telah ditambahkan ke dalam Radar Teknologi *ThoughtWorks* pada 30 Mei 2015 di cincin Percobaan. ZAP awalnya bercabang dari Paros, *proxy* pentesting lainnya. Simon Bennetts, pemimpin proyek, menyatakan pada tahun 2014 bahwa hanya 20% dari kode sumber ZAP masih dari Paros. Owaps zap melakukan beberapa fungsi keamanan termasuk:

1. Memindai permintaan web secara pasif.
2. Menggunakan daftar kamus untuk mencari *file* dan *folder* di server web.

3. Menggunakan perayap untuk mengidentifikasi struktur situs dan mengambil semua tautan dan URL.
4. Mencegat, menampilkan, memodifikasi, dan meneruskan permintaan web antara browser dan aplikasi web.

OWASP ZAP dapat mengidentifikasi kerentanan dalam aplikasi web termasuk *autentikasi* yang disusupi, pemaparan data sensitif, kesalahan konfigurasi keamanan, *injeksi* SQL, skrip lintas situs (XSS), deserialisasi tidak aman, dan komponen dengan kerentanan yang diketahui (Kusuma 2022).

BAB III

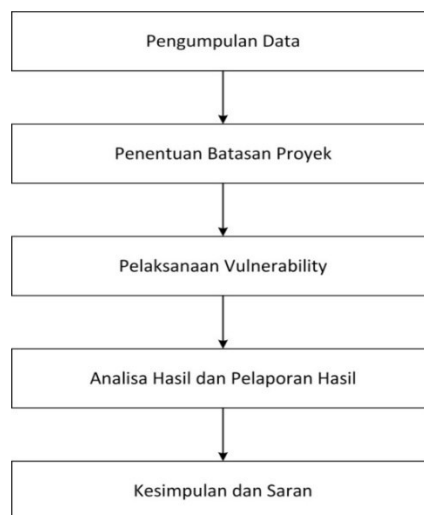
METODE PENELITIAN

3.1. Tempat dan Waktu Penelitian

Penelitian ini dilakukan di SMPK Bintang Laut Kota Ternate Provinsi Maluku Utara dan Waktu penelitian selama 2 bulan.

3.2. Tahapan Penelitian

Adapun tahapan penelitian dapat dilihat pada gambar 3.1.



Gambar 3.1 Tahapan Penelitian

Ada 5 tahapan dalam penelitian ini adalah sebagai berikut:

1. Pengumpulan data

Tahapan pengumpulan data diperlukan dengan tiga metode yaitu: Studi pustaka, dengan mengumpulkan data dari jurnal yang berhubungan dengan penelitian sebagai pendukung pembuatan penelitian. Observasi, dengan mengamati objek yang dianalisis monitoring secara langsung. Wawancara dengan memberikan pertanyaan kepada penanggung jawab sistem informasi untuk mendapatkan data-data *website* tersebut.

2. Penentuan batasan proyek

Pada tahapan ini mengenai batasan proyek, dilakukan agar *vulnerability assessment* tidak melebihi parameter yang ditentukan. Pada penelitian ini juga penulis menentukan lingkup penelitian yang akan diteliti, pada penelitian ini penulis menggunakan *website* SMP K Bintang Laut Ternate sebagai objek penelitian yang dilakukan.

3. Pelaksanaan *Vulnerability*

Pelaksanaan *vulnerabilty* tentunya mengacu dari batasan proyek yang telah didefinisikan sebelumnya dan dari standar keamanan sistem yang telah diterapkan di lapangan. Tahap pelaksanaan meliputi tahap pengumpulan data yang dilakukan wawancara langsung dengan tim tenaga teknologi informasi yang menerapkan keamanan sistem pada suatu *website* untuk mengetahui kelemahan. Di tahap ini tools yang akan digunakan adalah *Owaps Zap* Penetration Test menggunakan *Sql Map*, *Xss strike*, dan *scrip HTML*.

4. Tahap analisa dan laporan akhir

Tahapan analisa dan laporan akhir berisi analisa rekomendasi perbaikan yang merupakan hasil dari scanning menggunakan *Owaps Zap* dan *penetration* menggunakan *Sql Map*, *Xss strike*, dan *scrip HTML*.

5. Kesimpulan dan saran

Tahapan yang terakhir menyimpulkan apa yang telah dianalisa dan saran yang diberikan untuk memperbaikinya.

3.3. Alat Penelitian

Dalam melakukan analisa kerentanan *website*, penulis membutuhkan perangkat

pendukung untuk menjalankan penelitian ini dengan baik. Alat yang akan digunakan dalam penelitian ini meliputi perangkat lunak(*software*) dan perangkat keras(*hardware*).

3.1.1. Perangkat keras (*Hardware*)

Perangkat keras ini berupa PC/*Leptop* untuk melakukan analisa kerentanan dimana spesifikasi *hardware* dapat dilihat pada tabel 3.1.

Tabel 3.1 Spesifikasi Perangkat Keras (*Hadrware*)

Komponen	Spesifikasi
<i>Processor</i>	Intel(R) Core (TM) i5-8265U CPU @ 1.60GHz 1.80 GHz
RAM	8,00 GB
<i>Storage Memory</i>	512 GB
<i>System type</i>	64-bit operating system, x64-based processor

3.1.2. Perangkat Lunak (*Software*)

Selain kebutuhan *hardware* penulis juga membutuhkan *software* untuk melakukan analisa kerentanan *website* dapat dilihat pada tabel 3.2.

Tabel 3.2 Spesifikasi Perangkat Lunak (*Software*)

Jenis	Nama	Kegunaan
<i>Operating System</i>	<i>Windows 10</i>	Operasi yang digunakan selama melakukan penelitian analisa kerentanan <i>website</i>
Aplikasi	<i>Office Word 2016</i>	Untuk pembuatan dan pengolahan data penelitian menjadi dokumen laporan
Aplikasi	<i>Firefox</i>	untuk melihat <i>website</i> yang nantinya di analisa
Aplikasi	Owaps Zap	untuk <i>scanning</i> kerentanan <i>website</i> dan mengambil <i>risk rating</i>

3.4. Metode Pengumpulan Data

Dalam penyusunan penelitian ini penulis mengumpulkan data yang dibutuhkan dalam pengujian penetrasi menggunakan metode pengumpulan data sebagai berikut:

1. Studi perpustakaan (*Literature*)

Pada tahap ini peneliti mengambil teori-teori jurnal maupun penelitian terdahulu yang berkaitan dengan analisis *vulnerability* yang dapat mendukung laporan ini.

2. Pengamatan (*Observation*)

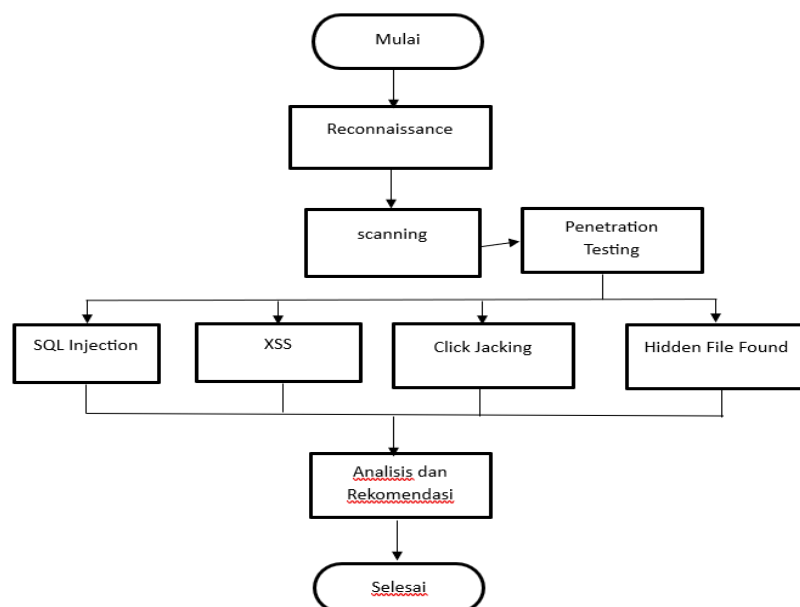
Pada tahapan ini peneliti melakukan peninjauan langsung ke *website* untuk dilakukan *Vulnerability* khususnya bagian teknis yang merupakan unit sistem informasi dengan pemilihan, perubahan, pencatatan dan pengkodean serangkaian perilaku dan suasana berkenaan dengan objek penelitian.

3. Wawancara (*interview*)

Wawancara dilakukan disekolah SMPK Bintang Laut Kota Ternate, demi mendapatkan informasi dan data-data yang berhubungan dengan penelitian ini maka penulis mengajukan beberapa pertanyaan dari diskusi kepada pihak penanggung jawab sistem informasi untuk mengetahui sistem operasi yang digunakan, konfigurasi web server dan DBMS, serta serangan yang pernah terjadi pada *website* tersebut. Dan gambaran jaringan web untuk memperoleh penelitian.

3.5. Tahapan *Vulnerability Assessment*

Adapun tahapan dari *vulnerability assessment* dapat dilihat pada gambar 3.2.



Gambar 3.2 Tahapan *Vulnerability Assessment*

Pada gambar 3.2 merupakan tahapan dari *Vulnerability Assessment* dalam melakukan analisis *vulnerability* dilakukan langkah-langkah berikut:

1. *Reconnaissance*

Pada tahapan ini dilakukan wawancara kepada pihak penanggung jawab sistem informasi untuk mengetahui sistem operasi yang digunakan, konfigurasi web server dan serta serangan yang pernah terjadi pada *website* tersebut.

2. *Scanning*

Tahapan selanjutnya dilakukan scanning menggunakan *tools* Owaps Zap dilakukan untuk mencari berbagai kemungkinan celah keamanan yang terdapat *website*.

3. *Penetration Test*

Setelah proses *scanning* selesai dan berhasil mengetahui celah kelemahan yang terdapat pada *website* tersebut, dilakukan proses *penetration test* terhadap *website* dengan dilakukan teknik penyerangan apakah berhasil atau tidak, dengan menggunakan beberapa Teknik dengan mengikuti pedoman pengujian pada *Document OWASP*.

4. *SQL Injection*

Pada tahap ini pengujian akan melakukan pengujian pada *website* dengan *sql injection* menggunakan *SqlMap*.

5. *XSS*

Penguji memiliki situs web yang memiliki kerentanan yang memungkinkan injeksi skrip. Dengan memasukkan skrip php dan akan mencuri setiap *cookie* pengunjung.

6. *ClickJacking*

Penguji dapat menyelidiki apakah yang halaman di target dapat dimuat dalam bingkai

menggunakan kode HTML.

7. *Hidden File Found*

Pada tahap ini *file sensitive* diidentifikasi sebagai di akses atau tersedia yang dapat membocorkan informasi

8. Analisa dan Rekomendasi

Tahapan ini menjelaskan tentang celah keamanan yang berhasil ditemukan kemudian memberikan rekomendasi perbaikan untuk mengatasi website tersebut.

9. Hasil

tahap ini bertujuan untuk mengurangi resiko kerentanan pada suatu *website*. Tahap ini merupakan kesimpulan akhir, berupa tabel dari sebuah penelitian yang telah dilakukan penulis pada *website* SMPK Bintang Laut Kota Ternate.

3.6. Analisa Hasil dan Pelaporan Hasil

Temuan ini dapat mengetahui kelemahan dari *website*, akan tetapi temuan ini perlu dievaluasi kembali untuk melihat sejauh mana temuan gangguan atau kelemahan ini akan berdampak terhadap sistem. Kemudian merekomendasi perbaikan kepada *administrator* dari sekolah SMPK Bintang Laut Kota Ternate.

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini akan membahas hasil analisis *vulnerability* pada *website* SMPK Bintang Laut Ternate menggunakan tahapan *vulnerability assessment* yang telah dijelaskan pada bab sebelumnya yang terdiri dari *Reconnaissance*, *scanning*, *penetration testing*, analisis dan rekomendasi untuk mengurangi resiko pada *website*. Ini merupakan *website* SMPK Bintang Laut Ternate, dapat dilihat pada gambar 4.1.



Gambar 4.1 Website SMPK Bintang Laut Ternate

Berikut adalah penjelasan mengenai tahapan yang akan dilakukan berdasarkan *vulnerability assessment*.

4.1. Reconnaissance

Reconnaissance adalah tahap dimana penyerang mengumpulkan informasi sebanyak mungkin mengenai target. Informasi yang diperoleh dari hasil kegiatan ini berupa informasi dasar yang berguna, seperti: *IP Address*, *topology network*, *network resources* dan informasi personal tentang *user* yang diperlukan untuk tahap selanjutnya. disini penguji menggunakan *tools Netcraft*. *Netcraft* adalah menyediakan *server web* dan *hosting web*, termasuk *server web* dan deteksi sistem operasi, informasi *netcraft*. Dapat dilihat pada tabel

4.1.

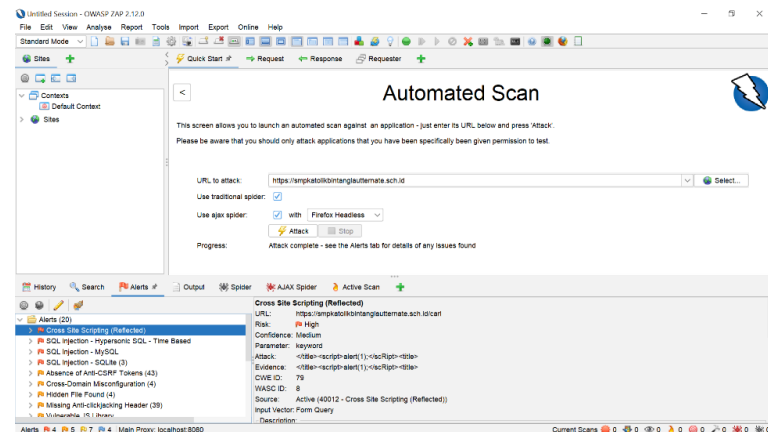
Tabel 4.1 Informasi *Netcraft*

Lokasi	https://smpkatolikbintanglautternate.sch.id
Pemilik <i>Netblock</i>	<i>Hostinger International Limited</i>
Perusahaan <i>hosting</i>	<i>Grup Hostinger</i>
Negara tuan rumah	<u>KITA</u>
alamat IPv4	195.179.238.102
sistem otonom IPv4	<u>AS47583</u>
alamat IPv6	2a02:4780:1:1162:0:2e46:6352:10
sistem otonom IPv6	<u>AS47583</u>
Domain	smpkatolikbintanglautternate.sch.id
Nama <i>server</i>	ns1.dns-parking.com
Pendaftar domain	tidak dikenal
Organisasi <i>server</i> nama	<i>whois.hostinger.com</i>
Organisasi	tidak dikenal
<i>Admin</i> DNS	<i>dns@hostinger.com</i>
Domain Tingkat Atas	Indonesia (sch.id)
Ekstensi Keamanan DNS	tidak dikenal

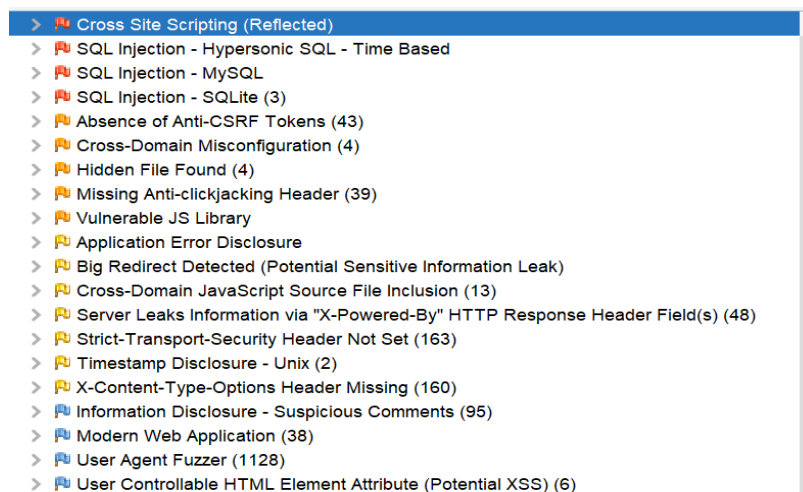
Netcraft adalah menyediakan analisis pangsa pasar *server* web dan *hosting* web, termasuk *server* web dan deteksi sistem operasi. Dalam beberapa kasus, tergantung pada sistem operasi *server* yang ditanyai, layanan mereka dapat memantau waktu *uptime*; pemantauan kinerja *uptime* adalah faktor yang umum digunakan dalam menentukan keandalan penyedia *hosting* web scanning.

4.2. **Scanning**

Tahapan selanjutnya dilakukan *scanning* menggunakan *tools* *owasp zap*, dilakukan untuk mencari berbagai kemungkinan celah keamanan yang terdapat *website*. Pengujian menggunakan *Owasp Zap* untuk mencari celah keamanan pada *website* *Owasp zap* adalah aplikasi untuk melakukan pentest untuk menemukan *vulnerabilities* dalam suatu web *applications* dengan cara mudah, ZAP menyediakan *scanner* *automatis* sebaik bila kita menggunakan *tool* untuk menemukan *vulnerabilities* secara manual. Perhatikan gambar 4.2 dan gambar 4.3.



Gambar 4.2 Tampilan OwapsZap



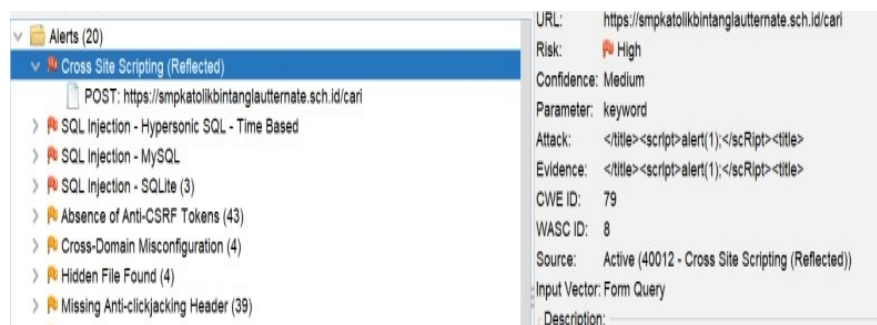
Gambar 4.3 Tampilan Hasil Scan Vulnerability

Pada gambar 4.3 merupakan hasil dari *vulnerability analysis* menampilkan informasi celah keamanan pada *website* smkatolikbintanglutterternate.sch.id. Berdasarkan hasil kerentanan yang ditemukan maka akan dilakukan pengujian pada beberapa celah yang lebih spesifik yaitu:

1. *Cross Site Scripting*

Cross-site Scripting (XSS) adalah teknik serangan yang melibatkan gema kode yang disediakan penyerang ke dalam *instance browser* pengguna. Contoh *browser* dapat berupa klien *browser* web standar, atau objek *browser* yang disematkan dalam produk perangkat

lunak seperti *browser* di dalam *WinAmp*, pembaca RSS, atau klien *email*. Kode itu sendiri biasanya ditulis dalam HTML/JavaScript, tetapi juga dapat meluas ke *VBScript*, *ActiveX*, *Java*, *Flash*, atau teknologi lain yang mendukung *browser*. Dapat dilihat pada gambar 4.4.

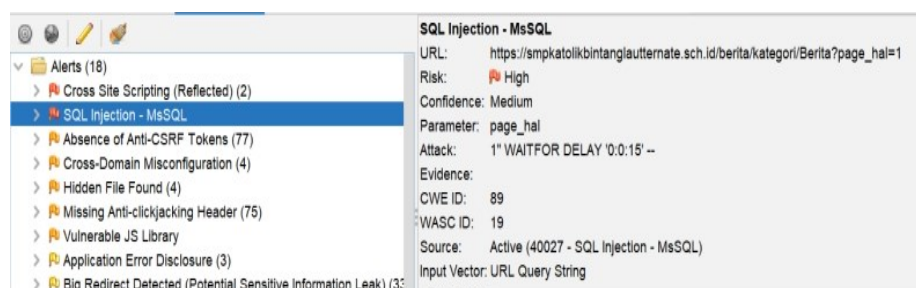


Gambar 4.4 Tampilan Hasil Scan *Cross Site Scripting*

Pada gambar 4.4 di jelaskan bahwa pada *website* SMPK Bintang Laut Ternate terdapat URL <http://smpkatolikbintanglautternate.sch.id/cari> adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko tinggi atau dampak serangan dari kerentanan itu tinggi. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu medium atau sedang. Dengan parameter *keyword* yang perlu diuji memungkinkan penyerang untuk melakukan serangan. Dan rekomendasi serangan (*attack*) `</title><script>alert(1) ;<script><title>`.

2. *Sql Injection*

Adapun hasil dari aql injection dapat dilihat pada gambar 4.5.



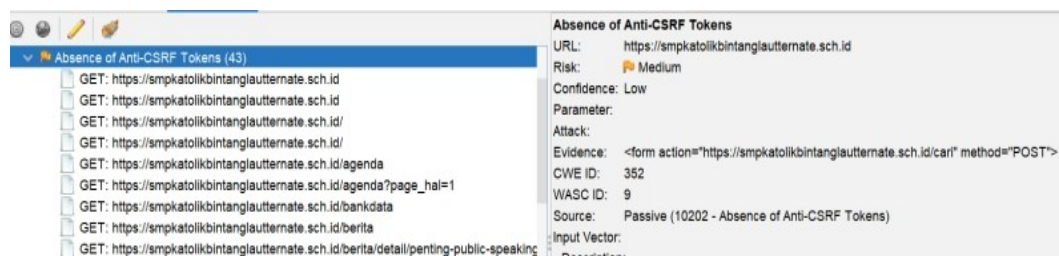
Gambar 4.5 Tampilan Hasil Scan *SQL Injection*

Pada gambar 4.5 di jelaskan bahwa pada *website* SMP Bintang Laut Ternate terdapat

URL http://smpkatolikbintanglautternate.sch.id/berita/kategori/Berita?page_hal=1 adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko tinggi atau dampak serangan dari kerentanan itu tinggi. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu medium atau sedang. Dengan parameter *page_hal* yang perlu diuji memungkinkan penyerang untuk melakukan serangan. Dan rekomendasi serangan (*attack*) 1"WAITFOR DELAY '0:0:15'--.

3. *Absence Of Anti-CSRF Token*

Pemalsuan permintaan lintas situs adalah serangan yang memaksa korban untuk mengirim permintaan HTTP ke tujuan target tanpa sepengetahuan atau niat mereka untuk melakukan tindakan sebagai korban. Dampaknya terdapat hak istimewa atau asumsikan identitas, mekanisme, proteksi *bypass*, baca data aplikasi, modifikasi data aplikasi dan serangan DoS *crash*, *exit*, atau *restart*. Dapat dilihat pada gambar 4.6.



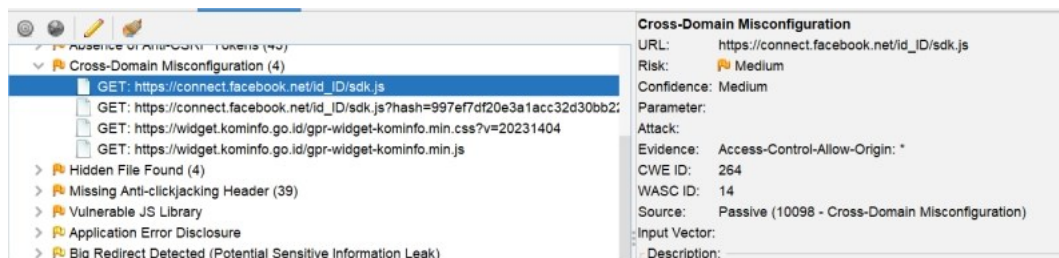
Gambar 4.6 Tampilan Hasil Scan *Absence Of Anti-CSRF Token*

Pada gambar 4.6 di jelaskan bahwa pada *website* SMP Bintang Laut Ternate terdapat URL <http://smpkatolikbintanglautternate.sch.id> adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko medium. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu *low* atau rendah.

4. *Cross-Domain Misconfiguration*

Kesalahan konfigurasi CORS di server web memungkinkan permintaan baca lintas domain dari domain pihak ketiga yang sewenang-wenang, menggunakan API yang tidak

diautentikasi pada domain ini. Implementasi *browser* web tidak mengizinkan pihak ketiga yang sewenang-wenang untuk membaca respons dari API yang diautentikasi. Ini agak mengurangi risiko. Kesalahan konfigurasi ini dapat digunakan oleh penyerang untuk mengakses data yang tersedia dengan cara yang tidak diautentikasi, tetapi menggunakan bentuk keamanan lain, seperti daftar putih alamat IP. Dapat dilihat pada gambar 4.7.

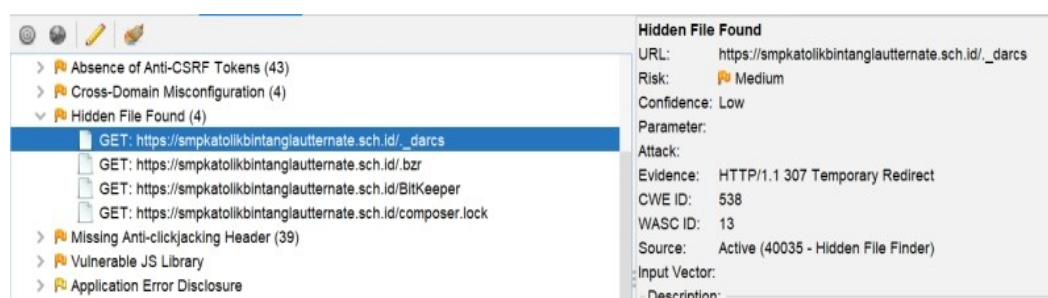


Gambar 4.7 Tampilan Hasil *Scan Cross-Domain Misconfiguration*

Pada gambar 4.7 di jelaskan bahwa pada *website* SMP Bintang Laut Ternate terdapat URL http://connect.facebook.net/id_ID/sdk.js adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko medium atau dampak serangan dari kerentanan itu medium. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu medium.

5. *Hidden File Found*

File sensitif diidentifikasi sebagai dapat diakses atau tersedia. Ini dapat membocorkan informasi administratif, konfigurasi, atau kredensial yang dapat dimanfaatkan oleh individu jahat untuk menyerang sistem lebih lanjut atau melakukan upaya rekayasa sosial. Dapat dilihat pada gambar 4.8.

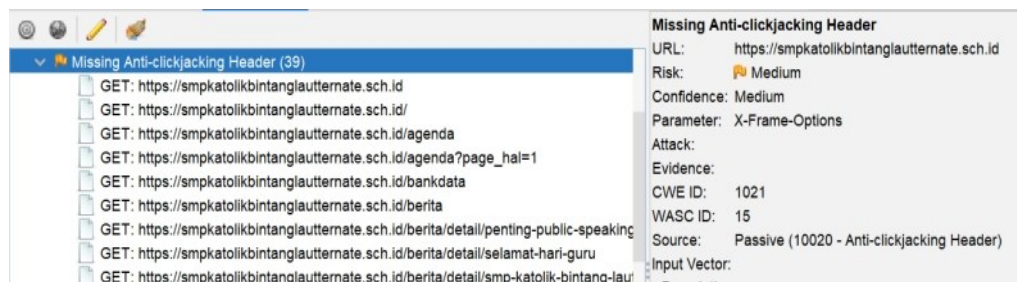


Gambar 4.8 Tampilan Hasil *Hidden File Found*

Pada gambar 4.8 di jelaskan bahwa pada *website* SMP Bintang Laut Ternate terdapat URL http://smpkatolikbintanglautternate.sch.id/_darcs adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko medium atau dampak serangan dari kerentanan itu medium. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu *low* atau rendah.

6. Missing Anti-clickjacking Header

Respons tidak menyertakan *Content-Security-Policy* dengan arahan '*frame-ancestors*' atau *X-Frame-Options* untuk melindungi dari serangan '*ClickJacking*'. Dapat dilihat pada gambar 4.9.

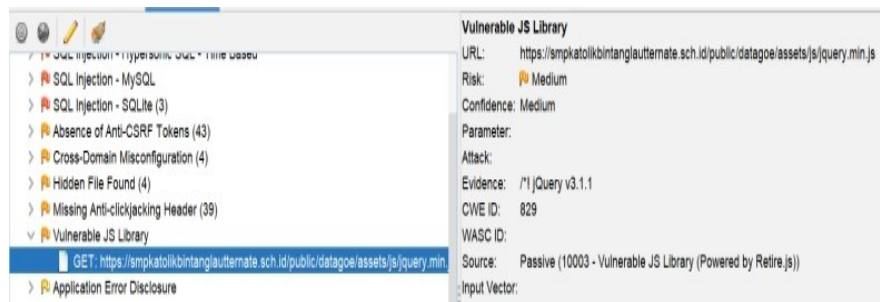


Gambar 4.9 Tampilan Hasil *Missing Anti ~click Jacking Header*

Pada gambar 4.9 di jelaskan bahwa pada *website* SMP Bintang Laut Ternate terdapat URL <http://smpkatolikbintanglautternate.sch.id> adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko medium atau dampak serangan dari kerentanan itu medium. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu medium. Dengan parameter *X-Frame-Options* yang perlu diuji memungkinkan penyerang untuk melakukan serangan.

7. Vulnerable JS Library

Jquery library yang teridentifikasi, versi 3.1.1 rentan. Untuk lebih dapat dilihat pada gambar 4.10.

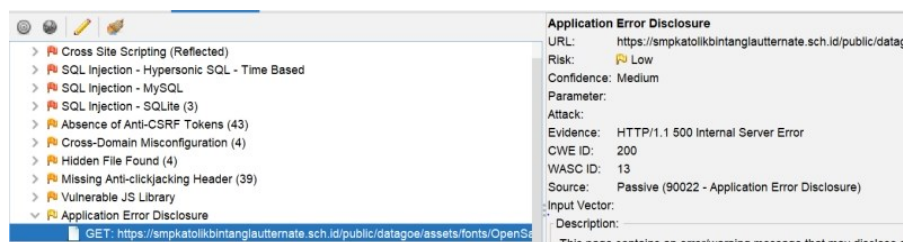


Gambar 4.10 Tampilan *Vulnerable JS Library*

Pada gambar 4.10 di jelaskan bahwa pada *website* SMP Bintang Laut Ternate terdapat URL <http://smpkatolikbintanglautternate.sch.id/public/datagoe/assets/js/jquery.min.js> adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko medium atau dampak serangan dari kerentanan itu medium. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu medium.

8. *Application Error Disclosure*

Halaman ini berisi pesan kesalahan/peringatan yang mungkin mengungkapkan informasi sensitif seperti lokasi *file* yang menghasilkan pengecualian yang tidak tertangani. Informasi ini dapat digunakan untuk meluncurkan serangan lebih lanjut terhadap aplikasi web. Lansiran bisa menjadi positif palsu jika pesan kesalahan ditemukan di dalam halaman dokumentasi. Dapat dilihat pada gambar 4.11.



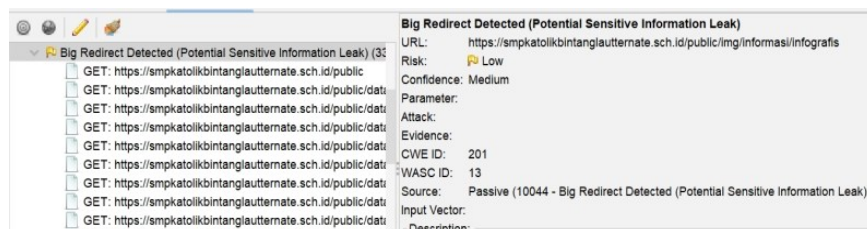
Gambar 4.11 Tampilan Hasil *Application Error Disclosure*

Pada gambar 4.11 di jelaskan bahwa pada *website* SMP Bintang Laut Ternate terdapat URL <http://smpkatolikbintanglautternate.sch.id/public/datagoe> adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko low atau dampak serangan dari

kerentanan itu lemah. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu medium.

9. *Big Redirect Detected (Potential Sensitive Information Leak)*

Server telah merespons dengan pengalihan yang tampaknya memberikan respons yang besar. Hal ini mungkin menunjukkan bahwa meskipun server mengirim pengalihan, ia juga merespons dengan konten isi (yang mungkin mencakup detail sensitif, PII, dll.). dapat dilihat pada gambar 4.12.

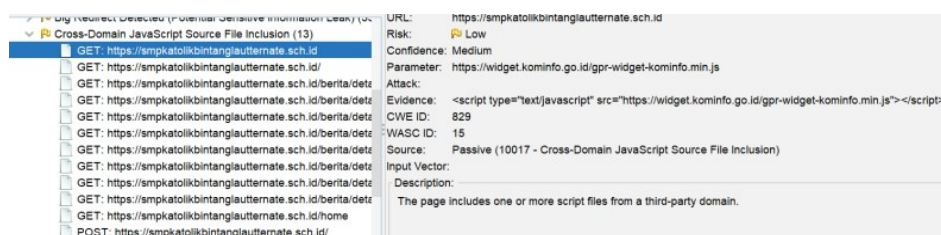


Gambar 4.12 Tampilan Hasil Scan *Big Redirect Detected*

Pada gambar 4.12 di jelaskan bahwa pada *website* SMP Bintang Laut terdapat URL <http://smpkatolikbintanglautternate.sch.id/public/img/informasi/infografis> adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko *low* atau dampak serangan dari kerentanan itu lemah. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu medium.

10. *Cross-Domain JavaScript Source File Inclusion*

Laman menyertakan satu atau beberapa *file* skrip dari domain pihak ketiga. Dapat dilihat pada gambar 4.13.

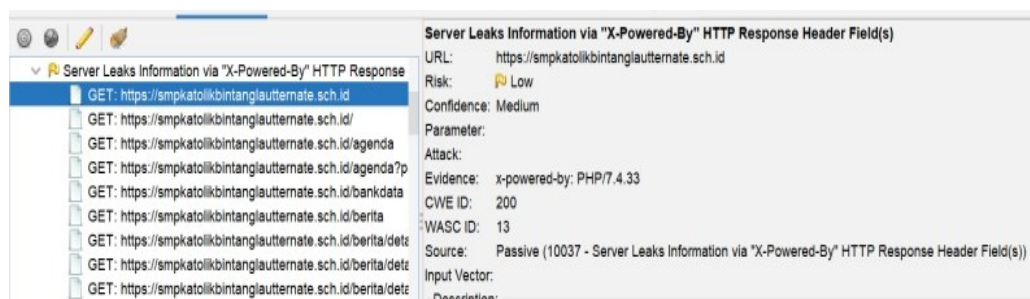


Gambar 4.13 Hasil Scan *Cross-Domain JavaScript Source File Inclusion*

Pada gambar 4.13 di jelaskan bahwa pada *website* SMP Bintang Laut Ternate terdapat URL <http://smkatolikbintanglautternate.sch.id> adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko medium atau dampak serangan dari kerentanan itu medium. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu medium. Dengan parameter <https://widget.kominfo.go.id/gpr-widget-kominfo.min.js> yang perlu diuji memungkinkan penyerang untuk melakukan serangan.

11. *Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)*

Server web/aplikasi membocorkan informasi melalui satu atau beberapa *header* respons HTTP "X-Powered-By". Akses ke informasi tersebut dapat memfasilitasi penyerang untuk mengidentifikasi kerangka kerja/komponen lain yang diandalkan oleh aplikasi web Anda dan kerentanan yang mungkin dialami oleh komponen tersebut. Dapat dilihat pada gambar 4.14.



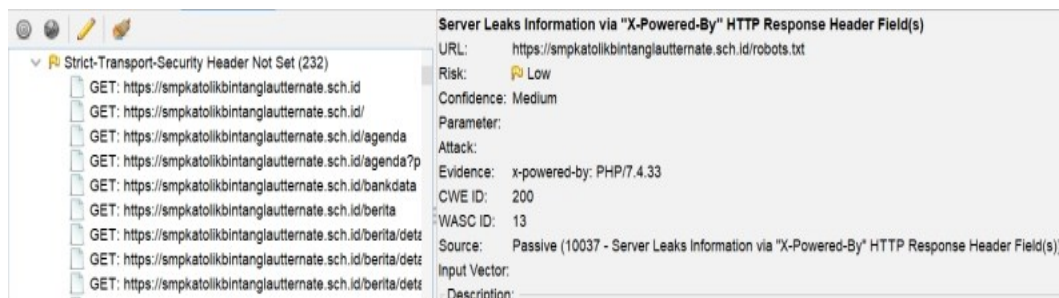
Gambar 4.14 Hasil Scan *Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)*

Pada gambar 4.14 di jelaskan bahwa pada *website* SMP Bintang Laut Ternate terdapat URL <http://smkatolikbintanglautternate.sch.id> adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko low atau dampak serangan dari kerentanan itu lemah. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu medium.

12. *Strict-Transport-Security Header Not Set*

HTTP Strict Transport Security (HSTS) adalah mekanisme kebijakan keamanan web

di mana server web menyatakan bahwa agen pengguna yang mematuhi (seperti *browser* web) harus berinteraksi dengannya hanya menggunakan koneksi HTTPS yang aman (yaitu HTTP berlapis di atas TLS/SSL). Dapat dilihat pada gambar 4.15.



Gambar 4.15 Hasil Scan *Strict-Transport-Security Header Not Set*

Pada gambar 4.15 di jelaskan bahwa pada *website* SMP Bintang Laut terdapat URL <http://smpkatolikbintanglautternate.sch.id> adalah bukti ada kerentanan pada url tersebut. Dengan tingkat resiko *low* atau dampak serangan dari kerentanan itu lemah. Kemudian *Confidence* ialah kemungkinan tingkat keberhasilan dalam menguji itu medium. Dengan parameter *X-Frame-Options* yang perlu diuji memungkinkan penyerang untuk melakukan serangan.

4.3. Penetration Testing

Berdasarkan hasil scanning menggunakan *owaspzap* pada website berdasarkan *Owasp* 10 untuk memberikan rekomendasi apa yang dibutuhkan nantinya untuk meningkatkan keamanan pada *website* SMPK Bintang Laut Kota.

4.3.1. Sql Injection

Pengujian ini menggunakan *payload* "`sqlmap -r /home/kali/desktop/sql.txt -p kode_hal --random-agent --level 5 --risk 3 --temper=space2comment --dbs`". Dengan menggunakan bantuan *linux* di *virtual box* untuk menginjeksi *payload*. Dapat di lihat pada gambar 4.16.


```

[20:58:56] [INFO] testing 'MySQL UNION query (NULL) - 31 to 40 columns'
[21:03:54] [INFO] testing 'MySQL UNION query (random number) - 31 to 40 columns'
[21:09:17] [INFO] testing 'MySQL UNION query (NULL) - 41 to 50 columns'
[21:14:37] [INFO] testing 'MySQL UNION query (random number) - 41 to 50 columns'
[21:21:12] [WARNING] GET parameter 'page_hal' does not seem to be injectable
[21:21:12] [CRITICAL] all tested parameters do not appear to be injectable

[*] ending @ 21:21:12 /2023-07-14/

```

Gambar 4.18 Tampilan Hasil *Payload*

Pada gambar 4.18 merupakan hasil eksekusi *payload* tersebut ternyata terdapat warning “GET parameter ‘*page_hal*’ does not seem to be injectable” parameter *page_hal* tidak bisa di suntikan, mungkin karena dari *input* beberapa karakter yang telah di batasi serta beberapa karakter khusus juga di blokir sehingga melihat dari hasil pengujian tidak ada kerentanan pada *sql injection*.

4.3.2. Cross Site Scripting

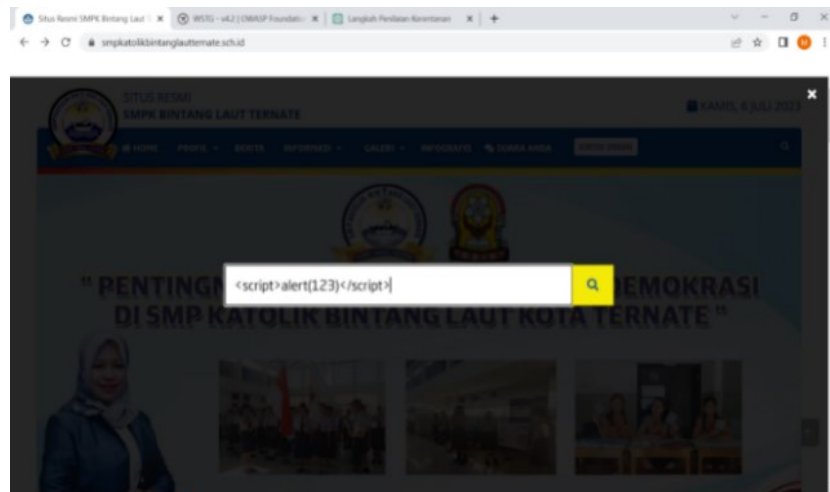
Pada target serangan terdapat variasi yang memungkinkan untuk di exploitasi, berikut uji menggunakan *payload*:

Script XSS :

```
<script>alert(123)</script>
```

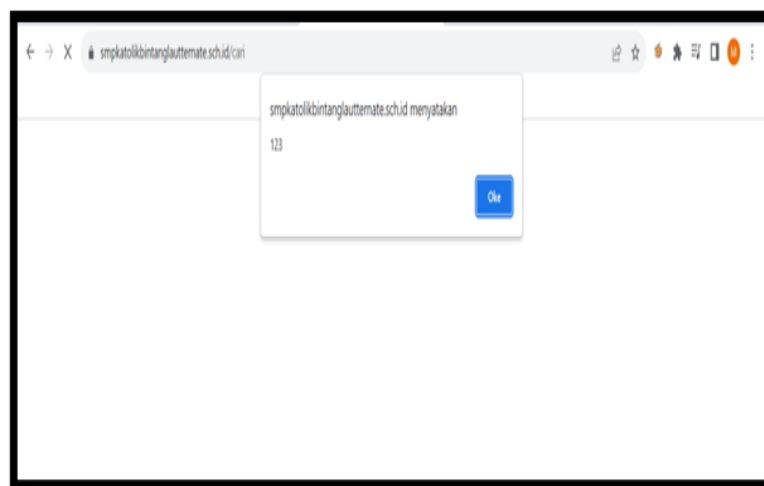
```
<script>alert(document.cookie)</script>
```

Source code pada *file* html atau php berjalan secara runtut. Variasi serangan 1 Misalnya, *user* memasukan inputan `<script>alert(123)</script>` di menu pencarian akan ditampilkan pada posisi pertengahan *file* php, maka *script* tersebut akan dijalankan terlebih dahulu untuk menampilkan *alert* yang berisikan XSS. Untuk lebih jelas dapat dilihat pada gambar 4.19.



Gambar 4.19 Tampilah *Payload* XSS

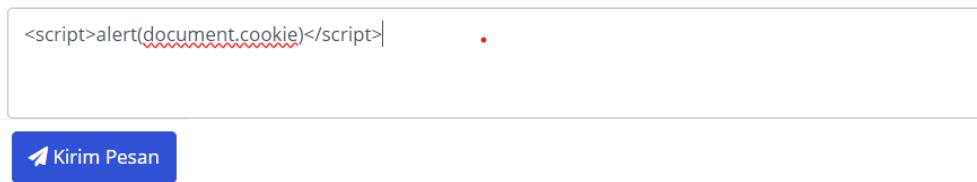
Pada gambar 4.19 adalah penerapan *payload* variasi pertama pada *website* smantigternate.sch.id dengan *payload* `<script>alert(123)</script>`. Hasilnya dapat di lihat pada gambar 4.20.



Gambar 4.20 Tampilan Hasil *Payload* Variasi 1

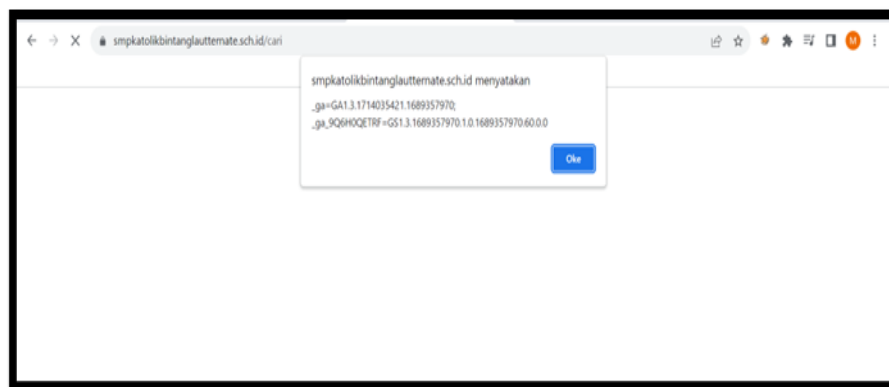
Pada gambar 4.20 adalah hasil *payload* `<script>alert(123)</script>` yang dimana hasil ini akan menampilkan angka 123 sesuai dengan *payload* yang dimasukkan.

Kemudian bisa juga menggunakan variasi serangan dengan memasukkan *inputan* `<script>alert(document.cookie)</script>` pada menu tambah saran. Dapat dilihat pada gambar 4.21.



Gambar 4.21 Tampilan *Payload* XSS

Gambar 4.21 adalah penerapan *payload* dasar pada *website* smantigternate.sch.id dengan *payload* `<script>alert(document.cookie)</script>`. Hasil dari *payload* variasi 2 dapat dilihat pada gambar 4.22.



Gambar 4.22 Tampilan Hasil *Payload* Variasi 2

Pada gambar 4.22 adalah gambar hasil *payload* dengan variasi 2 “`<script>alert(document.cookie)</script>`” untuk menampilkan isi dokumen *cookie*. Dan *cookie* yang didapatkan adalah `_ga=GA1.3.1714035421.1689357970;` dan `_ga_9Q6H0QETRF=GS1.3.1689357970.1.0.1689357970.60.0.0.` jika Hal ini berbahaya apabila dibiarkan akan sangat berbahaya pada sebuah *website*.

4.3.3. ClickJacking

Penguji dapat menyelidiki apakah halaman target dapat dimuat dalam bingkai sebaris dengan membuat halaman web sederhana yang menyertakan bingkai yang berisi halaman web target. kode HTML untuk membuat halaman web pengujian ini ditampilkan dalam skript berikut:

```

<html>

<head>

  <title>Clickjack test page</title>

</head>

<body>

  <h1>selamat website anda telah di hack</h1>

  <iframe src="https://smpkatolikbintanglautternate.sch.id/" width="500"

    height="500"></iframe>

</body>

</html>

```

Jika halaman <https://smpkatolikbintanglautternate.sch.id> berhasil dimuat ke dalam bingkai, maka situs tersebut rentan dan tidak memiliki jenis perlindungan terhadap serangan *clickjacking*. Kerentanan ini mengakibatkan *website* dapat di muat kedalam web lain sehingga akan rentan terhadap *social engineering*. Adapun *payload* yang digunakan `<iframe src="https://smpkatolikbintanglautternate.sch.id/" width="500" height="500"></iframe>` dan berhasil di muat. Untuk lebih jelas dapat dilihat pada gambar 4.23.



Gambar 4.23 Tampilan Implementasi Kode Script

4.3.4. Hidden File Found

File sensitif diidentifikasi sebagai dapat diakses atau tersedia. Ini dapat membocorkan informasi administratif, konfigurasi, atau kredensial yang dapat dimanfaatkan oleh individu jahat untuk menyerang sistem lebih lanjut atau melakukan upaya rekayasa sosial. Salah satu halaman yang rentan adalah <https://smpkatolikbintanglautternate.sch.id/composer.lock>, sehingga memastikan konsistensi dependensi di antara semua pengembang yang bekerja pada proyek tersebut perhatikan gambar 4.24.

```
{
  "_readme": [
    "This file locks the dependencies of your project to a known state",
    "Read more about it at https://getcomposer.org/doc/01-basic-usage.md#installing-dependencies",
    "This file is @generated automatically"
  ],
  "content-hash": "72f566a0f5a908dc74fe33bclabd7905",
  "packages": [
    {
      "name": "kint-php/kint",
      "version": "3.3",
      "source": {
        "type": "git",
        "url": "https://github.com/kint-php/kint.git",
        "reference": "335ac1bcafo4d87df70d8aa51e8887ba2c6d203b"
      }
    }
  ]
}
```

Gambar 4.24 Tampilan Implementasi Link pada Browser

Pada gambar 4.24 adalah *Composer.lock* adalah, manajer dependensi untuk PHP. *File* ini berisi daftar lengkap dari semua dependensi (pustaka atau paket) yang digunakan dalam sebuah proyek PHP bersama dengan versi yang spesifik untuk setiap dependensi. *Composer.lock* mencatat versi spesifik yang digunakan untuk setiap dependensi saat ini.

4.4. Analisis

Berdasarkan hasil scanning dan pentest yang dilakukan didapatkan beberapa celah keamanan yang ada akan di analisa yaitu *Sql Injaction*, *Croos Site Scripting*, *click jacking*

dan *Hidden file found* pada website smp katolik bintang laut berdasarkan Owasp 10, dapat dilihat pada tabel 4.2.

Tabel 4.2 Hasil Analisis

No	Kerentanan	Hasil	Uji	Rekomendasi
1	SQL Injection	SQL Injection adalah jenis serangan siber yang terjadi ketika penyerang menyuntikkan atau memasukkan kode SQL berbahaya ke dalam <i>input</i> yang diterima oleh aplikasi web atau basis data dalam hal ini kerentanan pada parameter <i>page-hal</i> dan memiliki resiko yang sangat tinggi dengan serangan nilai parameter [1" WAITFOR DELAY '0:0:15' --] yang dimana ini menunjukkan bahwa kelemahan sql adalah sql blind. Penetrasi dilakukan menggunakan sqlmap yang dimana hasil penetrasi menunjukkan tidak berhasil mendapatkan <i>database website</i> . Dengan ini <i>website</i> tetap memiliki kerentanan yang sangat tinggi dan harus diperbaiki	Tidak berhasil	Penggunaan parameter <i>ized queries</i>: Menggunakan parameter dalam query SQL untuk memisahkan <i>input</i> pengguna dari kode SQL. Validasi <i>input</i>: Memeriksa dan membersihkan <i>input</i> pengguna sebelum menggunakannya dalam query SQL. Escape characters: Melarang karakter-karakter khusus SQL untuk mencegah kode berbahaya dimasukkan. Penggunaan pengamanan lapisan aplikasi: Menggunakan <i>firewall</i> atau perangkat lunak keamanan untuk mendeteksi dan mencegah serangan SQL Injection. Prinsip <i>least privilege</i>: Memberikan akses terendah yang diperlukan pada basis data untuk mencegah akses yang tidak sah.

2	Cross-Site Scripting (XSS)	<i>Cross-Site Scripting (XSS)</i> adalah serangan keamanan pada aplikasi web yang memungkinkan penyerang menyisipkan skrip berbahaya (biasanya <i>JavaScript</i>) ke dalam halaman web yang dilihat oleh pengguna lain. Pengujian dilakukan menggunakan <i>payload</i> “<script>alert(123)</script>” dan “<script>alert(document.cookie)</script>” dimana saat memasukkan <i>payload</i> xss ternyata muncul <i>alert</i> sesuai dengan <i>payload</i> yg digunakan dan ini sangat berbahaya dalam sebuah <i>website</i>.	Berhasil	<i>Validasi Input:</i> Validasi dan membersihkan <i>input</i> pengguna sebelum memasukkannya ke dalam halaman web. <i>Escaping:</i> Menghindari penggunaan <i>input</i> pengguna dalam konteks HTML, <i>JavaScript</i>, atau URL tanpa karakter khusus (<i>escaping</i>). Penggunaan <i>Header HTTP</i> yang Aman: Mengatur <i>header HTTP</i>, seperti <i>Content Security Policy (CSP)</i>, untuk membatasi sumber skrip yang diizinkan untuk dijalankan di halaman web. Penanganan <i>Cookie</i> yang Aman: Memastikan bahwa <i>cookie</i> sesi tidak dapat diakses oleh skrip <i>JavaScript</i> yang tidak sah. Penetrasi Uji Keamanan: Melakukan pengujian keamanan reguler dan pengujian penetrasi untuk mengidentifikasi dan memperbaiki kerentanan XSS.
3	Clickjacking	<i>Clickjacking</i> adalah serangan siber yang memanipulasi pengguna agar melakukan tindakan yang tidak diinginkan secara tidak sadar pada situs web atau aplikasi web tertentu. Pengujian yang dilakukan yaitu memasukkan <i>header</i> kedalam skript yang sudah	Berhasil	<i>Validasi Input:</i> Validasi dan membersihkan <i>input</i> pengguna sebelum memasukkannya ke dalam halaman web. <i>Escaping:</i> Menghindari penggunaan <i>input</i> pengguna dalam konteks HTML, <i>JavaScript</i>, atau URL tanpa karakter khusus (<i>escaping</i>). Penggunaan <i>Header HTTP</i> yang Aman: Mengatur <i>header HTTP</i>, seperti <i>Content Security Policy (CSP)</i>, untuk membatasi sumber skrip yang diizinkan untuk dijalankan di halaman web.

		dibuatkan untuk untuk memastikan adanya <i>clickjacking</i> pada <i>website</i> tersebut. Berdasarkan hasil <i>scanning</i> dan <i>pentest</i> ternyata skript yang telah dibuat dan hasil <i>scanning</i> menunjukkan adanya <i>clickjacking</i> pada <i>website</i> tersebut dikarenakan <i>x-frame-options header</i> tidak di konfigurasi dengan baik sehingga hal tersebut dalam dilakukan pada <i>header</i>.		Penanganan <i>Cookie</i> yang Aman: Memastikan bahwa <i>cookie</i> sesi tidak dapat diakses oleh skrip <i>JavaScript</i> yang tidak sah.
4	<i>Hidden File Found</i>	<i>Hidden file found</i> adalah pemberitahuan umum yang membantu Anda menyadari keberadaan <i>file</i> tersembunyi di sistem Anda. Dengan berhati-hati dan pemahaman yang tepat, Anda dapat mengelola <i>file-file</i> tersebut sesuai kebutuhan. Dalam <i>website</i> tersebut terdapat kelemahan ini yang memungkinkan informasi tentang <i>website</i> tersebut terbuka dan informasi tersebut		Mengatur <i>Atribut File</i>: Anda dapat mengatur atau menghapus atribut tersembunyi dari <i>file</i> atau <i>folder</i> yang ingin Anda lihat. Pada sistem <i>Windows</i>, Anda dapat melakukan ini melalui properti <i>file</i> atau dengan mengubah pengaturan <i>folder</i> untuk menampilkan <i>file</i> tersembunyi. Di lingkungan <i>Unix/Linux</i>, Anda dapat menggunakan perintah <i>ls -a</i> untuk menampilkan semua <i>file</i>, termasuk yang tersembunyi. Periksa <i>File Tersembunyi</i>: Sebelum mengambil tindakan apa pun pada <i>file</i> tersembunyi, pastikan Anda memahami tujuan dan fungsi <i>file</i> tersebut. Terkadang, <i>file</i> tersembunyi adalah <i>file</i> konfigurasi atau sistem yang penting, dan menghapus atau mengubahnya dapat menyebabkan masalah. <i>Backup Data</i>: Sebelum menghapus

		sangat berbahaya pada sebuah <i>website</i> .		atau mengubah <i>file</i> apa pun, selalu sebaiknya membuat salinan cadangan (<i>backup</i>) terlebih dahulu. Hal ini akan membantu Anda mengembalikan data jika terjadi masalah. Pemeliharaan Berkala: Lakukan pemeliharaan sistem secara berkala untuk memeriksa dan membersihkan <i>file-file</i> yang tidak diinginkan atau yang tidak terlihat.
--	--	---	--	---

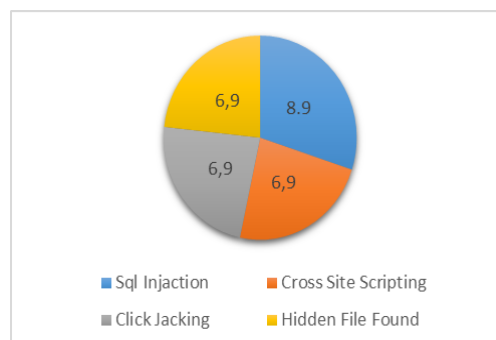
4.5. Hasil

Tahap ini merupakan penilaian kerentanan menggunakan dasar CVSS (*Common Vulnerability Scoring System*), berupa tabel dari sebuah penelitian yang telah dilakukan berdasarkan hasil *scanning* dan *pentest* pada *website* SMPK Bintang Laut Kota Ternate. Dapat dilihat pada tabel 4.3.

Tabel 4.3 *Vulnerability Assessment*

No	Kerentanan	Risk	Uji Kerentanan	Skor
1.	<i>Sql Injection</i>	<i>High</i>	Tidak berhasil	8.9
2.	<i>Cross Site Scripting</i>	<i>High</i>	Berhasil	6.9
3.	<i>Click Jacking</i>	<i>Medium</i>	Berhasil	6.9
4.	<i>Hidden File Found</i>	<i>Medium</i>	Berhasil	6.9

Pada tabel 4.3 merupakan hasil dari *vulnerability assessment* terdapat 1 yang tidak berhasil pada saat di *pentest* dan 3 berhasil Dengan dengan angka *scoring* yang berbeda-beda. Dapat dilihat pada gambar 4.25.



Gambar 4.25 Grafik *Scoring*

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan Pengujian pada *website* *smpkatolikbintanglautternate.sch.id* dengan beberapa Teknik serta menggunakan beberapa *tools*, hasil dari pengujian adalah *Sql Injection*, *Croos Site Scripting*, *click jacking* dan *Hidden file found* ditemukan (3) kerentanan dan tidak ditemukan kerentanan (1) yaitu *Sql injection*.

Serangan *Sql Injection* dilakukan menggunakan *sqlmap* pada terminal di *linux* dengan parameter *page_hal*. Pada serangan *XSS* terdapat 2 variasi serangan dimana, serangan pertama memasukkan *payload* pada kolom pencarian, dan yang kedua memasukkan *payload* pada kolom komentar.

5.2. Saran

Berdasarkan penelitian yang sudah dilakukan terdapat beberapa saran dalam pengembangan *website* sebagai berikut:

1. Pembuatan *website* lebih diperhatikan lagi pada *from* tertentu agar tidak ada celah yang bisa merugikan pemilik *website*.
2. Kepada pengelola *website* penulis memberikan saran agar kerentanan dari 71 *clickjacking* dapat segera di handel dan juga *application disclosure* dan *Vulnerable JS Library* dapat segera di perbaiki *source code* nya dan *jquery* dapat di *update* pada *patch* terbaru untuk *X-Frame-Header-Not Set* karena kasus di atas dapat melakukan *injection script clickjacking* kedalam halaman *login* dimana dalam saran ini pengelolah *website*, harus menggunakan *fortinet-next generator* sebagai *firewall* yang dapat mampu memblokir ancaman *clickjacking* secara *realtime* dan pindahkan element

halaman dan atur agar tidak sesuai dengan settingan bawaan supaya pengelola *website* dapat mengetahui kapan serangan *clickjacking* itu dilakukan.

3. Untuk penelitian berikutnya pada kelemahan sql yang masih lemah sangat dilakukan dengan pengujian yang berbeda
4. Ada beberapa celah yang masi sangat merugikan dan berpotensi dilakukannya serangan. Disarankan dilakukan dengan pengujian yang berbeda.

DAFTAR PUSTAKA

- Budi Dwi Wira, & Ardian Infantono. 2021. "Strategi Penguatan *Cyber Security* Guna Mewujudkan Keamanan Nasional Di Era *Society 5.0*." Prosiding Seminar Nasional Sains Teknologi dan Inovasi Indonesia (SenastindO) 3(November): 223–34.
- Dewi, Muliya et al. 2023. "*Vulnerability Assessment* Pada *Website* Rekrutasi Asisten (IRIS) Fakultas Rekayasa Industri Menggunakan Nikto Dan Nessus." 10(2): 1631–36.
- Kusuma, Gregorius. 2022. "Implementasi *Owasp Zap* Untuk Pengujian Keamanan Sistem Informasi Akademik." Jurnal Teknologi Informasi: Jurnal Keilmuan dan Aplikasi Bidang Teknik Informatika 16(2): 178–86.
- Masykur, Fauzan, et al. Analisis *Vulnerability Web Based Application* Menggunakan Nessus no. November, 2015, pp. 320–26.
- Mulyanto. 2021. "Sumbawa Menggunakan Metode *Vulnerability Aseement*." Jinteks 3(3): 394–400. <https://smanika-sumbawabesar.sch.id>.
- Nabila, Putri. 2023. "Uji *Vulnerability Assessment* Dalam Mengetahui Tingkat Keamanan Web Aplikasi Sistem Informasi Laporan Diskominfo dan Sandi Aceh." *Jintech: Journal Of Information Technology* 4(1): 1–15.
- Prasetya, Fajar. 2023. "Analisis Keamanan Situs Web Perpustakaan SMAN 3 Tambun Selatan Menggunakan Metode *Vulnerability Assessment*." Jurnal Sains dan Informatika 9(September 2022): 67–76.
- Purmanta Siagian, Harry. 2017. " *Vulnerability Aseement* pada *web server*." 223-32.
- Rendro, Bayu. 2020. "Analisis Monitoring Sistem Keamanan Jaringan Komputer Menggunakan *Software Nmap* (Studi Kasus Di Smk Negeri 1 Kota Serang)." Prosisko: Jurnal Pengembangan Riset dan Observasi Sistem Komputer 7(2): 108–15. <https://e-jurnal.lppmunsera.org/index.php/Prosisko/article/view/2522>.
- Riadi, Imam, Anton Yudhana, & Yunanri W. 2020. "Analisis Keamanan *Website Open Journal System* Menggunakan Metode *Vulnerability Assessment*." Jurnal Teknologi Informasi dan Ilmu Komputer 7(4): 853. doi:10.25126/jtiik.2020701928.
- Rustianto, April. 2010. Analisis *Vulnerability* pada *Website*.
- Wahyudi, Dimas. 2020. "*Reconnaissance* - Dimas Wahyudi." (1): 1–7.
- Yelvita, Feby Sri. 2022. Jurnal Teknologi Informasi dan Ilmu Komputer Pengujian dan

Analisis Keamanan *Website* Institut Teknologi Padang Menggunakan *Acunetix Vulnerability Scanner*. Jurnal Teknologi Informasi dan Ilmu Komputer.



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 24 NOVEMBER 2023

Pukul : 15:30 - 17:30

Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : MAHANI ALBAAR

NPM : 07351711012

Judul : ANALISIS VULNERABILITY PADA WEBSITE DI KOTA TERNATE
MENGUNAKAN METODE VULNERABILITY ASSESMENT (STUDI
KASUS: DUKCAPIL KOTA TERNATE)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

Revisi catatan dari para penguji

Kuasai metode yang digunakan

Lakukan vulnerability assesment lebih mendalam lagi dan lakukan analisis

6/5/2024

ACC Revisi

[Signature]

Dosen Pembimbing I,

[Signature]

Ir. HAIKIN UTFI, S.Kom., M.T., IPM
NIP. 198601112014041002



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 24 NOVEMBER 2023
Pukul : 15:30 - 17:30
Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : MAHANI ALBAAR
NPM : 07351711012
Judul : ANALISIS VULNERABILITY PADA WEBSITE DI KOTA TERNATE
MENGUNAKAN METODE VULNERABILITY ASSESMENT (STUDI
KASUS: DUKCAPIL KOTA TERNATE)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

① Lakukan perbaikan segera, sesuai yg di minta
oleh para pengisi :

Apa perbaikan hasil seminar hasil
sesuai yang telah ditetapkan

Dosen Pembimbing II,

Ir. AMAL KHAIRUN, S.T., M.Eng., IPM
NIP. 197401112003121003



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 24 NOVEMBER 2023
Pukul : 15.30 - 17.30
Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : MAHANI ALBAAR
NPM : 07351711012
Judul : ANALISIS VULNERABILITY PADA WEBSITE DI KOTA TERNATE
MENGUNAKAN METODE VULNERABILITY ASSESMENT (STUDI
KASUS: DUKCAPIL KOTA TERNATE)

Dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- ① Hasil Analisis tidak dapat dibuktikan karena belum ada bukti yang mendukung yg kpn dilakukan juga
- ② Ada banyak kalimat yg susah di mengerti karena kalimatnya di copy paste dan tools yg digunakan.
- ③ Objek Penelitian sebaiknya diganti. Bk masu belu aktif.

12/2/24
Aet

Abdul Mubarak

Dosen Penguji I,

Ir. ABDUL MUBARAK, S.Kom., M.T., IPM
NIP. 198212062014041002



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 24 NOVEMBER 2023

Pukul : 15:30 - 17:30

Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : MAHANI ALBAAR

NPM : 07351711012

Judul : ANALISIS VULNERABILITY PADA WEBSITE DI KOTA TERNATE
MENGUNAKAN METODE VULNERABILITY ASSESSMENT (STUDI
KASUS: DUKCAPIL KOTA TERNATE)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- Pahami langkah-langkah melakukan pentest

Ace 7-2

Dosen Penguji II,

MUHAMMAD FHADI, S.Kom., M.Sc.
NIP. 199611232023211012



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 24 NOVEMBER 2023
Pukul : 15:30 - 17:30
Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : MAHANI ALBAAR
NPM : 07351711012
Judul : ANALISIS VULNERABILITY PADA WEBSITE DI KOTA TERNATE
MENGUNAKAN METODE VULNERABILITY ASSESSMENT (STUDI
KASUS: DUKCAPIL KOTA TERNATE)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

1. latar belakang diperbaiki ✓

2. pelajari dan diskusi skripsi terkait punya Fajrul ✓

3. kriteria/data2 seperti apa? perlu lebih jelas ✓

4. format penulisan ✓

5. kesimpulan diperbaiki lagi ✓

6. teliti kembali implementasi metode VA ✓

Dosen Penguji III,

ACHMAD FUAD, S.T., M.T.
NIP. 197606182005011001

2, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 07 JUNI 2024

Pukul : 09:00 - 10:30

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : MAHANI ALBAAR

NPM : 07351711012

Judul : ANALISIS VULNERABILITY PADA WEBSITE SMPK BINTANG
LAUT KOTA TERNATE MENGGUNAKAN METODE
VULNERABILITY ASESSMENT (BERDASARKAN PANDUAN
OWASP 10)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

Revisi catatan dari para penguji

Kuasai pengetahuan dasar informatika

Kuasai Office

9/7/2024

Ace Revisi

Dosen Pembimbing I,

Ir. SALKIN LUTFI, S.Kom., M.T.
NIP. 198601112014041002



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 07 JUNI 2024

Pukul : 09:00 - 10:30

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : MAHANI ALBAAR

NPM : 07351711012

Judul : ANALISIS VULNERABILITY PADA WEBSITE SMPK BINTANG
LAUT KOTA TERNATE MENGGUNAKAN METODE
VULNERABILITY ASESSMENT (BERDASARKAN PANDUAN
OWASP 10)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- Segera lakukan perbaikan masalah skripsi sesuai
koreksi dan saran dari para penguji -

Agg perbaikan skripsi pada 31 07 2024
Segera kirim ulang jurnal

Dosen Pembimbing II,


Ir. AMAL KHAIKUN, S.T., M.Eng., IPM
NIP. 197401112003121003



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 07 JUNI 2024

Pukul : 09:00 - 10:30

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : MAHANI ALBAAR

NPM : 07351711012

Judul : ANALISIS VULNERABILITY PADA WEBSITE SMPK BINTANG
LAUT KOTA TERNATE MENGGUNAKAN METODE
VULNERABILITY ASESSMENT (BERDASARKAN PANDUAN
OWASP 10)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- ①. Rekomendasi keamanan buat dalam bentuk tabel.
- ②. Pelajari kembali tools yang digunakan sampai pada menghasilkan analisis
- ③. Pelajari kembali metode yang digunakan

Ata P...
17/7/2024
ABDUL MUBARAK

Dosen Penguji I.

Ir. ABDUL MUBARAK, S.Kom., M.T., IPM
NIP. 198212062014041002



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 07 JUNI 2024

Pukul : 09:00 - 10:30

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : MAHANI ALBAAR

NPM : 07351711012

Judul : ANALISIS VULNERABILITY PADA WEBSITE SMPK BINTANG
LAUT KOTA TERNATE MENGGUNAKAN METODE
VULNERABILITY ASESSMENT (BERDASARKAN PANDUAN
OWASP 10)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- Pahami alur penelitian

Acc
g. Tul
10/6/24

Dosen Penguji II,

MUHAMMAD FHADLI, S.Kom., M.Sc.
NIP. 199611232023211012



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 07 JUNI 2024

Pukul : 09:00 - 10:30

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : MAHANI ALBAAR

NPM : 07351711012

Judul : ANALISIS VULNERABILITY PADA WEBSITE SMPK BINTANG
LAUT KOTA TERNATE MENGGUNAKAN METODE
VULNERABILITY ASESSMENT (BERDASARKAN PANDUAN
OWASP 10)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

1. revisi kesalahan penulisan/format penulisan dan lengkapi pustaka dengan referensi yang tepat ✓
2. kesimpulan ditambahkan lebih spesifik, abstrak dan analisis akhir ✓
3. tahapan2 serangan setiap model dan hasilnya dilengkapi dan berikan penjelasan yang tepat! ✓
4. jenis2 serangan diperjelas termasuk variasi serangan ✓
5. tabelkan hasil2 yang dicapai dan buat diagramnya ✓

Dosen Penguji III,

ACHMAD FUAD, S.T., M.T.
NIP. 197606182005011001



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI
UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK PROGRAM STUDI TEKNIK INFORMATIKA
Kampus III Universitas Khairun, Kelurahan Jati Kota Ternate Selatan
<http://if.unkhair.ac.id>, <http://unkhair.ac.id> Group FB: if.unkhair

KARTU BIMBINGAN HASIL

Mahasiswa
Pembimbing
Laporan KP

: Mahani Albaar (07351711012)
: Salkin Lutfi, S.Kom., M.Kom
: Analisis Vulnerability Pada Website di Kota Ternate Menggunakan
Metode Vulnerability Assessment untuk mengetahui kelemahan dari
website (studi Kasus: SMPK Bintang Laut Kota Ternate)

NO	Tanggal	Uraian	Paraf
	05/10/2023	* Tambahkan hasil * Lanjut kesimpulan	
	10/08/2023	* Ganti Studi kasus * Format Penulisan	
	14/08/2023	* Analisis diperjelas * Gambar diperjelas	
	16/08/2023	* Format Penulisan	
	11/09/2023	* Format Penulisan	
		15/9/2023 Acc B 	



KARTU BIMBINGAN HASIL

Nama Mahasiswa
Dosen Pembimbing
Judul Laporan Hasil

: Mahani Albaar (07351711012)
: Ir, Amal Khairan, S.T., M.Eng.
: Analisis Vulnerability Pada Website di Kota Ternate Menggunakan
Metode Vulnerability Assessment untuk mengetahui kelemahan dari
website (studi Kasus: SMPK Bintang Laut Kota Ternate)

NO	Tanggal	Uraian	Paraf
1	16 10/23	- Penulisan ide mengenai studi pembacaan penulisan Skripsi - Caranya sudah dan terbukti (Sangat baik dan sebagai bimbingan)	
		- Sistematisasi Pembacaan tugas - Jadwal & bimbingan - Pembacaan pengantar pd	
		Kerangka awal - Kata "Anda" & bimbingan di naskah. Parafikilah kerangka.	
		Dengan Copy Paste di web - Teori & metode yg dipakai (Vulnerability Assessment)	
		Kerangka. Detilkan kerangka pd Bab 2	
		- dan juga lain yg sdg sdg tandi dan komentar di naskah.	



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI
UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK PROGRAM STUDI TEKNIK INFORMATIKA
Kampus III Universitas Khairun, Kelurahan Jati Kota Ternate Selatan
<http://if.unkhair.ac.id>, <http://unkhair.ac.id> Group FB: if.unkhair

2-	14 11 2023	- Rumusan Masalah di perbaiki (pilih 1 & 2 saja)	
		- Tujuan Penelitian di ganti (pilih 1 & 2 saja tulis 5 masalah)	
3	16 11 2023	Ace di Separat. perbaiki kembali kalimat yang ada di Rumusan Masalah dan tujuan penelitian separatenya di 1 & 2 saja saranan.	
		Ace hasil 16 2023	