SIMULASI *PENETRATION TESTING CENTER OF E-LEARNING AND EDUCATION FOR STUDENTS (CERDAS)* UNIVERSITAS ISLAM RIAU DENGAN METODE *BRUTE FORCE* MENGGUNAKAN *HATCH*

SKRIPSI

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Teknik Pada Fakultas Teknik
Universitas Islam Riau

R. MERLANG
173510178

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS ISLAM RIAU
PEKANBARU
2022

LEMBAR PENGESAHAN PEMBIMBING SKRIPSI

Nama : R. Merlang
NPM : 173510178
Jurusan : Teknik
Program Studi : Teknik Informatika
Jenjang Pendidikan : Strata Satu (S1)
Judul Skripsi : Simulasi Penetration Testing *Center Of E-Learning And Education For Students (CERDAS)* Universitas Islam Riau Dengan Metode *Brute Force* Menggunakan *Hatch*

Format sistematika dan pembahasan materi pada masing-masing bab dan sub bab dalam skripsi ini telah dipelajari dan dinilai relatif telah memenuhi ketentuan-ketentuan dan kriteria-kriteria dalam metode penulisan ilmiah. Oleh karena itu, skripsi ini dinilai layak serta dapat disetujui untuk disidangkan dalam ujian komprehensif

Pekanbaru, 10 Agustus 2022

Disahkan Oleh :	**Disetujui Oleh :**
Dosen Pembimbing	Ketua Prodi Teknik Informatika
(Apri Siswanto, S.Kom., M.Kom)	**(Apri Siswanto, S.Kom., M.Kom)**

LEMBAR PENGESAHAN TIM PENGUJI UJIAN SKRIPSI

Nama	:	R. Merlang
NPM	:	173510178
Fakultas	:	Teknik
Program Studi	:	Teknik Informatika
Jenjang Pendidikan	:	Strata Satu(S1)
Judul Skripsi	:	Simulasi Penetratition Testing Center of E-Learning and Education For Students (CERDAS) Universitas Islam Riau Dengan Metode Brute Force Menggunakan Hatch

Skripsi ini secara keseluruhan dinilai telah memenuhi ketentuan-ketentuan dan kaidah-kaidah dalam penulisan penelitian ilmiah serta telah diuji dan dapat dipertahankan dihadapan tim penguji. Oleh karena itu, Tim Penguji Ujian Skripsi Fakultas Teknik Universitas Islam Riau menyatakan bahwa mahasiswa yang bersangkutan dinyatakan **Telah Lulus Mengikuti Ujian Komprehensif Pada Tanggal 18 Agustus 2022** dan disetujui serta diterima untuk memenuhi salah satu syarat guna memperoleh gelar Sarjana Strata Satu
Bidang Ilmu **Teknik Informatika.**

Pekanbaru, 18 Agustus 2022

Tim Penguji

1. **Yudhi Artha, S.T., M.Kom** — **Sebagai Tim Penguji I** ( )
2. **Dr. Evizal, S.T., M.Eng** — **Sebagai Tim Penguji II** ( )

Disahkan Oleh

Ketua Prodi Teknik Informatika	**Dosen Pembimbing**
Dr. Apri Siswanto., S.Kom., M.Kom	**Dr. Apri Siswanto., S.Kom., M.Kom**

LEMBAR PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan dibawah ini:

Nama	: R. Merlang
Tempat/Tgl Lahir	: Rengat, 15 September 1999
Alamat	: Jl. Inspektur Koesen No.07, RT/RW : 014/005, Kec. Rengat, Kab. Indragiri Hulu

Adalah mahasiswa Universitas Islam Riau yang terdaftar pada:

Fakultas	: Teknik
Jurusan	: Teknik Informatika
Program Studi	: Teknik Informatika
Jenjang Pendidikan	: Strata-1 (S1)

Dengan ini menyatakan dengan sesungguhnya bahwa skripsi yang saya tulis adalah benar dan asli hasil dari penelitian yang telah saya lakukan dengan judul "**SIMULASI PENETRATION TESTING CENTER OF E-LEARNING AND EDUCATION FOR STUDENTS (CERDAS) UNIVERSITAS ISLAM RIAU DENGAN METODE BRUTE FORCE MENGGUNAKAN HATCH".** Apabila dikemudian hari ada yang merasa dirugikan dan atau menuntut karena penelitian ini menggunakan hasil tulisan atau karya orang lain tanpa mencantumkan nama penulis yang bersangkutan, atau terbukti karya ilmiah ini **bukan** karya saya sendiri atau **plagiat** hasil karya orang lain, maka saya bersedia menerima sanksi sesuai dengan peraturan perundangan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sesungguhnya untuk dapat digunakan sebagaimana mestinya.

Pekanbaru, 25 Agustus 2022

Yang membuat pernyataan,

R. Merlang

LEMBAR IDENTITAS PENULIS

Nama	: R. Merlang
NPM	: 173510178
Tempat/TanggalLahir	: Rengat, 15 September 1999
Alamat Orang Tua	: Jl. Inspektur Koesen No. 07, RT/RW : 014/005, Kec. Rengat, Kab. Indragiri Hulu
Nama Orang Tua	
Nama Ayah	: R. Zulhendra
Nama Ibu	: Yuliana Sophia L.
No.HP/Telp	: 081365010995
Fakultas	: Teknik
Program Studi	: Teknik Informatika
Masuk Th.Ajaran	: 2017
Keluar Th. Ajaran	: 2022
Judul Penelitian	: Simulasi Penetration Testing Center of E-Learning and Education For Students (CERDAS) Universitas Islam Riau Dengan Metode Brute Force Menggunakan Hatch

Pekanbaru, 25 Agustus 2022

R. Merlang

KATA PENGANTAR

Segala puji bagi Allah SWT yang telah memberikan kemudahan dan kekuatan sehingga penulis dapat menyelesaikan laporan penelitian skripsi dengan judul "Simulasi Penetration Testing *Center Of E-Learning and Education For Students (CERDAS)* Universitas Islam Riau Dengan Metode *Brute Force* Menggunakan *Hatch* " sebagai salah satu syarat wajib untuk menyelesaikan Program Sarjana pada Fakultas Teknik Program Studi Teknik informatika Universitas Islam Riau.

Penulis mengucapkan terima kasih kepada dosen-dosen program studi Teknik informatika yang telah memberikan dukungan berupa ilmu dan arahan sehingga laporan skripsi ini dapat terselesaikan. Kepada orang tua penulis yang selalu ada untuk memberikan dukungan dan kepada teman-teman seperjuangan yang membantu dalam pembuatan laporan ini.

Penulis menyadari bahwa laporan masih terdapat kekurangan di dalamnya. Untuk itu, penulis mengharapkan kritik dan saran yang bersifat membangun. Penulis juga meminta maaf atas kekurangan yang terdapat dalam laporan ini, semoga laporan ini dapat memberikan manfaat kepada yang membaca.

Pekanbaru 01 Agustus 2022

R. Merlang

SIMULASI *PENETRATION TESTING CENTER OF E-LEARNING AND EDUCATION FOR STUDENTS (CERDAS)* UNIVERSITAS ISLAM RIAU DENGAN METODE *BRUTE FORCE* MENGGUNAKAN *HATCH*

R. Merlang
Program Studi Teknik Informatika
Universitas Islam Riau
Email: r.merlang1999@gmail.com

ABSTRAK

Keamanan hak akses merupakan aspek penting agar, tidak terjadi penyalahgunaan oleh oknum yang mencari keuntungan. Pusat operasi keamanan siber nasional dan badan siber sandi negara mencatat terdapat 88 juta serangan siber pada tahun 2020. Untuk upaya pencegahan serangan terhadap CERDAS maka penulis melakukan penelitian dengan mengangkat masalah "Apakah sistem keamanan yang digunakan CERDAS dapat mencegah serangan *brute force* dan bagaimana proses penyerangan *brute force* dilakukan". Adapun tujuan penelitian ini adalah untuk menguji sistem keamanan halaman masuk yang digunakan oleh CERDAS. Metode yang digunakan yaitu NIST 800-115. Hasil penelitian menunjukan sistem keamanan halaman masuk yang digunakan oleh CERDAS memiliki celah terhadap serangan *brute force* dengan tingkat keberhasilan pelaksanaan penyerangan mencapai hingga 100%. Berdasarkan hasil penelitian, CERDAS harus meningkatkan sistem keamanan halaman masuk sebagai upaya menghindari serangan *brute force* yang bertujuan untuk merugikan sistem.

Kata Kunci : *Penetration testing, brute force,* keamanan akses

SIMULATION OF PENETRATION TESTING CENTER OF E-LEARNING AND EDUCATION FOR STUDENTS (CERDAS) ISLAMIC UNIVERSITY OF RIAU WITH BRUTE FORCE METHOD USING HATCH

R. Merlang
Informatic Engineering Program
Islamic University of Riau
Email: r.merlang1999@gmail.com

ABSTRACT

Security of access rights is an important aspect so, that it is not abused by people who seek profit. The National Cyber Security Operations Center and the State Cyber Security Agency recorded 88 million cyber attacks in 2020. In order to prevent attacks against CERDAS, the author conducted a study by raising the issue of "Can the security system used by CERDAS be able to prevent brute force attacks and how is the process of brute force attacks being carried out conducted". The purpose of this study is to test the login page security system used by CERDAS. The method used is NIST 800-115. The results of the study show that the page security system used by CERDAS has a gap against brute force attacks with a success rate of attack implementation reaching 100%. Based on the research results, CERDAS should improve the page security system in an effort to avoid brute force attacks that aim to harm the system.

Keywords : Penetration testing, brute force, security of access

DAFTAR ISI

KATA PENGANTAR ..... **i**
ABSTRAK ..... **ii**
ABSTRACT ..... **iii**
DAFTAR ISI ..... **iv**
DAFTAR TABEL ..... **vi**
DAFTAR GAMBAR ..... **vii**
BAB I PENDAHULUAN **1**
1.1 Latar Belakang 1
1.3 Batasan Masalah 2
1.4 Rumusan Masalah 3
1.5 Tujuan Penelitian 3
1.6 Manfaat Penelitian 3
BAB II LANDASAN TEORI **4**
2.1 Tinjauan Pustaka 4
2.2 Dasar Teori 7
2.2.1. Simulasi 7
2.2.2. PHP 8
2.2.3. Mysql 8
2.2.4. XAMPP 8
2.2.6. Penetration Testing 10
2.2.7. CERDAS 11
2.2.8. *Brute Force* 11
2.2.9. Hatch 12
BAB III METODOLOGI PENELITIAN **13**
3.1 Metode Penelitian 13
3.1.1 NIST SP 800 - 115 (National Institute of Standard and Technology) 13
3.2 Alat Kebutuhan Penelitian 14
3.3 Alur Penelitian 15
3.4 Kebutuhan Aplikasi 17
3.4.1 Hatch 17
3.4.2 Model CERDAS 17

BAB IV HASIL DAN PEMBAHASAN 20
4.1. Planning 20
4.1.1 Pemasangan Sistem Operasi Kali Linux 20
4.1.2 Pemasangan *Tool* Hatch Pada Kali Linux 21
4.2. Discovery 22
4.2.1. Pembuatan Model CERDAS 22
4.3 *Attack* 25
4.3.1 Pembuatan Wordlist 25
4.3.2 Daftar *Username* Yang Akan Diserang 28
4.3.3 Pengecekan Target 30
4.3.4 Mengisi Data Target 30
4.3.5 Proses Penyerangan 31
4.4 *Reporting* 32
4.4.1 Penyerangan terhadap *username* yang belum mengganti *password* ataupun masih menggunakan *password default* 32
4.4.2 Penyerangan terhadap *username* yang telah mengganti *password.* 32
BAB V KESIMPULAN DAN SARAN 34
5.1. Kesimpulan 34
5.2. Saran 35
DAFTAR PUSTAKA 36

DAFTAR TABEL

Tabel 2. 1 Daftar Penelitian Terkait 4
Tabel 2. 2 Simbol Flowchart 9
Tabel 3. 1 Spesifikasi Perangkat Yang Digunakan 15
Tabel 3. 2 Tabel Pengguna 18
Tabel 4. 1 Spesifikasi Kali Linux 20
Tabel 4. 2 Hasil Observasi 22
Tabel 4. 3 Kesimpulan Pengujian 24
Tabel 4. 4 Daftar Penyerangan 29
Tabel 4. 5 Hasil Penyerangan Kelompok A 32
Tabel 4. 6 Hasil Penyerangan Kelompok B 33

DAFTAR GAMBAR

Gambar 3. 1 Penetration Testing Methodology NIST 13
Gambar 3. 2 Alur Tahapan Penelitian 16
Gambar 3. 3 Alur Cara Kerja Hatch 17
Gambar 3. 4 Desain Tatap Muka Halaman Masuk Model CERDAS 18
Gambar 3. 5 Flowchart Model Halaman Masuk Model CERDAS 19
Gambar 4. 1 Tampilan Desktop Kali Linux 21
Gambar 4. 2 Direktori Hatch 21
Gambar 4. 3. Tampilan Halaman Masuk Tiruan CERDAS 23
Gambar 4. 4 Catatan Please Fill Out This Form 23
Gambar 4. 5 Catatan Gagal Login Username atau Password Salah 24
Gambar 4. 6 Skema Penyerangan 25
Gambar 4. 7 Kumpulan Wordlist Tanggal Lahir 26
Gambar 4. 8 Kumpulan Wordlist Umum 27
Gambar 4. 9 Total Wordlist Hatch 28
Gambar 4. 10 Pengecekan Alamat Target 30
Gambar 4. 11 Penginputan Data Hatch 31
Gambar 4. 12 Gambar Hasil Penyerangan Hatch 31

BAB I
PENDAHULUAN

1.1 Latar Belakang

Keamanan informasi adalah aspek penting untuk mencegah penyalahgunaan hak akses, informasi dan kemungkinan kerugian lain yang mungkin terjadi dari suatu sistem. Salah satu upaya yang dapat dilakukan untuk menjaga keamanan adalah dengan cara melakukan evaluasi terhadap sistem yang digunakan, Bentuk evaluasi yang dapat dilakukan adalah tes pengujian ketahanan sistem (*Penetration Testing).*

Pusat Operasi Keamanan Siber Nasional (Pusopskamsinas) dan Badan Siber dan Sandi Negara (BSSN) mencatat terdapat 88 juta serangan siber yang terjadi pada bulan Januari hingga April 2020 dengan 25 juta serangan pada bulan Januari, 29 juta pada bulan Febuari, 26 juta pada bulan Maret dan 7.5 juta pada tanggal 1 – 12 di bulan April. Rincian serangan yang terjadi adalah 56% *trojan activity, information gathering* 43% dan 1 % *web application attack.*

Center of E-learning and Education For Students (CERDAS) Universitas Islam Riau adalah sistem yang dibangun untuk keperluan belajar mahasiswa Universitas Islam Riau. Didalam CERDAS terdapat beberapa informasi dari mahasiswa yang bersifat pribadi dan harus dijaga keamanannya. Oleh karena itu perlu dilakukan evaluasi terutama pada halaman masuk kedalam sistem yang merupakan langkah awal dari keamanan sistem tersebut.

Untuk meminimalisir serangan pada CERDAS maka perlu dilakukan Penetration Testing dengan tujuan untuk evaluasi keamanan. Oleh karena itu

penulis melakukan penelitian skripsi dengan judul " Simulasi Penetration Testing *Center of E-Learning and Education for Students (CERDAS)* Universitas Islam Riau dengan Metode *Brute Force* menggunakan *Hatch*"

1.2 Identifikasi Masalah

Menurut penulis adapun beberapa hal dari sistem keamanan halaman masuk CERDAS yang berpotensi menimbulkan permasalahan yaitu :

1. Tidak membatasi jumlah percobaan masuk ke dalam sistem yang dapat menyebabkan serangan dengan percobaan masuk secara berulang.
2. Tidak menyediakan pengecekan manusia atau robot yang mengakses sistem.
3. Tidak mewajibkan pengguna mengganti *password default* yang diberikan.

1.3 Batasan Masalah

Adapun batasan masalah yang penulis tetapkan agar penelitian ini tetap pada jalurnya saat proses penelitian berjalan antara lain sebagai berikut :

1. Penulis mengambil acuan CERDAS sampai pembaharuan terakhir yaitu pada tanggal 12 Oktober 2021.
2. Proses penetrasi dilakukan dalam bentuk simulasi yaitu penulis membuat sebuah sistem tiruan berdasarkan cara kerja CERDAS.
3. Simulasi CERDAS yang dibangun menggunakan *web server* lokal.

4. Pengujian berfokus pada keamanan halaman masuk CERDAS terhadap serangan *brute force*.

1.4 Rumusan Masalah

Berdasarkan latar belakang masalah yang sebagaimana tertulis diatas penulis merumuskan masalah dalam penelitian ini sebagai berikut :

1. Apakah sistem keamanan halaman masuk yang digunakan oleh CERDAS dapat mencegah serangan *brute force* ?.
2. Bagaimana proses simulasi penetration testing serangan *Brute Force* menggunakan *Hatch* ?.

1.5 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk menguji keamanan halaman keluar masuk CERDAS dalam bentuk simulasi terhadap serangan *brute force* terutama dengan serangan menggunakan Hatch.

1.6 Manfaat Penelitian

Manfaat dari penelitian ini adalah untuk memberikan gambaran tentang keamanan CERDAS terhadap serangan *brute force* sebagai bentuk evaluasi keamanan.

BAB II
LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam melakukan penelitian ini penulis juga melakukan kajian studi kepustakaan dengan dasar yang merujuk pada penelitian-penelitian yang berkaitan sebagai bahan perbandingan dan referensi dalam melakukan penelitian ini yang dapat dilihat pada tabel 2.1.

Tabel 2. 1 Daftar Penelitian Terkait

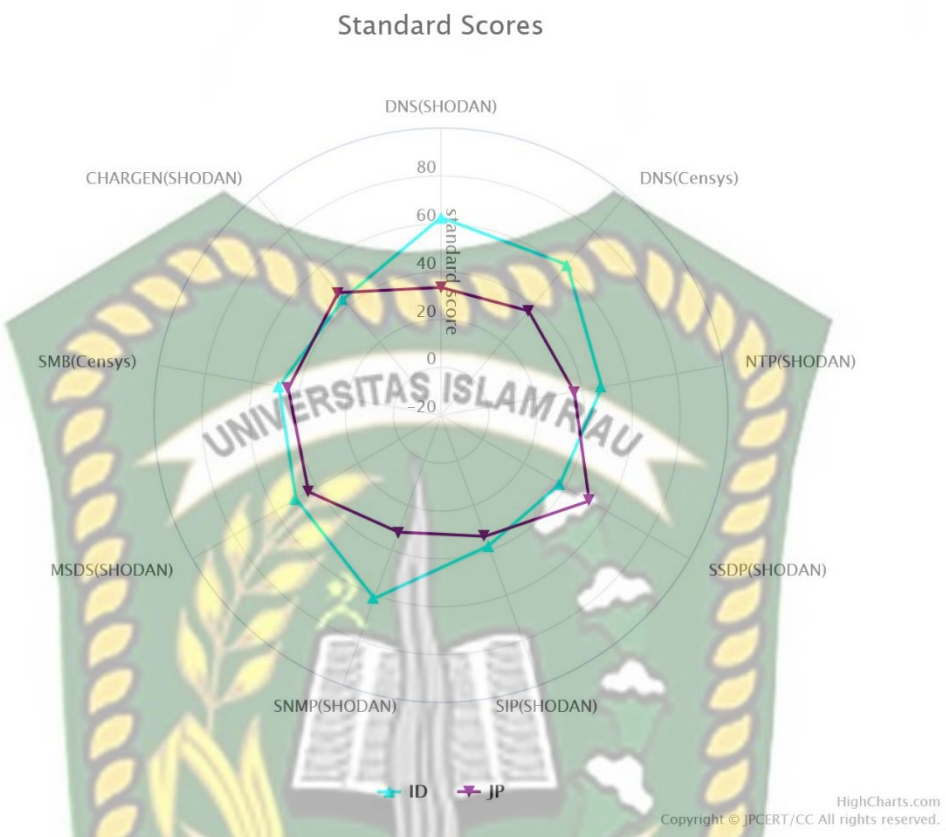
No	Peneliti	Judul Penelitian	Outer Penelitian	Tema
1	Hanif Sidiq Pratita dan Irwan Sembiring	Analisa *Brute Force Attack* Menggunakan *Scanning* Aplikasi pada *HTTP Attack (2016)*	Simulasi serangan *brute force* , menggunakan *Operating System* dan *tools* dari kali Linux, melakukan *Scan* menggunakan WPscan untuk mencari celah keamanan kemudian lakukan penyerangan.	*Penetration testing*, serangan *brute force*
2	Adi Adrian dan Angga Setiyadi	Analisis Keamanan Jaringan Dengan Metode *Penetration Testing Execution Standard* (PTES) di Dinas Kesehatan Provinsi Jawa Barat 2018	Merubah alamat IP dari penyerang , pengujian keamanan jaringan dengan serangan *brute force* menggunakan *air crack*	*Penetration Testing* dengan metode PTES.

No	Peneliti	Judul Penelitian	Outer Penelitian	Tema
3	Dimas Adhi Pratama, Deni Erlansyah dan Febriyanti panjaitan	Penerapan Algoritma *Brute Force* Pada Sistem Informasi Akademik Bina Darma (2019)	Implementasi algoritma *brute force* secara langsung kepada sistem menggunakan algoritma *brute force* menggunakan bahasa C#,	*Penetration testing,* serangan *brute force*
4	Muhammad Subagja Sastra Wardaya	*Penetration Testing* Terhadap *Website* Asosiasi Pekerja Professional Informasi Sekolah Indonesia (APISI)((2019).	*Scanning* untuk mencari celah keamanan pada sistem, setiap celah yang ditemukan akan diuji	*Penetration Testing*
5	Adetya Putra Dewamto	Penetration Testing pada Domain uii.ac.id menggunakan OWASP 10 (2018)	Mencari informasi target menggunakan WPscan, The Harvester, NMAP, Mass Scan dan OWASPZap, menguji setiap celah keamanan berdasarkan informasi yang didapatkan	*Penetration Testing* menggunakan OWASP
6	Setyo Utoro, Bayu Andi Nugroho, Meinawati, Septhian Rheno Widianto	Analisis Keamanan Website *E-Learning* SMKN 1 CIBATU Menggunakan Metode *Penetration Testing Execution Standard (2020)*	Melakukan *scan* untuk mencari kelemahan, melakukan uji keamanan terhadap *Sniffing, Cross Site Scripting, Cross Site Request Forgery* (CSRF)	*Penetration Testing* dengan metode PTES

No	Peneliti	Judul Penelitian	Outer Penelitian	Tema
7	Alde Alanda, Deni Satria, M. Isthofa Ardhana, Andi Ahmad Dahlan dan Hanriyawan Adnan Mooduto	Web Application Penetration Testing Using Sql Injection Attack (2021)	Menggunakan Metode Penetration Testing Exectution Standard (PTES), Melakukan *Port Scanning* untuk menemukan informasi, Melakukan Serangan Brute Force dan SQL Injection untuk mendapatkan informasi terkait pengguna	Penetration Testing Menggunakan Metode PTES, Serangan Brute Force dan SQL Injection
8	Rifki Azis dan Setiadi Yazid	Pengujian Kerentanan Website Wordpress Dengan Menggunakan Penetration Testing Untuk Menghasilkan Website Yang Aman (2021)	Menggunakan Wp-Scan untuk mencari celah keamanan. Melakukan Serangan Brute Force untuk mencari password masuk kedalam sistem dan melakukan exksploitasi databse	Penetration Testing Serangan Brute Force, Ekploitasi Database

2.1.1. Mejiro Analysis

Mejiro adalah layanan visualisasi untuk melihat tingkat resiko internet dari suatu wilayah negara. Data analisis Mejiro didapat dengan cara menghitung indeks yang ada dinegara tersebut. Mejiro untuk saat ini masih dalam proses perkembangan. Perbandingan tingkat resiko internet antara negara Jepang dan Indonesia yang dapat dilihat pada gambar 2.1.



Gambar 2. 1 *Radar Chart Analysis* Indonesia dan Jepang

2.2 Dasar Teori

Adapun beberapa teori pendukung yang penulis gunakan dalam penelitian ini adalah sebagai berikut :

2.2.1. Simulasi

Simulasi merupakan suatu metodologi untuk melaksanakan percobaan dengan menggunakan model dari suatu sistem yang nyata (Syahputri dkk., 2020). Simulasi banyak memberikan manfaat dalam penelitian karena dengan menggunakan simulasi dapat menghemat sumber daya dan waktu serta resiko

yang dimunculkan lebih sedikit dibandingkan langsung melakukan uji coba pada sistem yang sebenarnya.

2.2.2. PHP

Hypertext Preprocessor (PHP) adalah bahasa skrip yang dapat ditanamkan atau disisipkan di HTML. PHP sering digunakan sebagai *script server-side* untuk pengembangan halaman web (Sahi, 2020). Perintah PHP dieksekusi di server setelah itu dikirimkan ke browser dalam format HTML.

2.2.3. Mysql

MySQL (*My Structured Query Language*) adalah penyedia layanan untuk penyimpanan data. Mysql bersifat *Open Source* dan cocok digunakan dengan bahasa pemrograman PHP. Fitur-fitur yang ada pada Mysql dapat dilihat pada dokumentasi (Sahi, 2020).

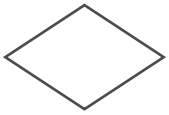
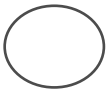
2.2.4. XAMPP

Nama XAMPP berasal dari gabungan awalan huruf dari Apache, MySQL, PHP dan Perl. X berarti *Cross Platform* yaitu aplikasi ini dapat dijalankan di berbagai sistem operasi yang berbeda. XAMPP adalah aplikasi *Web Server* yang didalamnya juga terdapat Mysql dan PHP (Santoso & Nurmalina, 2017).

2.2.5. *Flowchart*

Flowchart adalah alur yang terdiri dari beberapa symbol untuk menggambarkan suatu proses atau prosedur, masing-masing symbol pada *flowchart* memiliki arti tersendiri (Santoso & Nurmalina, 2017). Adapun simbol *Flowchart* sebagai berikut :

Tabel 2. 2 Simbol *Flowchart*

Simbol	Keterangan
	Terminator, Tanda mulai atau berakhirnya program.
	Garis Alir, Menunjukan arah aliran program .
	Preparation, Pemberian nilai awal/inisialisasi.
	Proses, Melakukan proses pengolahan data.
	Input/Output Data, Melakukan proses masukan atau pun keluaran.
	Sub Program, Menjalankan Sub program.
	Pilihan, Melakukan seleksi dan menentukan Langkah selanjutnya.
	On Page Connector, Penghubung bagian-bagian yang berada dihalaman yang sama.
	Off Page Connector, Penghubung bagian-bagian yang berada pada halaman yang berbeda.

2.2.6. Penetration Testing

Penetration Testing adalah uji coba untuk mencari celah ataupun kelemahan dari suatu sistem dengan cara menyerang sistem tersebut dengan tujuan untuk memperbaiki keamanan sistem agar meminimalisir terjadinya serangan dari oknum yang dapat merugikan pemilik sistem tersebut. (Ramadhan dkk., 2020) Penetration Testing Memiliki 6 tipe yaitu:

a. Blind

   Pentester mendapatkan hanya sedikit informasi dari sistem yang akan diserang sedangkan pihak target mengetahui dan telah mempersiapkan sistem untuk dilakukan penyerangan.

b. Double-blind

   Pentester tidak dibekali informasi yang cukup tentang sistem yang akan diserang dan target juga tidak mempersiapkan sistem untuk proses uji penyerangan.

c. Graybox

   Pihak target telah mempersiapkan sistem untuk diuji sedangkan *pentester* hanya diberitahukan sedikit informasi dari kelompok target terkait sistem yang akan diuji.

d. Double graybox

   Pihak target menyiapkan sistem untuk proses *penetration testing* dan akan memberitahukan *pentester* ruang lingkup yang akan dilakukan pengujian.

e. Tandem

Pihak target dan *pentester* menyiapkan bersama-sama sistem yang akan diuji.

f. Reversal

Pihak target tidak mempersiapkan dan tidak mengetahui sistem akan dilakukan pengujian sedangkan *pentester* mengetahui secara rinci terkait sistem yang akan diuji. Biasanya, *pentester* dan pihak target berasal dari satu organisasi yang sama.

2.2.7. CERDAS

Center Of E-Learning and Education For Students (CERDAS) adalah aplikasi berbasis *online* yang digunakan untuk memenuhi kebutuhan pembelajaran secara *online* atau *E-Learning* bagi mahasiswa yang dapat diakses melalui gadget seperti *smartphone* maupun laptop. CERDAS diluncurkan pada tanggal 12 Maret 2021. CERDAS menyediakan beberapa layanan seperti media diskusi antara mahasiswa dan dosen, tempat pengumpulan tugas, menyediakan informasi terkait perkuliahan dan menyediakan histori akademik mahasiswa.

2.2.8. *Brute Force*

Brute Force adalah salah satu bentuk penyerangan sistem dengan cara mencoba segala bentuk kombinasi untuk mencari *password* yang tepat dari suatu sistem. Serangan *brute force* dilakukan secara berulang dan akan berhenti ketika *password* ditemukan, kombinasi yang diberikan telah habis dan ketika dicegah

oleh sistem keamanan suatu sistem. *Brute force* menggunakan algoritma untuk memecahkan masalah secara sederhana. (Pratita, 2016). Keunggulan dari *brute force* adalah cara kerjanya yang sederhana namun memiliki tingkat efektifitas yang baik untuk sebagian sistem. Kelemahan dari *brute force* adalah memelurkan waktu yang cukup banyak untuk menemukan kombinasi *password.*

Brute force memiliki 2 bentuk penyerangan yaitu :

1. Melakukan serangan dengan mencoba segala kombinasi angka, huruf dan symbol untuk menemukan password yang tepat.
2. Melakukan serangan dengan cara mencoba segala kombinasi *password* yang telah ditentukan didalam kumpulan password (*wordlist)*

2.2.9. Hatch

Hatch adalah sebuah *tools brute force* berbasis Python. Pengguna harus mempersiapkan *wordlist* untuk menjalankan Hatch, dan Hatch akan melakukan penyerangan sampai *password* ditemukan atau *wordlist* yang disediakan habis. (Kody, 2019).

2.2.10. Virtual Box

Virtual box adalah sebuah perangkat lunak virtualisasi yang berguna untuk menjalankan sistem operasi lain ditempat Virtual box terinstall. Virtual box membuat sistem virtual yang tidak berhubungan dengan sistem operasi utamanya. (Soepomo, 2014).

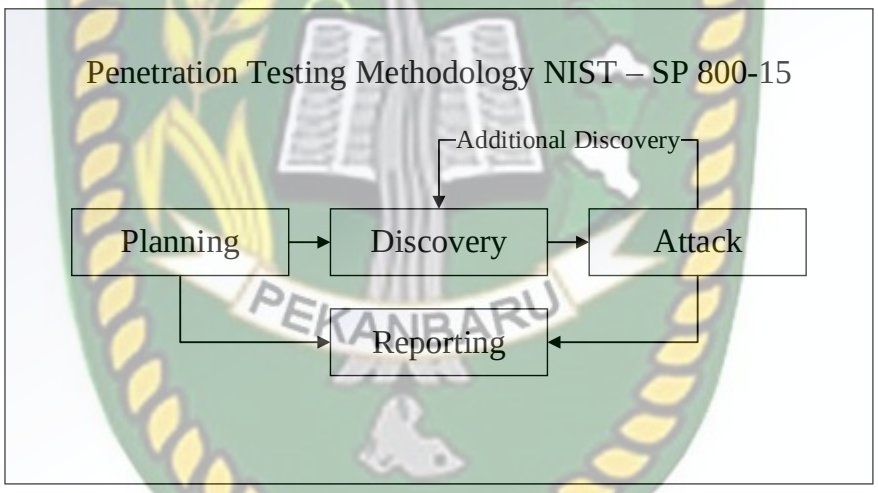
BAB III

METODOLOGI PENELITIAN

3.1 Metode Penelitian

3.1.1 NIST SP 800 - 115 (National Institute of Standard and Technology)

Mengacu pada dokumen NIST SP dengan kode 800-15 penetration testing terbagi dalam beberapa tahapan yaitu *planning*, *discovery*, *attack* dan *reporting*. Adapun alur tahapan tersebut dapat dilihat pada gambar 3.1.



Gambar 3. 1 Penetration Testing Methodology NIST

Adapun rencana yang akan penulis lakukan dalam penelitian simulasi *penetration testing* CERDAS bedasarkan metode NIST adalah :

1. *Planning*

   Adapun perencanaan dalam penetration testing ini adalah :

   a. Membangun sistem tiruan dari CERDAS.

b. Penyerangan dilakukan kepada tiruan CERDAS menggunakan sistem operasi *virtual* Kali Linux dan *tool* Hatch.

c. Membuat laporan mengenai penetration testing yang dilakukan.

2. *Discovery*

   Penulis akan mencari informasi terhadap CERDAS untuk mengidentifikasi celah keamanan dengan cara melakukan observasi dan mencari kajian literatur yang mendukung penetration testing ini.

3. *Attack*

   Pada proses *attack* penulis akan melakukan penyerangan terhadap sistem tiruan CERDAS untuk memastikan celah keamanan yang telah diidentifikasi ditahap *Discovery*.

4. *Reporting*

   Dalam tahapan *reporting* penulis akan melakukan pencatatan hasil penetration testing yang telah dilakukan.

3.2 Alat Kebutuhan Penelitian

Adapun alat yang digunakan dalam proses penelitian terdiri dari *hardware* dan *software. Hardware* yang digunakan adalah laptop dan *software* yang digunakan adalah XAMPP dan Hatch dan beberapa *software* pendukung lainnya. Virtual Box untuk mensimulasikan CERDAS, XAMPP sebagai *web server* CERDAS dan Hatch *tool* untuk melakukan *pentest.* Berikut spesifikasi laptop yang digunakan dapat dilihat pada tabel 3.1.

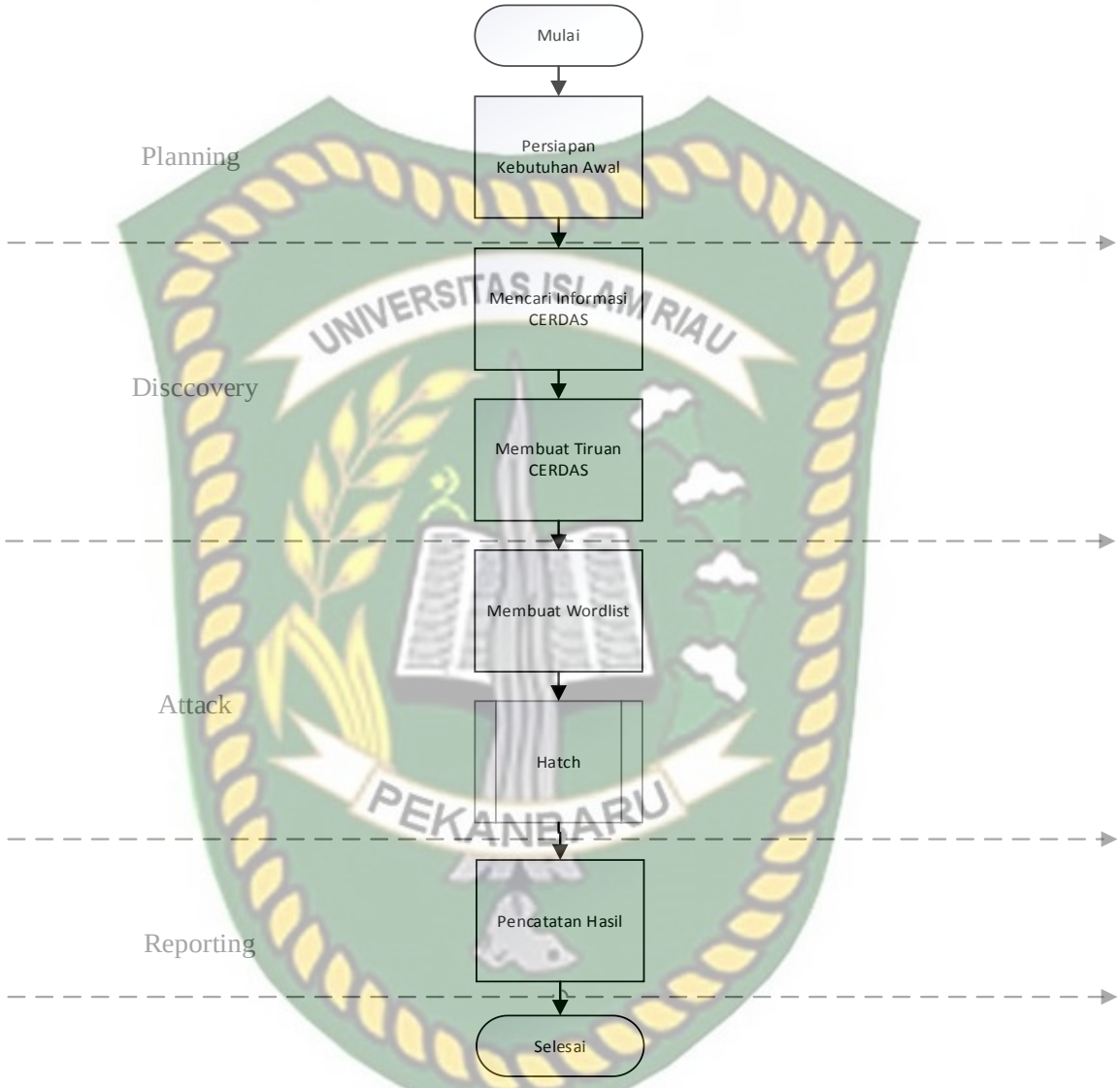
Tabel 3. 1 Spesifikasi Perangkat Yang Digunakan

Komponen	Spesifikasi	Fungsi
Processor	Intel® Core i5-10300H CPU (2.50 GHz)	Memproses instruksi yang diberikan oleh aplikasi
RAM	8 GB	Tempat penyimpanan sementara saat prosesor memproses perintah
Storage Memory	500 GB	Tempat penyimpanan aplikasi
VGA	NVIDIA GeForce GTX 1650 Ti 4GB GDDR6	Memberikan output berupa tampilan dari aplikasi

3.3 Alur Penelitian

Adapun alur dari penelitian yang akan penulis lewati adalah tahapan *planning, discovery , attack* dan *report.* Pada tahapan *planning* penulis akan melakukan persiapan dasar untuk melakukan penelitian yaitu kebutuhan dasar dari sisi *hardware* dan *software*. *Discovery* yaitu tahap untuk mencari informasi terkait tentang penelitian mulai dari observasi langsung terhadap sistem untuk mendapatkan informasi yang berguna untuk keperluan penelitian. Melakukan kajian pustaka yang berkaitan dengan penelitian yang sedang dilakukan dan membuat sebuah sistem tiruan berdasarkan informasi yang didapat. Tahapan selanjutnya adalah *attack* yaitu melakukan penyerangan terhadap model CERDAS dengan menggunakan *Hatch*. Dalam tahapan ini akan dilakukan beberapa percobaan serangan *brute force* ter*hadap* model sistem. Setelah itu hasil dari percobaan penyerangan akan dicatat dan dihitung proses persentase keberhasilan penyerangan, langkah ini disebut sebagai *reporting.*

Adapun alur tahapan dari penelitian dapat dilihat pada gambar gambar 3.2.

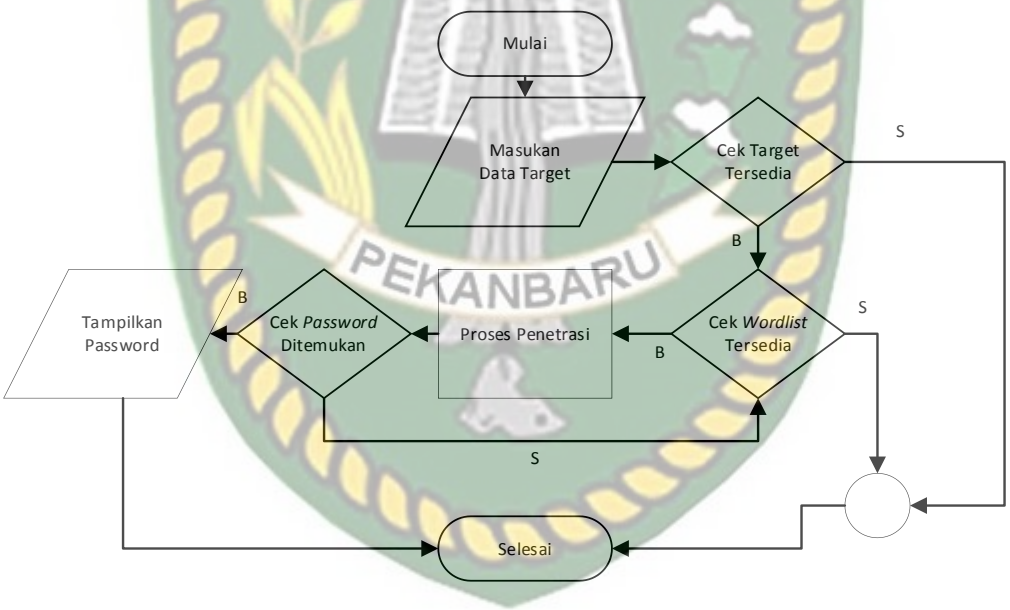


Gambar 3. 2 Alur Tahapan Penelitian

3.4 Kebutuhan Aplikasi

3.4.1 Hatch

Hatch sebelum digunakan pengguna harus menyediakan kumpulan kata (*wordlist)* yang mungkin menjadi password dari sistem target. Semakin kita mengenal sistem atau kebiasaan pengguna dalam menentukan *password* maka tingkat keberhasilan dalam memasuki sistem secara paksa semakin besar. Hatch akan melakukan serangan berulang kali sampai *wordlist* habis atau ketika Hatch menemukan kombinasi password yang tepat. Adapun alur dari cara kerja hatch seperti pada gambar 3.3.



Gambar 3. 3 Alur Cara Kerja Hatch

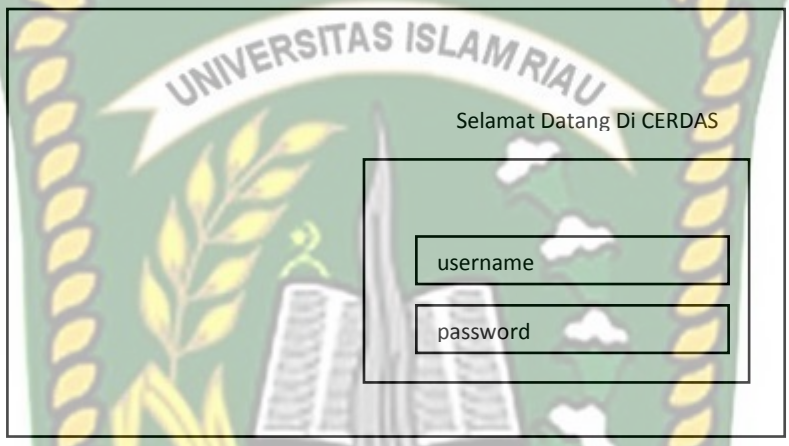
3.4.2 Model CERDAS

Desain Model CERDAS disesuaikan dengan fungsional CERDAS yang sebenarnya. Model ini memerlukan *web server* sebagai penyedia layanan agar

dapat berjalan dan disesuaikan untuk kebutuhan penelitian. Adapun desain untuk model CERDAS :

1. Desain Tatap Muka Model Halaman Masuk CERDAS

   Adapun desain tatap muka dari model CERDAS dapat dilihat pada gambar 3.4.



Gambar 3. 4 Desain Tatap Muka Halaman Masuk Model CERDAS

2. Skema Data Tabel Pengguna

   Adapun bentuk struktur dari penyimpanan data pengguna dapat dilihat pada tabel 3.2.

Tabel 3. 2 Tabel Pengguna

No	*Field*	*Type Data*	*Size*	*Description*
1.	NPM	INT	11	*Primary Key*
2.	Password	Varchar	50	Kata kunci untuk masuk sistem

3. *Flowchart* Model Halaman Masuk CERDAS

   Untuk alur dari Model halaman masuk cerdas adalah ketika pengguna mengakses alamat dari model CERDAS maka pengguna akan melalui beberapa tahapan yaitu memasukkan *username* dan *password* yang telah terdaftar. Setelah

itu sistem akan melakukan cek terhadap data yang telah dimasukkan. Jika data yang dimasukkan benar maka pengguna akan dapat masuk kesistem. Jika data yang dimasukkan salah maka pengguna akan mendapatkan pesan bahwa *username* atau *password* yang dimasukkan salah.

Adapun *flowchart* dari Model Halaman Masuk CERDAS dapat dilihat pada gambar 3.5.

Gambar 3. 5 *Flowchart* Model Halaman Masuk Model CERDAS

BAB IV

HASIL DAN PEMBAHASAN

4.1. Planning

Mempersiapkan kebutuhan awal penelitian seperti sistem operasi Kali linux dan tools hatch. Pemasangan sistem operasi kali linux dilakukan dengan secara virtual dengan menggunakan Virtual box dan pemasangan *tools* hatch akan dilakukan dalam sistem operasi kali linux. Berikut hasil dari proses *planning* :

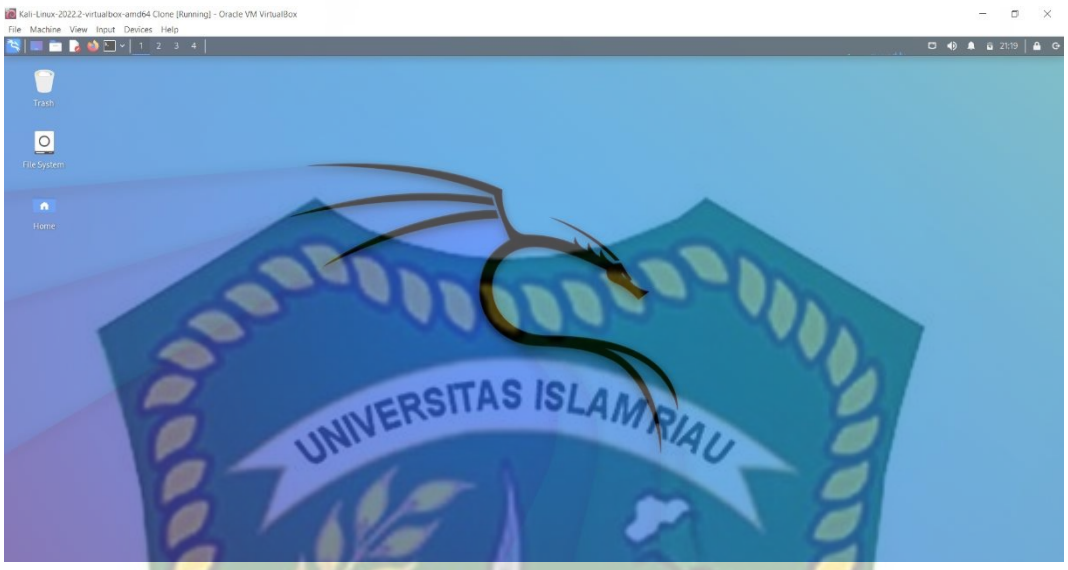
4.1.1 Pemasangan Sistem Operasi Kali Linux

Berikut spesifikasi sistem operasi Kali linux yang terpasang pada Virtual box dapat dilihat pada tabel 4.1.

Tabel 4. 1 Spesifikasi Kali Linux

Komponen	Spesifikasi	Fungsi
Processor	Intel® Core i5-10300H CPU (2.50 GHz)	Memproses instruksi yang diperintahkan dalam aplikasi virtual box
RAM	4 GB	Tempat penyimpanan sementara saat prosesor memproses perintah
Storage Memory	80 GB	Tempat penyimpanan aplikasi

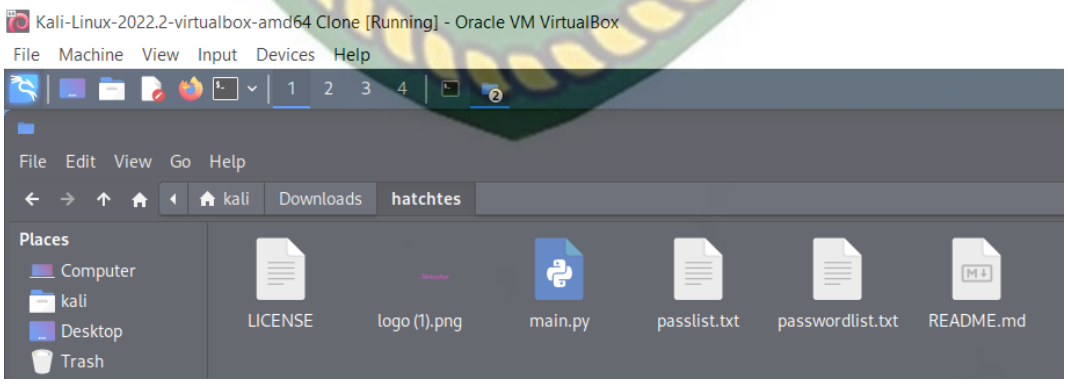
Adapun tampilan *desktop* dari Kali linux yang telah terpasang pada Virtual box pada gambar 4.1.



Gambar 4. 1 Tampilan Desktop Kali Linux

4.1.2 Pemasangan *Tool* Hatch Pada Kali Linux

Untuk pemasangan *tool* Hatch dapat dilakukan dengan memberikan perintah "git clone https://github.com/MetaChar/Hatch". Ketika Hatch telah terpasang maka akan muncul direktori bernama Hatch yang dapat dilihat pada gambar 4.2.



Gambar 4. 2 Direktori Hatch

4.2. Discovery

Setelah melakukan observasi dan beberapa percobaan masuk kedalam sistem CERDAS penulis mendapatkan informasi mengenai cara kerja masuk kedalam CERDAS, *username* dan *password* yang digunakan untuk masuk dan beberapa informasi lainnya yang dapat dilihat pada tabel 4.2.

Tabel 4. 2 Hasil Observasi

No	Informasi Yang Didapat
1.	Syarat untuk masuk kedalam sistem adalah mengisi *username* dan *password*
2.	*Username* adalah NPM mahasiswa.
3.	*Password default* adalah tanggal lahir mahasiswa.
4.	Tidak ada batasan percobaan masuk kedalam sistem CERDAS
5.	*Username* dan *password* terhubung dengan Sistem Informasi Akademik Mahasiswa Universitas Islam Riau (SIKAD UIR)
6.	Tidak ada pengecekan manusia atau robot yang mengakses sistem
7.	Menampilkan catatan kesalahan saat salah memasukkan *username* atau *password*

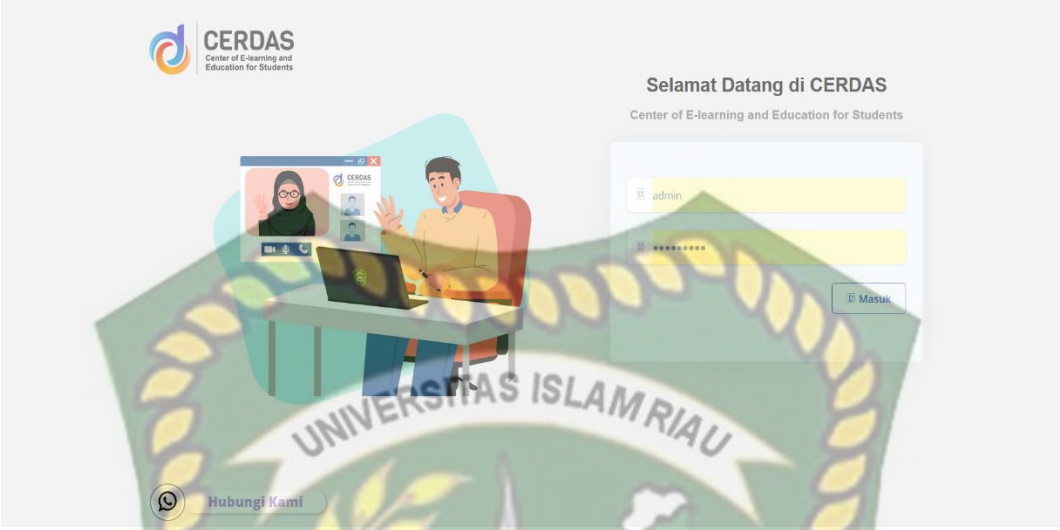
4.2.1. Pembuatan Model CERDAS

Pembuatan model CERDAS dilakukan secara lokal dan dibuat dengan menggunakan bahasa PHP. Model CERDAS memanfaatkan XAMPP sebagai penyedia layanan basis data dan *web server*.

Dalam proses pembuatan sistem tiruan cerdas dibagi menjadi 2 langkah yaitu :

a. Pembuatan Tampilan Tiruan CERDAS

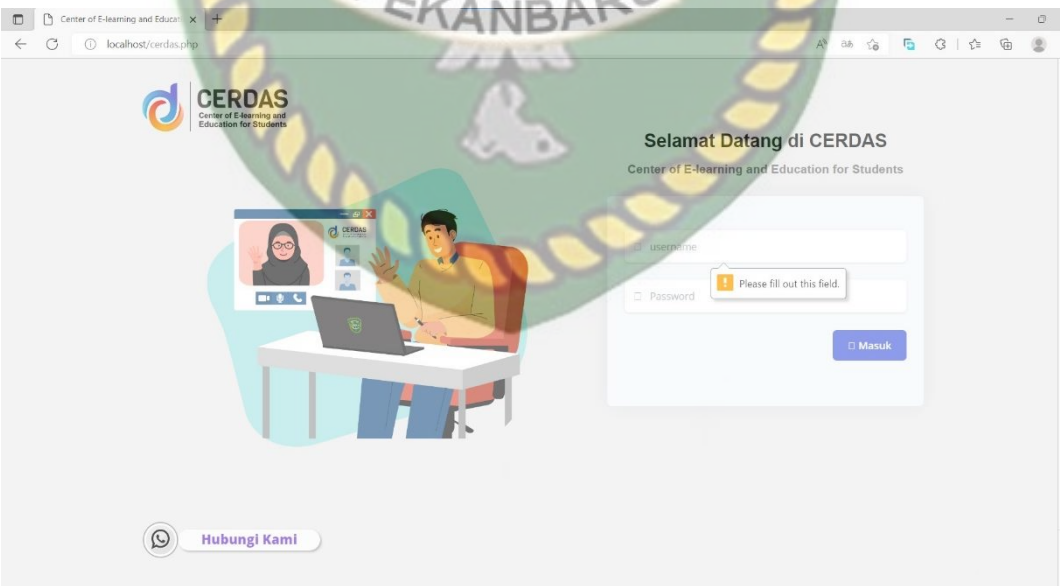
Adapun tampilan hasil dari pembuatan tampilan tiruan cerdas dapat dilihat pada gambar 4.1.



Gambar 4. 3. Tampilan Halaman Masuk Model CERDAS

b. Pengujian *Blackbox* Tampilan Halaman Masuk Tiruan CERDAS

Jika *username* atau *password* tidak diisi maka akan muncul pesan "*Please fill out this form"* yang dapat dilihat pada gambar 4.2.



Gambar 4. 4 Catatan *Please Fill Out This Form*

Jika memasukkan *username* / *password* yang salah maka akan muncul catatan "Gagal Login, Username atau Password Salah" seperti yang dapat dilihat pada gambar 4.3.

Gambar 4. 5 Catatan Gagal Login Username atau Password Salah

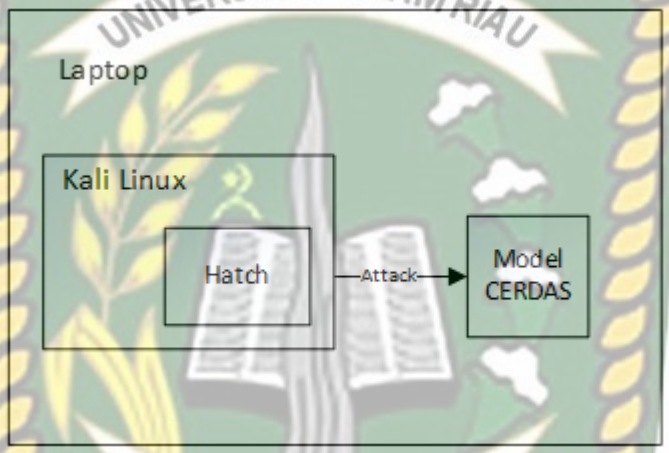
Kesimpulan dari pengujian Halaman Masuk Tiruan Cerdas dapat dilihat pada tabel 4.3.

Tabel 4. 3 Kesimpulan Pengujian

Skenario Pengujian	*Test Case*	Hasil Yang Diharapkan	Hasil Pengujian
Form Username dan *Password*	Mengosongkan *field username* dan *pasword*	Menampilkan catatan "Fill Out This Field "	Sesuai Harapan
	Mengosongkan *field username* atau *password*		
	Mengisi field *username* atau *password* salah	Menampilkan catatan "Gagal Login, *Username* atau *Password* salah	Sesuai Harapan

4.3 *Attack*

Penyerangan dilakukan melalui *Hatch* yang terpasang pada sistem operasi virtual Kali linux dengan lokasi target yaitu model CERDAS yang terpasang pada laptop utama. Berikut gambaran dari skema penyerangan model CERDAS yang dapat dilihat pada gambar 4.4.



Gambar 4. 6 Skema Penyerangan

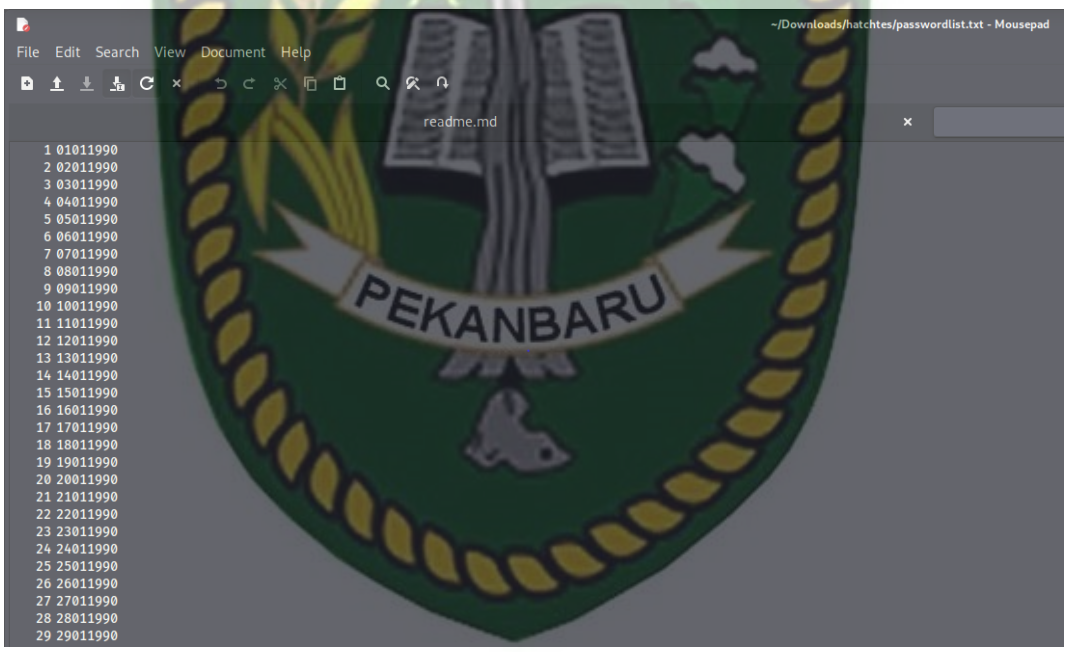
4.3.1 Pembuatan Wordlist

Sebelum melakukan penyerangan harus menyediakan kumpulan data yang digunakan untuk pencarian *password.* Pembuatan kumpulan password dibagi menjadi 2 yaitu pembuatan kumpulan *password* yang berupa tanggal lahir dilakukan karena *password default* CERDAS adalah tanggal lahir sehingga target dari *wordlist* ini adalah pengguna yang masih menggunakan *password default* dari CERDAS dan yang kedua adalah membuat kumpulan *password* yang berupa kumpulan yang terdiri dari kata serta kombinasi angka yang umum digunakan.

Berikut adalah proses pembuatan dari *wordlist :*

1. Pembuatan *wordlist* tanggal lahir

Pembuatan wordlist tanggal lahir menggunakan *tool* berbasis python bernama date-generator. dengan format perintah “python3 date_generator.py {starting year} {ending year} {display format} {separator}”. Pembuatan wordlist dimulai dari tahun 1990 hingga tahun 2005 yang disimpan dengan nama passwordlist.txt. Berikut tampilan hasil dari kumpulan wordlist yang telah dibuat dan dapat dilihat pada gambar 4.7.

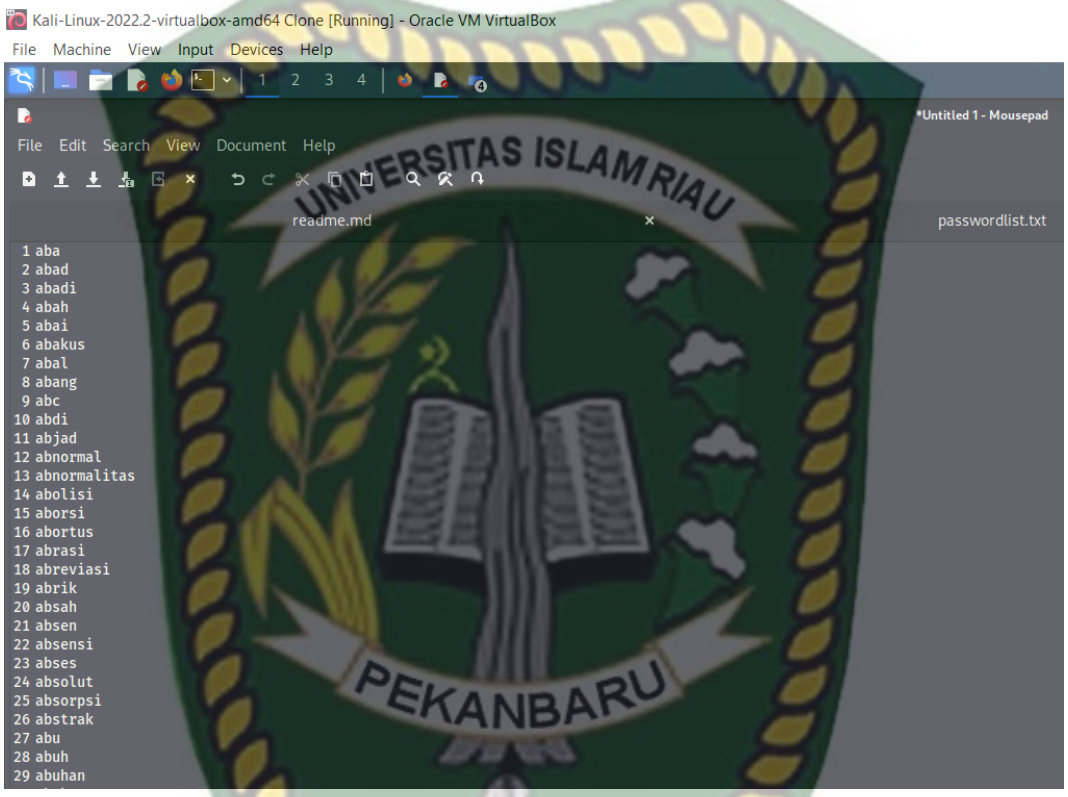


Gambar 4. 7 Kumpulan Wordlist Tanggal Lahir

2. Pembuatan *Wordlist* Umum

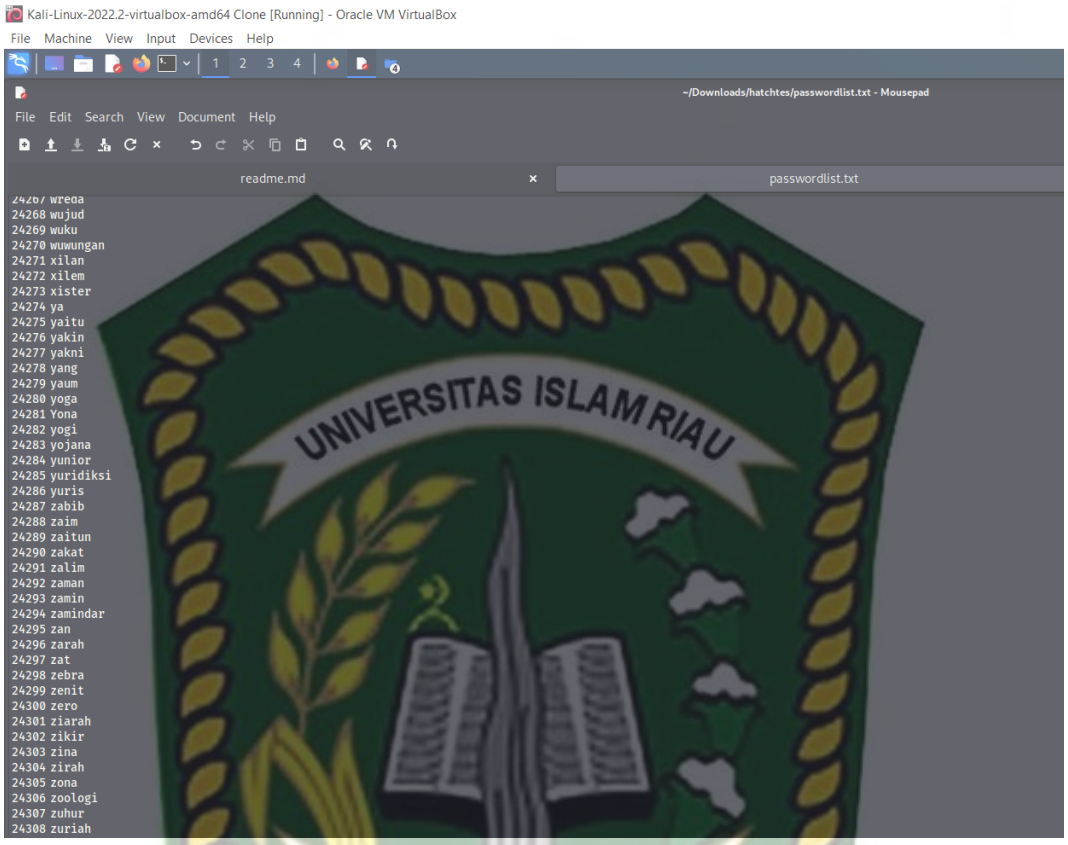
Dalam pembuatan *Wordlist* umum penulis mengambil dari halaman “https://github.com/geovedi/indonesian-wordlist/blob/master/05-

ivanlanin2011-sort-alpha.lst" yang menyediakan kumpulan *wordlist* sebanyak 18313 baris wordlist. Adapun gambar beberapa kumpulan *wordlist* umum ini dapat dilihat pada gambar 4.8



Gambar 4. 8 Kumpulan *Wordlist* Umum

Kumpulan *wordlist* disatukan dalam satu *file* yang bernama passwordlist.txt sehingga total kumpulan *wordlist* adalah 24308 baris *wordlist* dapat dilihat pada gambar 4.9.



Gambar 4. 9 Total *Wordlist* Hatch

4.3.2 Daftar *Username* Yang Akan Diserang

Penulis akan menyediakan daftar nama pengguna beserta *password* yang akan diserang oleh Hatch untuk memastikan apakah Hatch bisa menemukan *password* dari nama pengguna tersebut. Berdasarkan informasi yang didapat ditahapan *discovery*, penulis mengasumsikan bahwasannya pengguna CERDAS terbagi 2 yaitu pengguna yang masih menggunakan *password default* dan pengguna yang sudah mengganti *passwordnya*. Berikut adalah daftar pengguna yang akan diserang oleh Hatch pada tabel 4.4.

Tabel 4. 4 Daftar Penyerangan

No	*Username*	*Password*	Kelompok
1	173510000	05101998	A
2	173510002	03021995	A
3	173510003	11051994	A
4	193510001	08062001	A
5	183510787	09072000	A
6	203510021	11101999	A
7	203510566	05042002	A
8	203510723	03022003	A
9	181032001	09091999	A
10	191103099	07062000	A
11	173510001	Ferdi198@	B
12	203510001	11111111	B
13	213510007	12345678	B
14	203510002	Administrator	B
15	219978033	DarkStorm123	B
16	209873105	Prasetyo1997	B
17	167809998	Andilaw3124	B
18	175789801	PasukanBumi89	B
19	191078567	Monkeydluffy12	B
20	207219876	MobelLejen125	B

Keterangan :

A = Kelompok pengguna yang masih menggunakan *password default.*

B = Kelompok pengguna yang telah mengganti *password*

Total target penyerangan adalah 20 pengguna yang terdiri dari kelompok A dan B. hasil penyerangan antara kelompok A dan B akan dibandingkan untuk menentukan kelompok yang lebih aman terhadap serangan *brute force*.

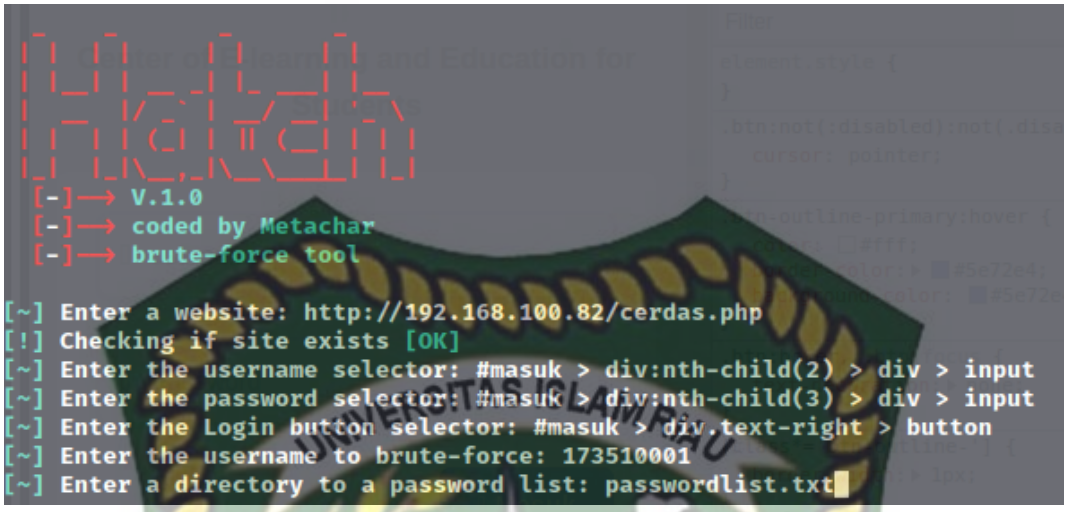
4.3.3 Pengecekan Target

Penyerangan dimulai dengan melakukan pengecekan dari target yang akan diserang. Jika target tersedia maka akan menampilkan tulisan “OK” seperti pada gambar 4.10.

Gambar 4. 10 Pengecekan Alamat Target

4.3.4 Mengisi Data Target

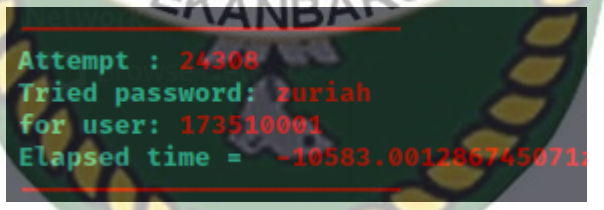
Dalam proses ini *user* perlu memasukan data yang diperlukan oleh Hatch untuk melaksanakan penyerangan seperti *selector CSS form login*. *User* juga perlu memasukan *username* dari target yang ingin diserang dan lokasi dari *wordlist* yang telah disediakan. Proses ini dapat dilihat pada gambar 4.11.



Gambar 4. 11 Penginputan Data Hatch

4.3.5 Proses Penyerangan

Hatch akan memberikan hasil yang berupa jumlah percobaan, *password* yang digunakan,total waktu yang digunakan dari penyerangan seperti pada gambar 4.12.



Gambar 4. 12 Gambar Hasil Penyerangan Hatch

Jika Hatch menemukan *password* maka *tools* akan berhenti secara langsung. Dan jika *tools* tidak menemukan *password* sampai *wordlist* habis maka *tools* akan menunggu untuk *wordlist* yang baru. Untuk menambahkan *wordlist* baru maka *tools* harus dihentikan terlebih dahulu dengan menekan *ctrl + c*.

4.4 *Reporting*

Dalam tahapan *reporting* dilakukan pencatatan hasil dari penyerangan yang telah dilakukan dan dibuat dalam bentuk tabel terpisah antara percobaan serangan terhadap *password default* dan *password* yang telah diganti.

4.4.1 Penyerangan terhadap *username* yang belum mengganti *password* ataupun masih menggunakan *password default*

Setelah melakukan beberapa kali percobaan penyerangan berikut adalah hasil dari perobaan penyerangan terhadap pengguna yang masih menggunakan *password default* yang dapat dilihat pada tabel 4.5.

Tabel 4. 5 Hasil Penyerangan Kelompok A

No	*Username*	Total Eksekusi Wordlist	Waktu Yang Dihabiskan (Detik)	Password Ditemukan
1	173510000	3261	1041	✓
2	173510002	1895	814	✓
3	173510003	1624	711	✓
4	193510001	4256	1828	✓
5	183510787	3916	1993	✓
6	203510021	3639	1970	✓
7	203510566	4563	2104	✓
8	203510723	4871	1712	✓
9	181032001	3606	1439	✓
10	191103099	3883	1545	✓

Berdasarkan hasil percobaan penyerangan dari tabel 4.4 keberhasilan untuk menemukan *password* dari kelompok A adalah 100% .

4.4.2 Penyerangan terhadap *username* yang telah mengganti *password.*

Hasil dari penyerangan terhadap pengguna yang telah merubah *password* dapat dilihat pada tabel 4.6.

Tabel 4. 6 Hasil Penyerangan Kelompok B

No	*Username*	Total Eksekusi Wordlist	Waktu Yang Dihabiskan (Detik)	Password Ditemukan
1	173510001	24308	10583	×
2	203510001	5954	2489	✓
3	213510007	5964	2590	✓
4	203510002	6063	3419	✓
5	219978033	24308	9196	×
6	209873105	24308	12500	×
7	167809998	24308	11974	×
8	175789801	24308	10877	×
9	191078567	24308	11378	×
10	207219876	24308	12577	×

Berdasarkan tabel 4.5 keberhasilan untuk menemukan password dari kelompok B adalah 30%.

Hasil akhir dari kedua percobaan tersebut jika disatukan maka keberhasilannya untuk menemukan password adalah 13 dari 20 kali percobaan dengan tingkat keberhasilan 65% dan tingkat keberhasilan pelaksanaan serangan *brute force* menggunakan Hatch adalah 20 dari 20 kali percobaan yaitu 100%. Berdasarkan hasil dari penyerangan dapat disimpulkan bahwasannya kelompok B memiliki tingkat keamanan lebih baik, karena pada penyerangan *brute force dictionary attack* menggunakan *password* yang sering digunakan oleh pengguna secara umum, sehingga semakin kompleks *password* yang digunakan, maka semakin sulit untuk *password* itu ditemukan.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Setelah melakukan simulasi *penetration testing* terhadap CERDAS dengan metode *brute force* menggunakan *hatch* dapat disimpulkan bahwa.

1. CERDAS memiliki kerentanan terhadap serangan *brute force* karena kurangnya verifikasi keamanan pada halaman masuk kedalam sistem.
2. Setelah dilakukan percobaan penyerangan, *CERDAS* 100% tidak aman dari serangan *brute force* dengan tingkat keberhasilan pelaksanaan penyerangan *brute force* yaitu 20 dari 20 kali percobaan, tingkat keberhasilan menemukan *password* adalah 10 dari 10 kali percobaan atau setara dengan 100% terhadap pengguna yang masih menggunakan *password default*, tingkat keberhasilan terhadap pengguna yang telah mengganti *password* yaitu 3 dari 10 kali percobaan atau setara dengan 30% dan tingkat keberhasilan gabungan dari percobaan menemukan *password* terhadap pengguna yang masih menggunakan *password default* maupun yang telah mengganti *password default* adalah 13 dari 20 kali percobaan atau setara dengan 65%.
3. Penggunaan *password* yang kompleks akan lebih aman terhadap serangan *brute force* daripada penggunaan *password default.*

5.2. Saran

Berdasarkan hasil kesimpulan dan analisis yang dilaksanakan penulis menyarankan kepada CERDAS :

1. Memberikan verifikasi tambahan terhadap pengguna yang ingin masuk kedalam sistem untuk meminimalisir atau mencegah percobaan serangan yang dilakukan secara sistem atau robot.
2. Memberikan batasan percobaan masuk agar mencegah percobaan serangan secara berulang terhadap sistem.
3. Mewajibkan pengguna mengganti *password* dan menyarankan untuk menggunakan *password* yang kompleks seperti gabungan antara kata, angka dan simbol pada saat pertama kali mengakses CERDAS agar keamanan lebih terjamin.

Saran penulis untuk penelitian selanjutnya adalah :

1. Menggunakan metode penelitian yang berbeda dalam melakukan penelitian seperti metode ISSAF (*Information System Security Assessment)*
2. Menggunakan *tools* yang berbeda seperti Burp Suite, Hydra atau yang lainnya.
3. Menggunakan kumpulan *wordlist* yang berbeda.

DAFTAR PUSTAKA

Adrian, A., & Setiyadi, A. (2018). Analisis Keamanan Jaringan Dengan Metode Penetration Testing Execution Standard (Ptes) Di Dinas Kesehatan Provinsi Jawa Barat. Jurnal Unikom Repisitory, 1, 1–8.

Alanda, A., Satria, D., Isthofa Ardhana, M., Dahlan, A. A., & Mooduto, A. (2021). International Journal On Informatics Visualization journal homepage : www.joiv.org/index.php/joiv International Journal On Information Visualization Web Application Penetration Testing Using SQL Injection Attack. 5(September), 320–326. www.joiv.org/index.php/joiv

Amijoyo, T., Umar, R., & Yudhayana, A. (2020). Bruteforce In The Hydra Process And Telnet Service Using The Naïve Bayes Method. Jurnal Mantik, Volume 4 N. https://iocscience.org/ejournal/index.php/mantik/index

Azis, R., & Yazid, S. (2021). Pengujian Kerentanan Website Wordpress Dengan Menggunakan Penetration Testing. 3(3), 93–105.

Dewanto, A. P. (2018). Penetration Testing pada Domain uii.ac.id Menggunakan OWASP 10. Https://Dspace.Uii.Ac.Id/. https://dspace.uii.ac.id/bitstream/handle/123456789/11281/13523025-Adetya Putra D-laporan skripsi.pdf?sequence=1&isAllowed=y

Haeruddin, H., & Kurniadi, A. (2021). Analisis Keamanan Jaringan WPA2-PSK Menggunakan Metode Penetration Testing (Studi Kasus: TP-Link Archer A6). CoMBInES-Conference on Management …, 1(1), 508–515. https://journal.uib.ac.id/index.php/combines/article/view/4475

Hidayatulloh, S., & Saptadiaji, D. (2021). Penetration Testing pada Website Universitas ARS Menggunakan Open Web Application Security Project (OWASP). Jurnal Algoritma, 18(1), 77–86. https://doi.org/10.33364/algoritma/v.18-1.827

Kody. (2019). Brute Force Nearly Any Website Login with Hatch, diakses November 2021.(https://null-byte.wonderhowto.com/how-to/brute-force-nearly-any-website-login-with-hatch-0192225/).

Paramitha. (2011). Implementasi Penetrasi Testing Untuk Mengetahui Keamanan Penggunaan Aplikasi Sosial Media Menggunakan Metode Action Reserch.

Pratama, D. A., Erlansyah, D., & Panjaitan, F. (2019). Penerapan Algoritma Brute Force Pada Sistem Informasi Akademik Universitas Bina Darma. Bina Darma Conference on Computer Science, 1032–1038. https://kominfo.go.id

Pratita, H. S. (2016). Analisa Brute Force Attack Menggunakan Scanning Aplikasi Pada HTTP Attack. 2016, 672010194.

Ramadhan, R. A., Aresta, R. M., & Hariyadi, D. (2020). Sudomy: Information Gathering Tools for Subdomain Enumeration and Analysis. IOP Conference Series: Materials Science and Engineering, 771(1). https://doi.org/10.1088/1757-899X/771/1/012019

Sahi, A. (2020). Aplikasi Test Potensi Akademik Seleksi Saringan Masuk Lp3I Berbasis Web Online Menggunakan Framework Codeigniter. Tematik, 7(1), 120–129. https://doi.org/10.38204/tematik.v7i1.386

Sastra Wardaya, M. S. (2019). Penetration Testing Terhadap Website Asosiasi Pekerja Profesional Informasi Sekolah Indonesia (APISI). Skripsi, 11(1), 1–14.

Soepomo, P. (2014). Analisis dan Perancangan Proxy Server. 2, 1–9.

Syahputri, T. A., Az-zahra, T. S., Setifani, N. A., Ningrum, K. P., & Rolliawati, D. (2020). Pemodelan Dan Simulasi Proses Produksi Peralatan Bayi Pada Home Industri Puppy Putra Perdana. JUST IT : Jurnal Sistem Informasi, Teknologi Informasi Dan Komputer, 11(1), 24. https://doi.org/10.24853/justit.11.1.24-31