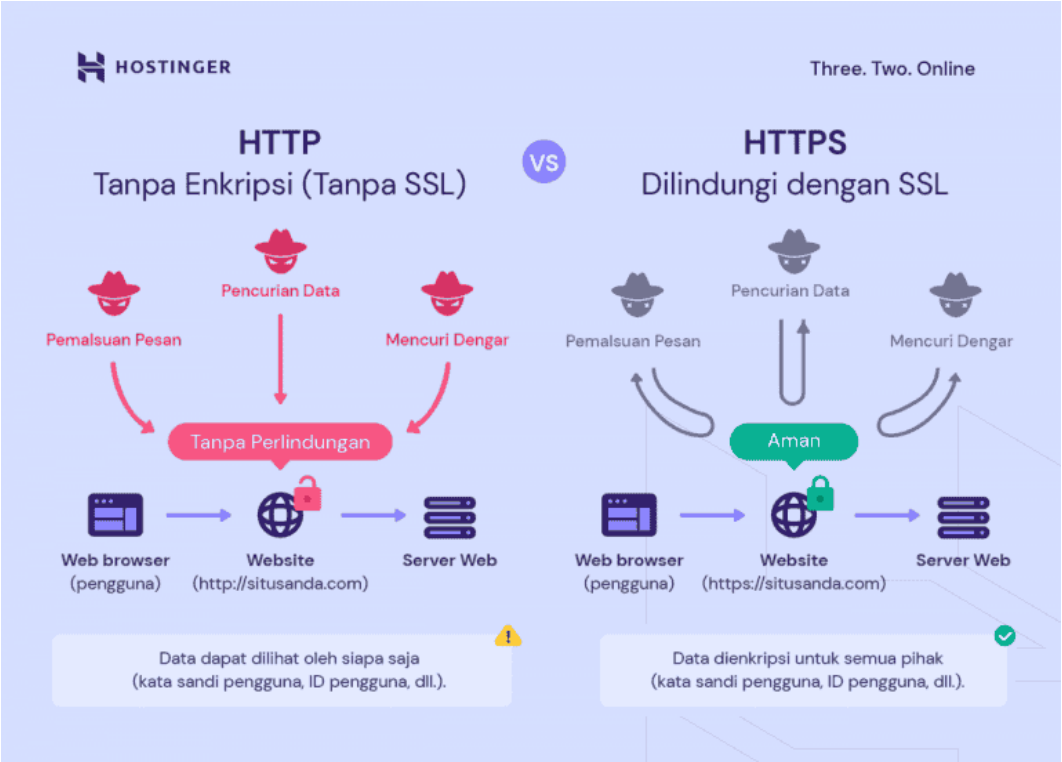
BAB 2
LANDASAN TEORI

2.1 Konsep Dasar Website

Website merupakan sekumpulan halaman web yang saling berkaitan dan berisi berbagai elemen seperti teks, gambar, suara, video, dan animasi [16]. Secara terminologi, website merupakan kumpulan dari halaman situs yang terangkum dalam sebuah domain atau subdomain yang berada di dalam *World Wide Web* (WWW) pada Internet. Format dokumen pada suatu halaman web ditulis dalam format HTML (*Hyper Text Markup Language*) yang merupakan suatu bahasa pemrograman yang bisa diakses melalui protokol HTTP untuk menyampaikan informasi. *World Wide Web* (WWW) [17] atau yang lebih dikenal sebagai Web, telah menjadi medium utama untuk penyebaran informasi. *World Wide Web* merupakan merupakan sebuah ruang informasi global yang memungkinkan pengguna mengakses dan berinteraksi dengan berbagai sumber daya melalui jaringan internet. Agar sistem ini dapat berfungsi, terdapat dua komponen krusial yang menjadi dasarnya, yaitu Website dan *Uniform Resource Locator* (*URL*) [18]. Halaman ini dapat diakses secara daring (online) oleh individu, organisasi, perusahaan, instansi pendidikan, pemerintahan, dan instansi lainnya. Dengan adanya website, setiap individu bisa mendapatkan informasi yang dibutuhkan secara cepat dan mudah, serta pemilik website dapat menyebarkan informasi, melakukan promosi, serta menjalin komunikasi atau interaksi melalui platform tersebut. Secara esensial, sebuah website berfungsi sebagai representasi digital dari individu, organisasi, atau entitas tertentu yang bertujuan untuk menyediakan informasi, menawarkan layanan, atau memfasilitasi komunikasi [16]. Setiap website memiliki sebuah halaman utama yang disebut homepage. *Homepage* merupakan tampilan standar atau default dari sebuah situs web dan menjadi gerbang awal bagi pengunjung untuk menavigasi konten lebih lanjut. Navigasi antar halaman dimungkinkan melalui penggunaan *hyperlink*, yaitu sebuah tautan yang menghubungkan satu halaman *web* ke halaman lainnya. Aktivitas mencari dan menjelajahi halaman web untuk mendapatkan informasi inilah yang dikenal dengan istilah *browsing*.

2.2 HyperText Transfer Protocol (HTTP/HTTPS)

HTTP (*Hypertext Transfer Protocol*) merupakan sebuah protokol pada lapisan aplikasi (*Application Layer*) yang menjadi dasar bagi sistem informasi hypermedia yang bersifat *distributed* dan *collaborative*. Protokol ini memiliki karakteristik generic dan stateless, yang berarti tidak menyimpan status dari request sebelumnya. Sifat *generic* ini memungkinkan HTTP untuk diimplementasikan pada berbagai tugas di luar penggunaan utamanya untuk *hypertext*, seperti pada *name servers* dan *distributed object management systems*, melalui mekanisme extension pada *request methods*, *error codes*, dan *headers*. Salah satu fitur fundamental dari HTTP adalah proses *typing* dan *negotiation of data representation*, yang memungkinkan sistem untuk dibangun secara independently dari data yang sedang di-transfer. Sedangkan HTTPS (*Hypertext Transfer Protocol Secure*) merupakan versi aman dari HTTP yang mengenkripsi data antara peramban (*browser*) Anda dan situs web. Dengan HTTPS, komunikasi data dilindungi oleh enkripsi melalui protokol seperti SSL (*Secure Sockets Layer*) atau TLS (*Transport Layer Security*), sehingga data sensitif seperti informasi masuk atau pembayaran tidak dapat dicegat dan dibaca oleh pihak ketiga [19].



Gambar 2.1. Perbedaan Protokol HTTP dan HTTPS.

Cara kerja SSL Certificate terdiri dari tiga tahapan penting yakni *Handshake Protocol*, *Record Protocol*, dan *Alert Protocol* yang masing-masing berfungsi untuk melakukan negosiasi enksipsi dan autentikasi, menjaga kerahasiaan serta integritas data selama proses transmisi, serta memberikan peringatan apabila terjadi kesalahan atau ancaman keamanan. Walaupun HTTPS meningkatkan keamanan komunikasi data, pada praktiknya banyak situs phishing modern juga menggunakan HTTPS untuk meningkatkan kredibilitas palsu di mata korban. Kondisi ini menunjukkan bahwa protokol komunikasi saja tidak cukup untuk membedakan antara situs aman dan berbahaya. Oleh karena itu, analisis keberadaan suatu URL menjadi aspek penting dalam penelitian deteksi phishing berbasis URL. Analisis terhadap struktur URL, panjang URL, penggunaan karakter khusus, jumlah subdomain, serta pola penamaan domain yang menyerupai situs resmi menjadi faktor penting dalam mengidentifikasi phishing. Pendekatan berbasis karakteristik URL ini memungkinkan sistem deteksi phishing untuk bekerja secara lebih efektif, terlepas dari jenis protokol komunikasi yang digunakan oleh situs web tersebut.

2.3 Teori Dasar Cyber Security

Cyber security (keamanan siber) merupakan suatu upaya perlindungan terhadap individu, organisasi, sistem, dan teknologi dari berbagai aktivitas abnormal atau berbahaya yang terjadi di ruang lingkup digital dan siber. *Cyber security* juga didefinisikan sebagai proses menjaga kerahasiaan (*confidentiality*), integritas (*integrity*), dan ketersediaan (*availability*) sumber daya komputer yang dimiliki oleh suatu organisasi atau yang terhubung dengan jaringan organisasi lainnya. Ketiga aspek tersebut dikenal sebatai CIA Triad dalam menjadi fondasi utama dalam penerapan keamanan siber. Pesatnya peningkatan ketergantungan masyarakat terhadap teknologi informasi dan internet, *cyber security* menjadi elemen yang sangat krusial dalam menjaga stabilitas ekonomi, sosial, serta operasional sistem informasi. Seluruh sektor strategis seperti perbankan, kesehatan, pemerintahan, pendidikan, dan transportasi telah memanfaatkan teknologi digital dalam menjalankan tugas dan aktivitasnya yang berpotensi menimbulkan celah kerentanan dan menjadi target sasaran serangan siber [20]. Hal tersebut menjadi kutipan kata yang dinyatakan oleh Kevin Mitnick dan menjadi sebuah quotes berupa *"Keamanan siber adalah tentang mengurangi risiko. Tidak ada sistem yang sepenuhnya aman, tetapi kita bisa membuatnya lebih sulit untuk diserang."*. Tanpa penerapan *cyber security* yang memadai, sistem informasi rentan terhadap

ancaman seperti pencurian data, penyalahgunaan informasi, gangguan layanan, dan serangan siber lainnya yang dapat menimbulkan kerugian besar. *Cyber security* berperan sebagai mekanisme utama dalam mencegah kejahatan siber dan serangan digital dengan memastikan interaksi digital berlangsung secara aman dan terpercaya. Penerapan *cyber security* tidak hanya melibatkan aspek teknis, tetapi juga mencakup kebijakan, regulasi, serta kesadaran pengguna dalam menjaga keamanan sistem informasi. Dengan demikian, *cyber security* menjadi komponen fundamental dalam mendukung penggunaan teknologi informasi yang aman dan berkelanjutan di era digital.

Gambar 2.2. Kutipan Kevin David Mitnick.

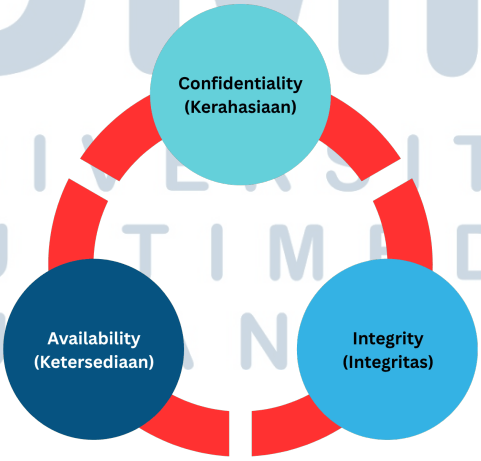
2.4 Metode CIA Triad

CIA Triad merupakan suatu metode yang digunakan dalam pengembangan kebijakan keamanan untuk mengidentifikasi masalah dan solusi yang dibutuhkan untuk keamanan dan sistem informasi [21]. CIA Triad terdiri dari tiga sifat penting meliputi [22]:

a. *Confidentiality* (Kerahasiaan) merujuk pada langkah-langkah yang diambil untuk memastikan bahwa privasi atau kerahasiaan sebuah aplikasi web tetap terjaga, sehingga hanya orang yang memiliki izin yang dapat mengaksesnya. Contohnya adalah penerapan fitur blokir akses langsung (*block direct*) agar pengguna yang tidak berwenang tidak bisa langsung mengakses data atau halaman tertentu.

b. *Integrity* (Integritas) adalah keaslian dan kepercayaan suatu data dimana data tidak boleh diubah oleh pihak yang tidak berwenang. Data harus tetap utuh dan dapat dipercaya. Tindakan yang perlu dilakukan seperti penyaringan data dan pengelolaan pengguna, sehingga hanya pengguna yang berwenang dapat mengubah atau mempengaruhi data tersebut.

c. *Availability* (Ketersediaan) mengacu pada langkah-langkah yang menjamin bahwa data dan layanan dalam web app tetap dapat diakses oleh pengguna yang berwenang kapan saja dibutuhkan. Sebagai contoh, penerapan sistem autentikasi seperti panggunaan *username* dan *password* sebelum pengguna dapat mengakses data dalam aplikasi tersebut.



Gambar 2.3. Elemen CIA Triad.

Dalam konteks ancaman phishing, ketiga elemen CIA Triad berperan penting dalam menjelaskan dampak yang ditimbulkan dari serangan siber phishing. Serangan phishing pada umumnya menargetkan aspek *Confidentiality* (Kerahasiaan), yaitu dengan mencuri informasi sensitif seperti kredensial login, data pribadi, maupun informasi finansial pengguna melalui URL palsu yang disebar. Dari aspek *Integrity* (Integritas), phishing dapat mengganggu keaslian data dan komunikasi, karena korban diarahkan ke situs tiruan yang seolah-olah sah, sehingga data yang dimasukkan tidak lagi sesuai dengan tujuan sebenarnya. Sedangkan dari aspek *Availability*, phishing berpotensi membuka akses bagi penyerang untuk melakukan eksploitasi lebih lanjut, misalnya menyisipkan *backdoor*, mengambil alih kontrol perangkat, pengambilalihan akun, penyisipan *malware*, atau merusak sistem layanan agar tidak dapat digunakan pengguna yang berhak. Oleh karena itu, upaya deteksi phishing termasuk melalui analisis URL menggunakan algoritma *Random Forest* dapat memenuhi tujuan CIA Triad, yakti menjaga kerahasiaan, integritas, dan ketersediaan informasi.

2.5 Cyber Security Threat

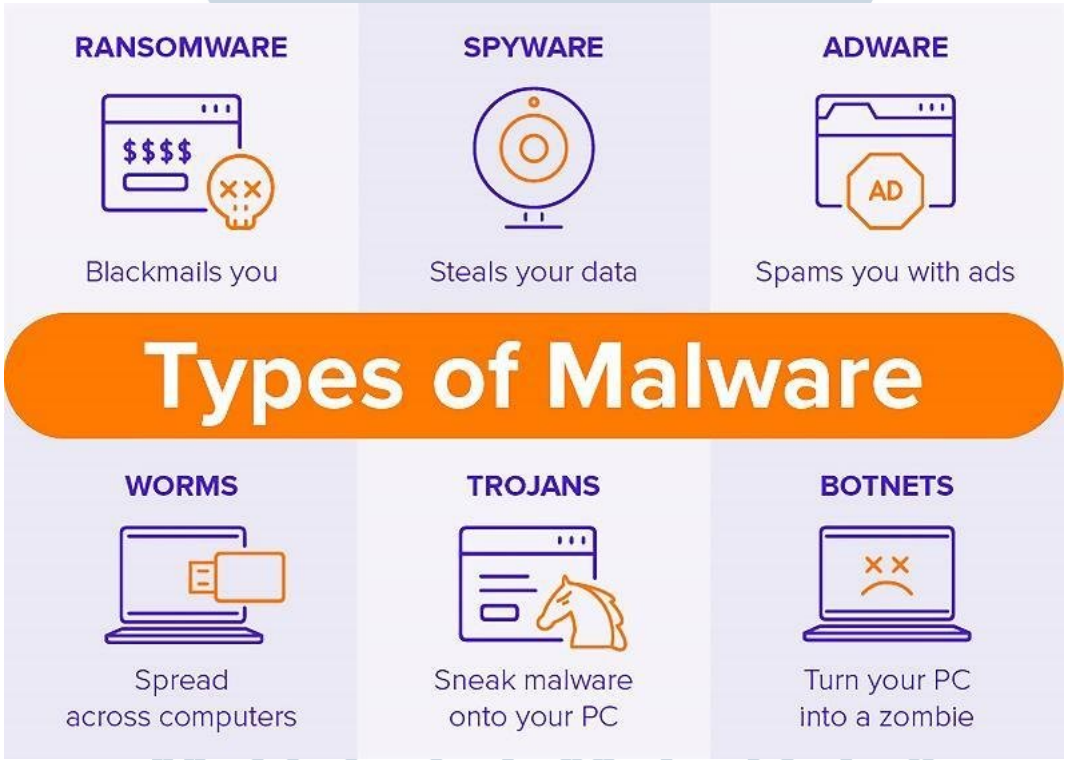
Cyber threat atau ancaman siber merupakan potensi kejadian, kondisi, atau tindakan yang dapat mengeksploitasi kerentanan (*vulnerable*) dan kelemahan pada sistem informasi, jaringan, atau data digital. Ancamana kejahatan yang dapat berpotensi mengganggu kerahasiaan (*confidentiality*), integritas (*integrity*), dan ketersediaan (*availability*) informasi. *Cyber threat* tidak selalu berbentuk serangan yang terjadi secara langsung, namun sebuah resiko yang dapat berkembang seiring berjalannya waktu apabila tidak dilakukan mitigasi dan antisipasi dengan baik. Ancaman siber dapat berasal dari berbagai sumber, baik internal maupun eksternal suatu organisasi. Ancaman internal umumnya disebabkan oleh kelalaian manusia, pengguna, dan penyalahgunaan hak akses oleh pihak yang memiliki otoritas. Sementara itu, ancaman eksternal berasal dari luar pihak yang berupaya mengeksploitasi celah keamanan melalui jaringan internet dan teknik rekayasa sosial [23]. Pemahaman cyber threat sangat penting dalam perancangan sistem keamanan informasi yang efektif dan berkelanjutan.

2.5.1 Macam-macam Cyber Threat

Cyber Threat atau ancaman siber dapat diklasifikasikan ke dalam beberapa jenis berdasarkan karakteristik dan metode yang digunakan, antara lain sebagai berikut.

a. *Malware Threat*

Malware atau *malicious software* merupakan sebuah program yang secara sengaja dirancang untuk merusak sistem komputer maupun penggunanya. Pelaku ancaman memanfaatkan *malware* untuk memperoleh akses tanpa izin, melumpuhkan sistem yang terinfeksi, merusak atau menghapus data, serta mencuri informasi sensitif yang penting bagi keberlangsungan dan keamanan sistem operasi. Malware terdapat beberapa jenis terdiri dari *ransomware*, *spyware*, *adware*, *worms*, *trojans*, dan *botnets* [24].

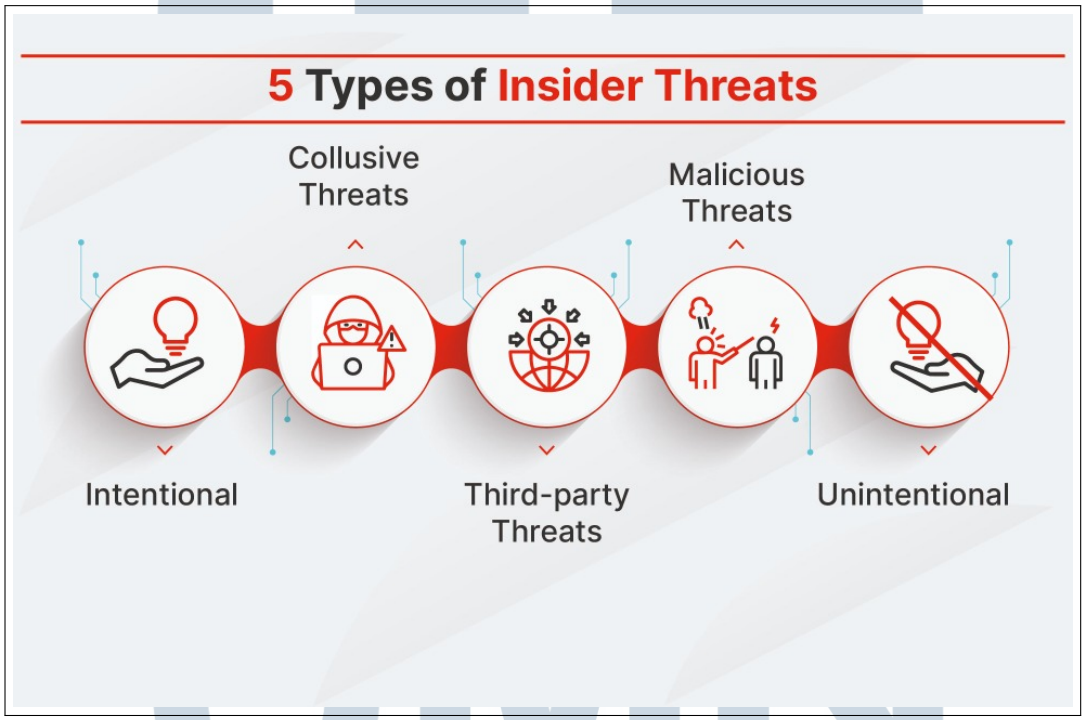


Gambar 2.4. Jenis-jenis Ancaman Malware.
Sumber: [24]

b. *Insider Threat*

Ancaman ini berasal dari individu dalam organisasi yang memiliki hak akses penuh terhadap sistem dan sumber daya, baik dilakukan secara sengaja

maupun tidak disengaja. Ancaman ini dapat berupa penyalahgunaan hak akses, kebocoran data, maupun kelalaian pengguna yang membuka celah kerentanan. Kasus yang sering terjadi adalah tanpa disadari menginstal *malware* atau menekan link dari sumber tidak valid kebenarannya yang kemudian dimanfaatkan pihak yang tidak bertanggung jawab. *Insider threat* tidak selalu didorong niat jahat, namun kualitas sumber daya manusia karyawan yang lalai dan bahkan sengaja menyalahgunakan aksesnya untuk memperoleh keuntungan pribadi atau sebagai bentuk ketidakpuasan dan balas dendam terhadap organisasi.

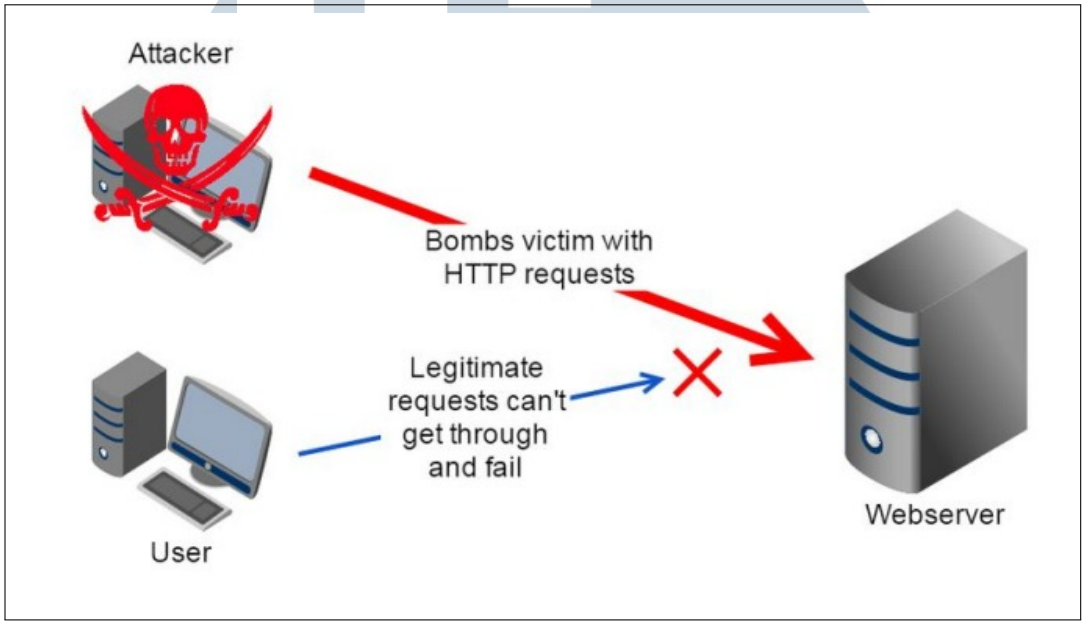


Gambar 2.5. Jenis-jenis Insider Threat.
Sumber: Fortinet.com

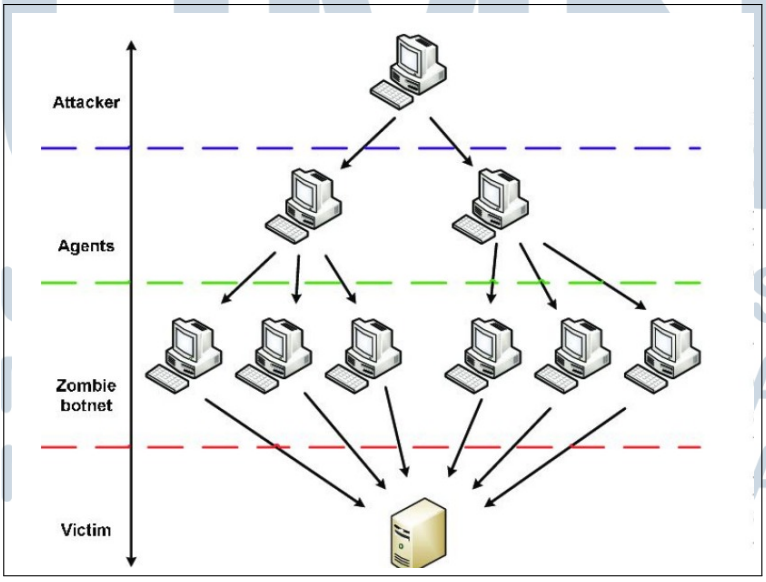
c. *Denial of Service* (DoS) dan *Distributed Denial of Services* (DDoS)

Denial of Service (DoS) merupakan serangan siber yang bertujuan untuk mengganggu ketersediaan (*availability*) sistem, layanan, atau jaringan dengan membanjiri target menggunakan permintaan dalam jumlah besar atau memanfaatkan celah keamanan, sehingga sumber daya seperti CPU, memori, dan *bandwidth* mengalami kelebihan beban dan layanan tidak dapat diakses oleh pengguna yang sah. Serangan DoS umumnya dilakukan dari satu mesin penyerang, dengan alamat IP yang dapat disamarkan menggunakan

proxy. Sementara itu, *Distributed Denial of Service* (DDoS) merupakan pengembangan dari DoS, di mana serangan dilakukan secara terdistribusi dari banyak mesin yang telah terinfeksi dan dikendalikan oleh penyerang, yang dikenal sebagai *botnet*, sehingga memiliki dampak yang lebih besar dan lebih sulit untuk dideteksi maupun dimitigasi.



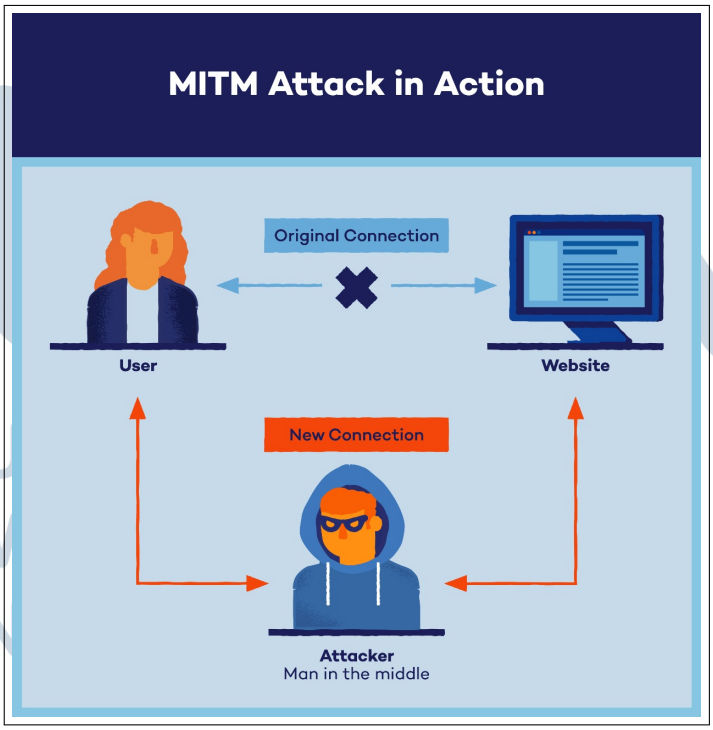
Gambar 2.6. Serangan DoS dilakukan ”sendirian”.
Sumber: Course Net



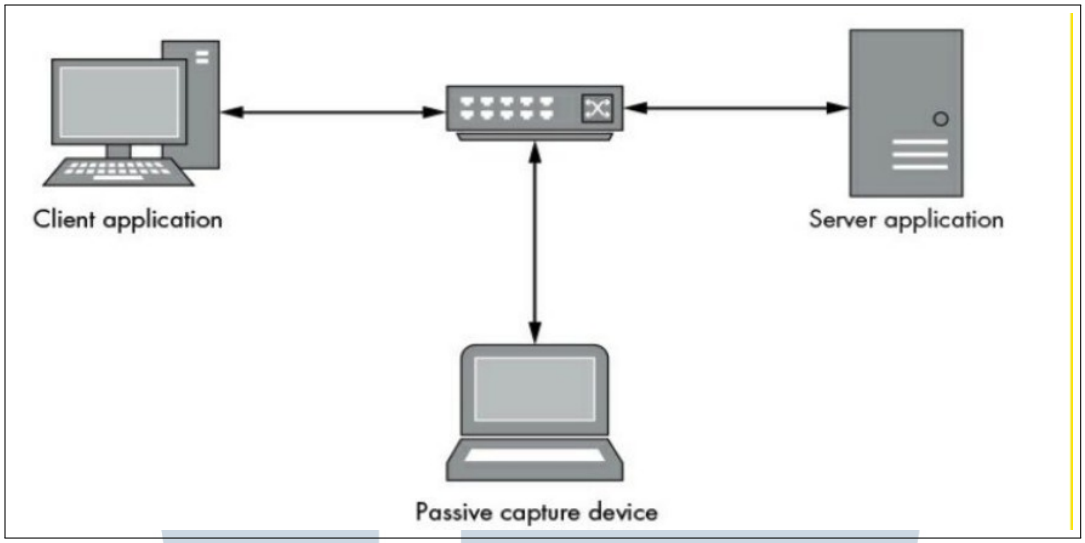
Gambar 2.7. Serangan DDoS dilakukan oleh “grup”.
Sumber: Course Net

d. *Man-in-the-Middle* (MITM) *Threat*

Man-in-the-Middle (MitM) *threat* merupakan ancaman keamanan siber yang berkaitan erat dengan teknik pengendusan (*sniffing*) lalu lintas data, baik secara pasif maupun aktif. Pengendusan pasif dilakukan dengan menganalisis paket data yang sedang ditransmisikan tanpa mengganggu komunikasi, misalnya menggunakan aplikasi packet *analyzer* seperti *Wireshark*. Sebaliknya, pengendusan aktif dilakukan dengan menempatkan penyerang secara langsung di tengah jalur komunikasi menggunakan aplikasi proxy yang bertindak sebagai perantara koneksi antara klien dan server, sehingga penyerang dapat memantau, memodifikasi, bahkan mengeksploitasi data yang ditransmisikan. Teknik pengendusan aktif inilah yang dilakukan dengan skenario *Man-in-the-Middle*, karena penyerang secara aktif memposisikan dirinya sebagai "perantara" komunikasi, sering kali melalui metode seperti *ARP spoofing*, untuk memperoleh informasi sensitif seperti kredensial autentikasi dan data sesi pengguna. Ancaman MitM ini menjadi serius karena dapat berlangsung tanpa disadari oleh pengguna dan secara langsung mengancam kerahasiaan serta integritas data yang dipertukarkan dalam komunikasi jaringan.



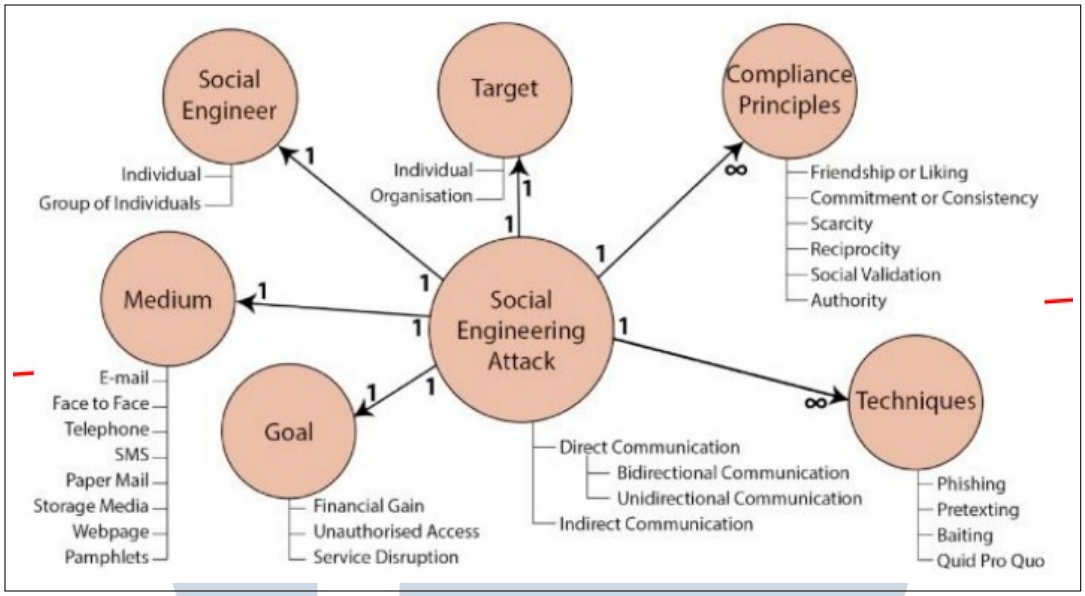
Gambar 2.8. Ancaman MITM.
Sumber: https://www.pandasecurity.com/



Gambar 2.9. Passive Capture Device.
Sumber: Course Net

e. *Social Engineering Threat*

Social Engineering atau Rekayasa Sosial merupakan ancaman keamanan siber yang memanfaatkan kombinasi faktor psikologis dan interaksi sosial untuk memanipulasi individu agar secara sukarela memberikan informasi penting atau melakukan tindakan tertentu yang menguntungkan penyerang. Rekayasa sosial tidak bergantung pada kelemahan teknis sistem, melainkan mengeksploitasi kepercayaan, emosi, dan perilaku manusia sebagai titik lemah utama dalam keamanan informasi. Ancaman ini umumnya dilakukan melalui berbagai media komunikasi seperti email, pesan singkat, telepon, atau situs web, dengan tujuan memperoleh informasi sensitif, akses tidak sah, atau menyebabkan gangguan layanan. *Social engineering threat* menjadi sangat berbahaya karena sering kali sulit dideteksi oleh pengguna dan dapat melewati mekanisme keamanan teknis, sehingga berperan sebagai faktor awal dalam berbagai serangan siber, termasuk *phishing* dan pencurian kredensial.



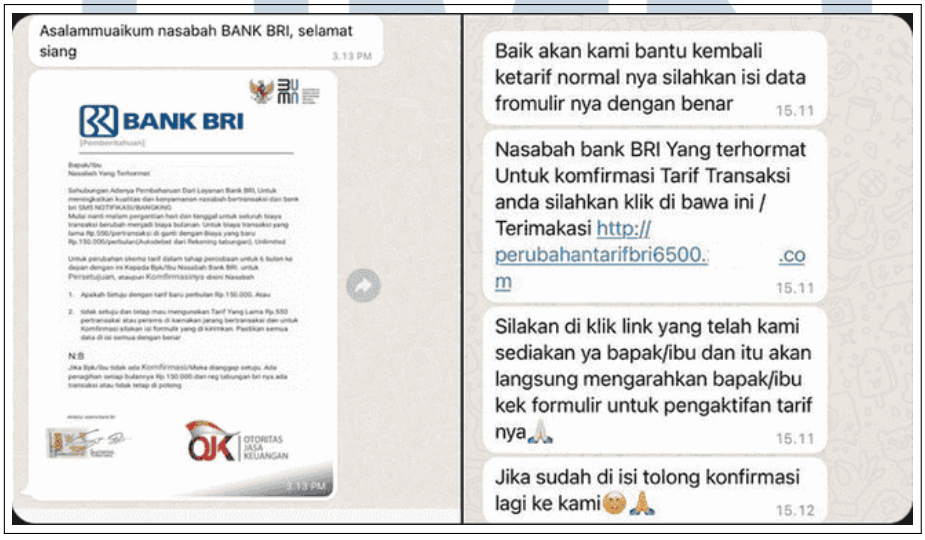
Gambar 2.10. Gambar Detail dari Rekayasa Sosial atau *Social Engineering*.
Sumber: Course Net

2.6 Phishing

Istilah *phishing* secara etimologis merupakan variasi dari kata bahasa Inggris ”*fishing*”. Metafora ini merujuk pada aktivitasnya, yaitu ”memancing” informasi berharga dari target secara daring. Awal mula kemunculan phishing dikenalkan oleh seorang *spammer* dan *hacker* bernama Khan C. Smith pada tahun 1990-an yang bertujuan untuk mendapatkan *username* dan *password* dari akun pengguna *Bank American Online* (AOL). Menurut ahli teknologi informasi Dendy Eka Puspawadi, *phishing* adalah modus penipuan yang mengelabui target menggunakan pesan palsu, seperti email, untuk memancing mereka masuk ke situs web jebakan demi mencuri akses akun korban [25]. Secara definitif, *phishing* merupakan salah satu metode rekayasa sosial dengan aksi penipuan siber yang paling sering digunakan untuk memperoleh informasi identitas pribadi secara ilegal, seperti kata sandi dan informasi kartu kredit [26]. Pelaku kejahatan siber (penipu) atau yang biasa dipanggil sebagai *phiser* menyebarkan email palsu yang mengarahkan penerimanya ke sebuah situs web tiruan. Situs ini sengaja dirancang untuk menjadi replika dari situs yang sah, dengan tujuan akhir mengelabui pengguna agar mereka memasukkan data-data rahasia miliknya [27]. Serangan ini juga dapat bertujuan memanipulasi korban untuk mengunduh perangkat lunak berbahaya (*malware*). Akibatnya, korban berpotensi mengalami kerugian ganda yang mencakup kehilangan data pribadi dan kerugian finansial. Ancaman ini tidak hanya menargetkan individu,

tetapi juga menyasar kelompok atau organisasi. Dalam praktiknya, pelaku *phishing* memanfaatkan kelalaian sumber daya manusia sebagai sebuah celah kerentanan.

Mekanisme kerja phishing dilakukan dengan cara pelaku (*phisher*) berpura-pura sebagai pihak resmi seperti bank, penyedia layanan telekomunikasi, atau perusahaan internet, kemudian menghubungi korban melalui *email*, media sosial, telepon, maupun pesan singkat (SMS dan WhatsApp). Pelaku sering kali menyertakan file atau undangan digital yang tampak sah, seperti dokumen dengan format PDF atau APK, link URL yang mengarahkan ke situs berbahaya, yang ketika dibuka dapat memicu peringatan keamanan atau notifikasi yang telah dimodifikasi, perangkat korban bisa terinfeksi dan data pribadinya berisiko dicuri. Pengguna harus selalu waspada dalam menggunakan berbagai jenis aplikasi komunikasi atau sosial media, terutama WhatsApp yang kini menjadi salah satu sasaran utama *phishing* di Indonesia. Jumlah pengguna internet di Indonesia yang sangat banyak dengan jumlah mencapai 112 juta pengguna, dengan jumlah pengguna WhatsApp terbanyak menjadi sasaran serangan *phishing* dengan memanfaatkan berbagai kelemahan pengguna, berupa minim literasi akan keamanan jaringan komputer serta beragan jenis tindak kejahatan teknologi digital. Oleh karena itu, pengguna harus selalu mengkoreksi dan memastikan kembali keakuratan pesan yang diterimanya apakah situs yang diterima merupakan situs resmi atau bukan, serta tidak mudah percaya terhadap kiriman file berupa PDF atau APK yang tidak jelas benarannya [28]. Berikut adalah beberapa jenis serangan phishing yang sering dilakukan:

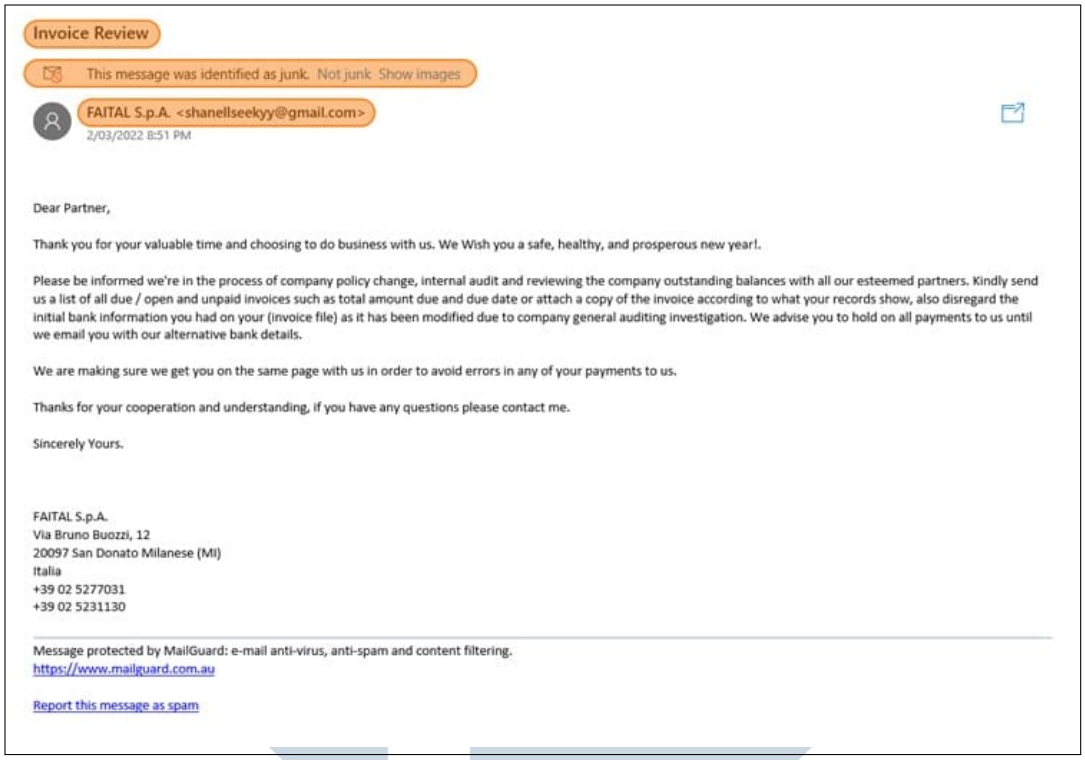


Gambar 2.11. Contoh Penipuan WhatsApp.
Sumber: https://www.hostinger.com/id/tutorial/phising-adalah

a. *Email Phishing*

Jenis serangan phishing ini biasanya dilakukan menggunakan *email* palsu untuk mengelabui korbannya dengan *email* yang dikirim secara acak kepada calon korban. Pelaku akan membuat *email* dengan format yang menyerupai *email* asli dan resmi dari perusahaan besar, sehingga korban dapat percaya untuk melakukan aksi menekan *link* atau fitur yang dibuat. Mitigasi Email Phishing ini dapat dilakukan dengan selalu mengecek keabsahan suatu domain *email* sebelum menekan *link* yang dicantumkan menggunakan *tools* yang ada seperti pada *OSINT Framework* yang menyediakan berbagai *tools* pengecekan keabsahan domain *email*. Kemudian kita dapat mengecek *header* dan pola pesan pada *email* yang masuk untuk mengetahui informasi penting suatu domain.

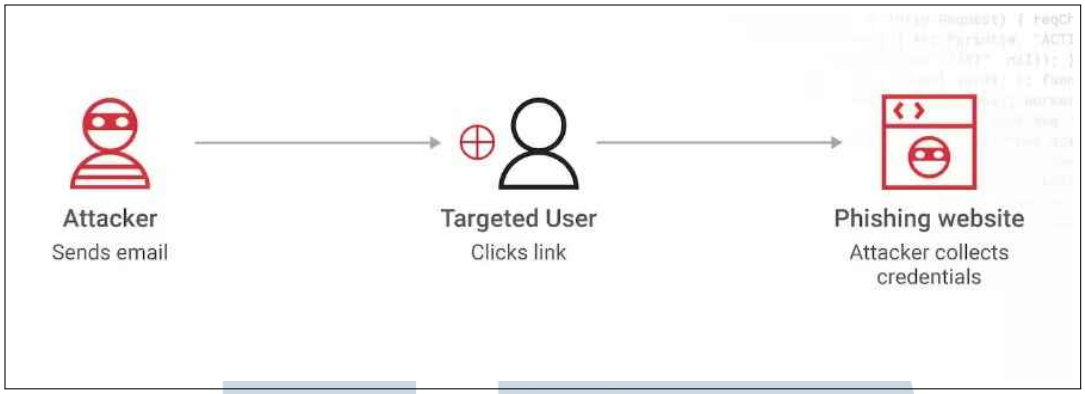
Email phishing dapat diidentifikasi melalui beberapa karakteristik umum yang mencurigakan. Salah satu ciri utamanya adalah penggunaan alamat pengirim yang tidak asli, yang sering kali sengaja dibuat mirip dengan domain resmi namun mengandung perbedaan kecil, seperti webbhosting.com dengan penambahan kata b yang meniru alamat sah webhosting.com. Selain itu, email ini umumnya menggunakan nada mendesak atau ancaman, seperti ”Akun Anda Diblokir!”, untuk menciptakan kepanikan dan memaksa penerima bertindak cepat. Tujuannya adalah untuk memancing korban agar memberikan informasi sensitif, seperti kata sandi atau nomor kartu kredit, yang tidak akan pernah diminta oleh perusahaan resmi. Ciri lainnya adalah keberadaan tautan (*link*) yang mencurigakan, dan sering kali, *email* tersebut berisi kesalahan tata bahasa atau typo (kesalahan ejaan) yang jelas.



Gambar 2.12. Contoh *Email Phishing*.
Sumber: https://www.rumahweb.com

b. *Spear Phishing*

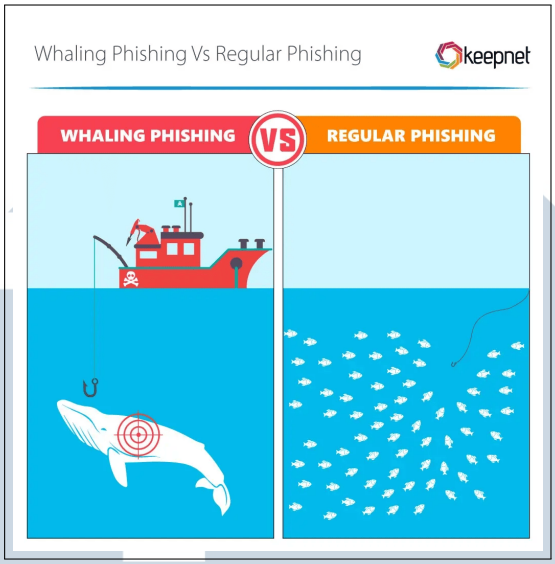
Jenis serangan yang menargetkan individu atau kelompok tertentu dengan pesan yang dipersonalisasikan untuk mengecoh korban agar mengungkapkan informasi sensitif atau menjalankan aksi yang merugikan suatu pihak. Pelaku umumnya melakukan pengumpulan informasi latar (*reconnaissance*) dalam jangka waktu tertentu untuk memahami profil, kebiasaan, dan konteks korban, sehingga pesan yang disampaikan terlihat relevan, meyakinkan, dan sulit dibedakan dari komunikasi yang sah [29]. Pendekatan yang terarah ini membuat tingkat keberhasilan serangan menjadi lebih tinggi dibandingkan dengan serangan *phishing* massal, karena memanfaatkan kepercayaan dan kedekatan konteks antara pelaku dan korban. Serangan semacam ini sering digunakan sebagai tahap awal untuk memperoleh kredensial, akses sistem, atau menyebarkan serangan lanjutan yang berdampak lebih luas.



Gambar 2.13. Proses *Spear Phishing*.
Sumber: https://www.akamai.com/glossary/what-is-spear-phishing

c. *Whaling*

Whaling merupakan Jenis serangan spear phishing yang sangat terarah dan berisiko tinggi, karena menargetkan individu dengan jabatan penting dalam suatu organisasi seperti eksekutif tingkat atas, manajer senior, atau pemilik perusahaan. Disebut ”*Whaling*” karena sasaran serangan ini diibaratkan sebagai ”ikan paus” dalam struktur organisasi, yakni pihak yang memiliki akses besar terhadap informasi strategis dan sistem internal perusahaan. Serangan ini memanfaatkan rekayasa sosial yang canggih dan riset personalisasi untuk menyamar sebagai pihak berwenang, sehingga pesan tampak meyakinkan dan mendorong korban membuka lampiran berbahaya, mengungkap kredensial, atau menyetujui transaksi strategis. Akibatnya, *whaling* berpotensi menyebabkan kebocoran data rahasia, manipulasi keputusan organisasi, serta kerugian finansial dan reputasi yang signifikan. Oleh karena itu pencegahan menuntut kombinasi pelatihan kesadaran keamanan, kebijakan kontrol akses yang ketat, dan mekanisme deteksi ancaman berlapis dalam organisasi [30].



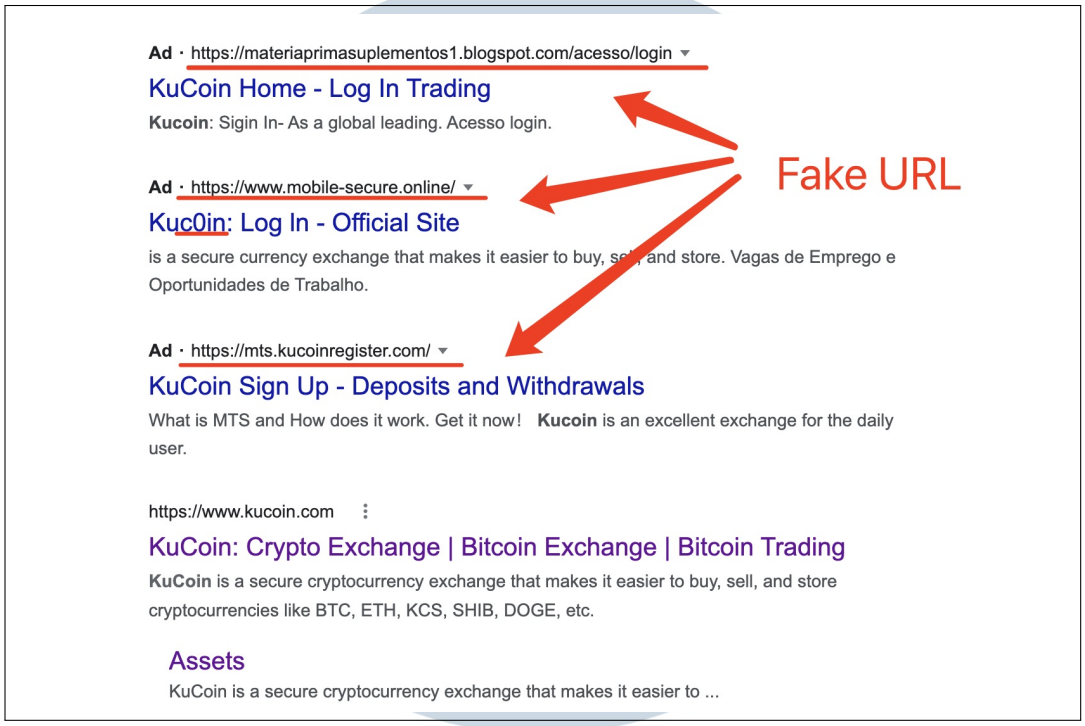
Gambar 2.14. Perbedaan *Phishing* Reguler dan *Phishing* Whaling.
Sumber: https://keepnetlabs.com/blog/what-is-a-whaling-attack

d. *Web Phishing*

Web phishing merupakan jenis serangan siber yang dilakukan dengan membangun situs web palsu yang dirancang menyerupai situs resmi suatu institusi atau organisasi dengan tujuan menipu korban agar memberikan informasi sensitif atau melakukan tindakan tertentu. Situs palsu tersebut dibuat semirip mungkin dengan situs aslinya, baik dari segi tampilan antarmuka, struktur halaman, maupun penggunaan nama domain yang menyerupai domain resmi. Dalam banyak kasus, pelaku juga memanfaatkan protokol HTTPS untuk memberikan kesan bahwa situs tersebut aman dan terpercaya, meskipun keberadaan HTTPS tidak menjamin keabsahan suatu situs web.

Serangan web phishing sering kali berhasil karena rendahnya tingkat ketelitian pengguna dalam memverifikasi keabsahan URL yang diakses, seperti perbedaan kecil pada penulisan domain, penggunaan subdomain mencurigakan, atau struktur URL yang tidak lazim. Kondisi ini membuat pengguna tidak menyadari bahwa mereka telah mengakses situs palsu. Selain berpotensi mencuri informasi sensitif seperti kredensial login dan data pribadi, situs web phishing juga dapat menjadi media penyebaran malware yang secara otomatis terunduh atau terpasang ke dalam perangkat korban saat halaman diakses. Oleh karena itu, web phishing menjadi salah satu

ancaman siber yang signifikan karena menggabungkan manipulasi visual, struktur URL yang menyesatkan, dan eksploitasi kelalaian pengguna.



Gambar 2.15. Contoh Kemiripan Antara Website *Phishing* dan Website Asli Suatu Instansi. Sumber: https://experteq.com

Serangan *phishing* memiliki dampak yang luas dan mendalam terhadap keamanan informasi dalam kelompok organisasi maupun individu. Sebagai bentuk serangan rekayasa sosial yang memanfaatkan kebohongan untuk mengelabui korban agar memberikan kredensial atau data sensitif, serta dapat mengambil alih secara penuh suatu perangkat dan mendapatkan informasi pribadi seperti IP Address, geolokasi, akses kamera, dan lain sebagainya dari jarak jauh. *Phishing* terbukti dapat merusak tiga pilar utama keamanan informasi, yakni kerahasiaan (*confidentiality*), integritas (*integrity*), dan ketersediaan (*availability*) dalam CIA Triad.

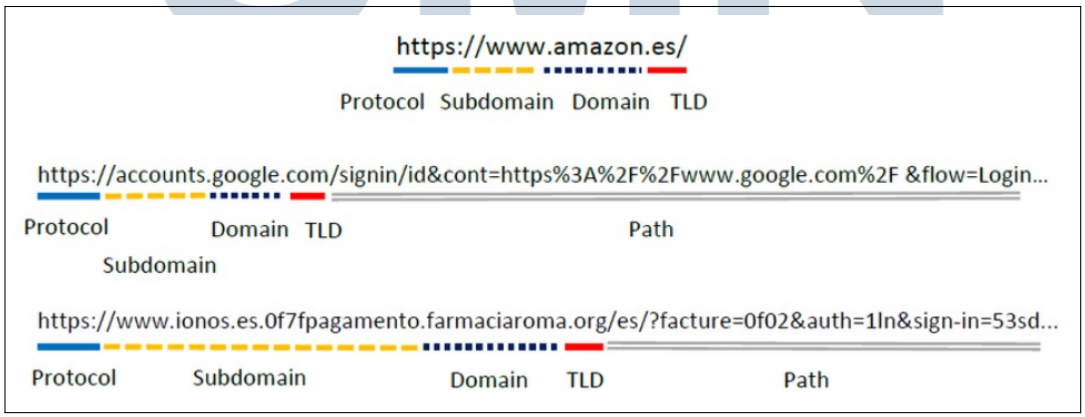
2.6.1 Struktur URL Phishing

URL merupakan rangkaian karakter yang berfungsi untuk menunjukkan lokasi atau alamat suatu sumber informasi di jaringan internet [31]. URL pada dasarnya berfungsi sebagai alamat yang menunjukkan lokasi suatu sumber informasi di internet. Melalui alamat ini, pengguna dapat mengakses berbagai

jenis konten, seperti dokumen, halaman web, gambar, video, atau file lainnya menggunakan protokol tertentu seperti HTTP maupun HTTPS. URL memiliki peran penting dalam membentuk struktur internet modern karena setiap halaman web saling terhubung satu sama lain melalui alamat unik tersebut. URL terdapat komponen penyusun, terdiri dari *Protocol*, *Domain Name*, *Port*, *Path*, *Query String* atau Parameter, dan *Fragment* sehingga anatomi URL berbentuk seperti https://sub.domain.com:433/path/page.php?user=123#section1. Dalam konteks kemana siber, URL sering menjadi sasaran utama pelaku phishing, yang sengaja membuat tiruan dari alamat asli untuk menipu pengguna agar mengunjungi situs berbahaya. Tujuan utamanya adalah mencuri informasi pribadi, seperti kata sandi atau data keuangan, dengan menyamarkan situs palsu yang menyerupai situs resmi dan meyakinkan. URL pada website *phishing* memiliki pola atau ciri yang mencurigakan yang sering digunakan untuk menipu pengguna, antara lain:

a. Panjang URL (*URL Length*)

   Pada umumnya, URL *phishing* memiliki panjang yang tidak wajar karena pelaku berusaha menambahkan berbagai elemen seperti *parameter*, *subdomain*, atau *path* tambahan untuk menyembunyikan alamat domain sebenarnya. Penambahan elemen-elemen tersebut membuat struktur URL menjadi lebih kompleks dan sulit dikenali oleh pengguna awam. Semakin panjang URL yang digunakan, semakin besar kemungkinan bahwa tautan tersebut mengandung manipulasi, pengalihan tersembunyi, atau struktur yang dirancang untuk mengelabui pengguna agar tidak menyadari tujuan asli dari URL yang diakses. Karakteristik ini menjadikan panjang URL sebagai salah satu indikator penting dalam proses deteksi phishing berbasis URL.



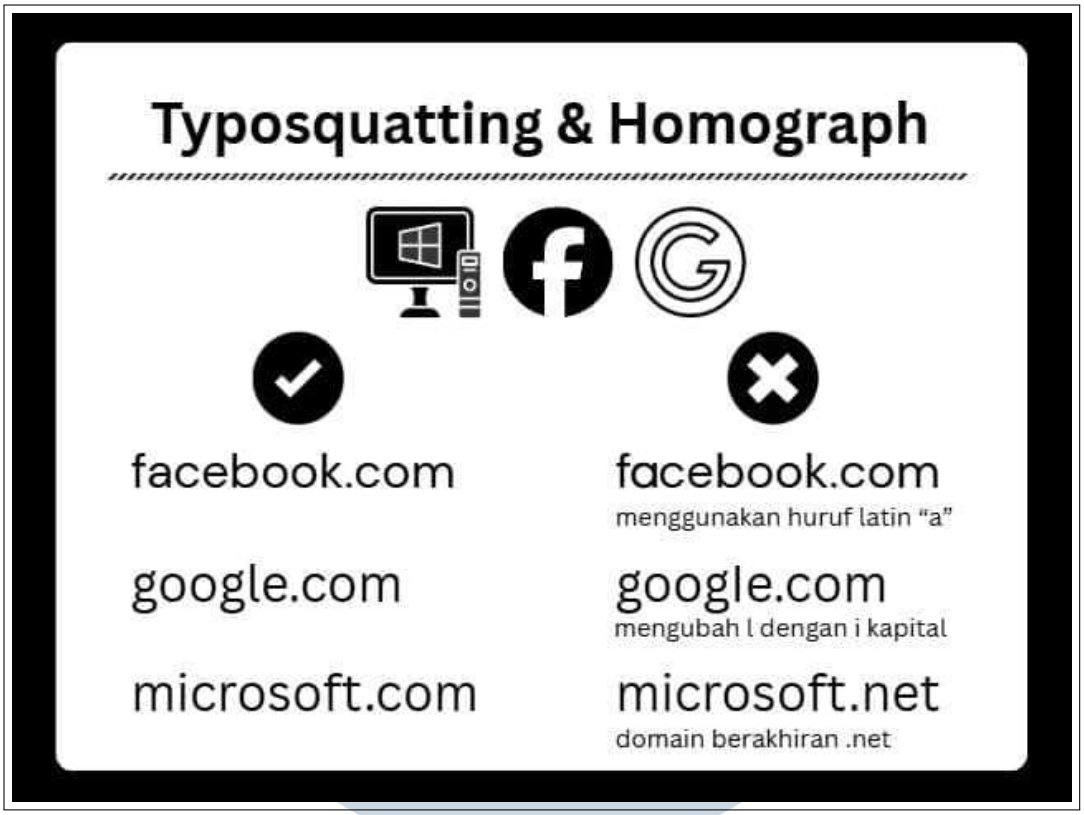
Gambar 2.16. Struktur Panjang URL.
Sumber: [32]

b. Jumlah dan Posisi Simbol Khusus

Pelaku phishing kerap menyisipkan berbagai simbol seperti @, -, _, %, =, dan ? secara berlebihan dalam struktur URL. Penggunaan simbol @, sering berfungsi sebagai alat redirect untuk mengarahkan pengunjung ke domain lain tanpa disadari. Teknik ini dilakukan melalui *JavaScript Redirect* atau *meta refresh* [33]. Hal ini dapat mengarahkan pengunjung ke halaman yang berisi malware yang kemudian dapat terinstall otomatis ke sistem operasi perangkat tanpa disadari.

c. Typosquatting dan Homograph

Typosquatting merupakan teknik dimana pelaku membuat domain palsu yang sangat menyerupai dengan domain asli dengan mengganti, menambah, atau menghapus karakter tertentu. Contohnya seperti go0gle.com, faceboook.com atau bahkan *homograph attack* dengan menggunakan bahasa lain yang memiliki abjad, alphabet, atau huruf yang mirip bacaannya seperti penulisan manusia sehari-hari yang dapat mengelabuhi pengunjung. Alphabet yang sering digunakan dengan kemiripan bacaan adalah alphabet rusia, dan latin. Teknik ini dapat hanya memasukkan satu alphabet yang mirip dengan penulisan karakter namun berbeda dendan pengetikan asli di komputer, seperti contoh facebook.com, dimana huruf kedua adalah *latin small letter alpha* dari huruf ”a”. Teknik ini memanfaatkan ketidak-telitian visual pengguna dalam membaca domain.



Gambar 2.17. Contoh Typosquatting dan Homograph Attack.
Sumber: https://cybersecurityasia.net/typosquatting-explained/

d. Pola Token dan Struktur URL yang Mencurigakan

Selain panjang dan simbol, URL phishing juga sering menunjukkan pola token dan struktur yang tidak lazim, seperti penggunaan kata-kata tertentu pada path atau parameter URL, struktur subdomain yang berlebihan, serta kombinasi karakter acak yang tidak merepresentasikan entitas resmi. Pola-pola ini dirancang untuk mengelabui pengguna dan menyamarkan tujuan sebenarnya dari URL. Analisis terhadap struktur dan token URL tersebut memungkinkan sistem deteksi mengenali anomali yang umum ditemukan pada serangan phishing berbasis web.

2.7 Machine Learning

Machine Learning atau pembelajaran mesin merupakan bidang dalam kecerdasan buatan yang berfokus pada pengembangan algoritma yang mampun mengenali pola dan mempelajari struktur dari data secara otomatis. Melalui proses pembelajaran ini, sistem komputer dapat membuat prediksi atau mengambil

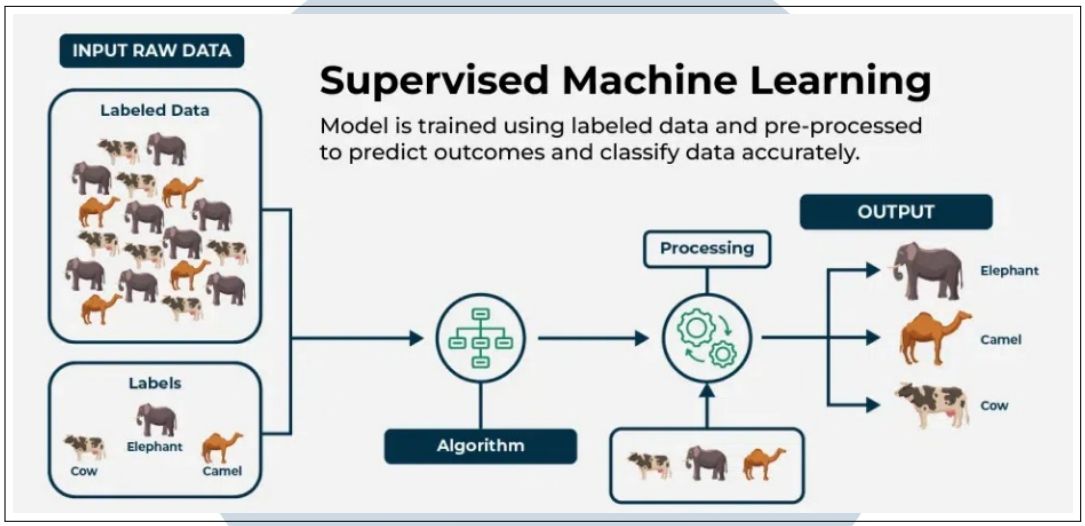
keputusan tanpa harus diprogram secara eksplisit untuk setiap kasus [34]. Pembelajaran mesin terdiri dari dua pendekatan, yaitu *supervised learning* (pembelajaran terawasi) dan *unsupervised learning* (pembelajaran tidak terawasi, yang digunakan untuk mengolah berbagai bentuk dan kompleksitas data [35]. Pendekatan tradisional seperti *blacklist* memiliki keterbatasan dalam mendeteksi situs *phishing* baru (*zero-day attack*), karena metode tersebut bergantung pada daftar domain yang telah diketahui sebelumnya. Safi et al. (2023) [36] menyatakan bahwa metode berbasis *blacklist* memiliki kelemahan dalam mendeteksi situs *phishing* baru, sehingga diperlukan pendekatan *machine learning* yang mampu mempelajari pola URL dan sertifikat secara adaptif.

2.7.1 Supervised Learning

Supervised Learning merupakan salah satu pendekatan dalam pembelajaran mesin yang memanfaatkan dataset berlabel (*input* dan *output* yang diketahui) untuk melatih algoritma agar mampu melakukan prediksi secara akurat. Tujuan dari pendekatan ini adalah untuk mempelajari fungsi pemetaan (*mapping function*) antara *input* dan *output* agar model dapat memprediksi hasil *output* untuk data baru berdasarkan pola yang telah dipelajari dari data latih (*training*). Dalam metode ini, proses pembelajaran berlangsung secara terarah dan eksplisit, dimana setiap sampel data memiliki label target yang digunakan untuk mengarahkan proses pelatihan agar kesalahan prediksi dapat diminimalkan. *Supervised learning* terbagi menjadi dua kategori utama, yaitu *classification* dan *regression*. Model *classification* digunakan untuk memprediksi kategori diskret, seperti mendeteksi apakah suatu email termasuk spam atau bukan. Sedangkan model *regression* berfungsi memprediksi nilai keberlanjutan, seperti memperkirakan harga rumah atau nilai saham.

Dalam penelitian ini, Penggunaan dataset phishing digunakan sebagai objek pembelajaran dapat melatih model agar dapat mengenali pola dan hubungan antara *input* dan *output*. Kelebihan pendekatan ini adalah tingkat akurasi yang tinggi dalam melakukan prediksi dan interpretasi hasil yang lebih mudah. Akan tetapi, kelemahannya terletak pada kebutuhan dataset berlabel dalam jumlah besar, serta potensi *overfitting* bila model terlalu menyesuaikan diri dengan data latih. Beberapa algoritma yang umum digunakan dalam *supervised learning* antara lain *Decision Tree*, *Support Vector Machine* (SVM), *Naive Bayes*, *Logistic Regrression*, dan *k-Nearest Neighbors* (k-NN). Dalam konteks keamana siber,

terutama pada deteksi phishing berbasis URL, *supervised learning* digunakan untuk mengklasifikasikan apakah suatu URL tergolong aman atau berbahaya berdasarkan pola fitur karakteristik URL yang telah dipelajari dari dataset sebelumnya [37].



Gambar 2.18. *Supervised Learning.*
Sumber: https://media.geeksforgeeks.org/

2.7.2 Ensemble Learning

Pembelajaran Gabungan *Ensemble Learning* merupakan paradigma dalam *machine learning* yang bertujuan untuk mencapai performa prediksi yang lebih optimal dengan menggabungkan keluaran dari beberapa model dasar (*base model*) yang relatif lemah (*weak learners*). Hipotesis utamanya adalah bahwa keputusan kolektif dari beragam model cenderung lebih akurat dan stabil dibandingkan dengan keputusan model tunggal. Model *Ensemble Learning* terbagi menjadi tiga kategori utama, yaitu *Bagging*, *Boosting*, dan *Stacking*. Persamaan dari ketiganya adalah semua menggunakan beberapa base model. Model Ensemble Learning terbagi menjadi tiga kategori, yaitu *Bagging*, *Boosting*, dan *Stacking*. Ketiga kategori ini memiliki kemiripan dalam penggunaan *base model* untuk menghasilkan prediksi yang lebih akurat dan stabil dibanding model tunggal. *Bagging* bekerja dengan melatih beberapa model pada subset data hasil bootstrap sampling dan menggabungkan melalui *voting* atau rata-rata, sehingga efektif dalam mengurangi varians model, seperti pada algoritma Random Forest. *Boosting* membangun model secara bertahap dengan memberikan perhatian lebih pada data yang salah diklasifikasikan oleh model sebelumnya untuk meningkatkan akurasi,

namun cenderung sensitif terhadap *noise*. Sementara itu, *Stacking* menggabungkan prediksi dari beberapa model menggunakan *meta-classifier* untuk menghasilkan keputusan akhir, meskipun memiliki kompleksitas yang lebih tinggi.

Dalam penelitian ini, algoritma Decision Tree (CART) berfungsi sebagai *base model* yang lemah. Untuk mengatasi keterbatasan akurasi Decision Tree tunggal, penelitian ini mengadopsi Random Forest (RF), yang merupakan contoh metode ensemble dengan menggunakan seluruh dataset. RF terdiri dari sekumpulan Decision Tree, di mana setiap pohon dilatih menggunakan sampel data individu dan setiap atribut dipecah secara acak. Proses ini, yang juga disebut sebagai *Random Subspace*, adalah kunci mengapa RF dapat meningkatkan hasil akurasi dan memiliki proses seleksi fitur internal untuk memilih fitur terbaik. Meskipun proses pembentukan pohon RF sama dengan CART, pada RF proses *pruning* (pemangkasan) tidak dilakukan, karena stabilitas model sudah didapatkan melalui gabungan prediksi banyak pohon. Selain meningkatkan akurasi, RF juga memiliki kelebihan berupa kemampuan menahan *outliers* dan dapat bekerja secara efektif pada *big data* dengan parameter yang kompleks [38].

2.7.3 Klasifikasi

Klasifikasi adalah proses pengelompokan benda atau informasi ke dalam kelompok secara sistematis berdasarkan persamaan dan perbedaan ciri-cirinya. Tujuannya adalah untuk memudahkan analisis, pemahaman, dan pengelolaan informasi secara terstruktur. Klasifikasi dalam machine learning dipahami sebagai tugas *supervised learning* yang memetakan representasi fitur suatu objek ke label diskret, sehingga mampu memprediksi kelas untuk data baru secara sistematis. Klasifikasi terdapat dua proses yang dilakukan yaitu dengan membangun model untuk disimpan sebagai memori dan hasilnya digunakan untuk melakukan pengklasifikasian prediksi objek pada data lain agar diketahui terdapat di kelas mana objek data yang disimpan [39].

2.7.4 Confusion Matrix

Confusion matrix merupakan salah satu metode evaluasi yang digunakan untuk menilai kinerja model klasifikasi dengan membandingkan hasil prediksi model terhadap label aktual pada data uji. *Confusion matrix* disajikan dalam bentuk tabel yang menunjukkan jumlah data yang diklasifikasikan dengan benar

maupun salah, sehingga memberikan gambaran yang lebih jelas mengenai performa sistem klasifikasi. Tabel ini terdiri dari empat komponen utama, yaitu *True Positive* (TP) dan *True Negative* (TN) yang menunjukkan prediksi benar, serta *False Positive* (FP) dan *False Negative* (FN) yang menunjukkan prediksi yang keliru. Melalui informasi tersebut, *confusion matrix* dapat digunakan sebagai dasar dalam menghitung berbagai metrik evaluasi seperti *accuracy*, *precision*, *recall*, dan *F1-score*, sehingga membantu peneliti dalam mengevaluasi kinerja model secara menyeluruh serta mengidentifikasi jenis kesalahan klasifikasi yang terjadi, khususnya dalam penelitian deteksi website *phishing* [40].

inpows.com

	Positive	**Negative**	
Positive	True Positive (TP)	False Negative (FN)	**Sensitivity** $\frac{TP}{(TP+FN)}$
Negative	False Positive (FP)	True Negative (TN)	**Specifity** $\frac{TN}{(TN+FP)}$
	Precision $\frac{TP}{(TP+FP)}$	**Negative Predictive Value** $\frac{TN}{(TN+FN)}$	**Accuracy** $\frac{TP+TN}{(TP+TN+FP+FN)}$

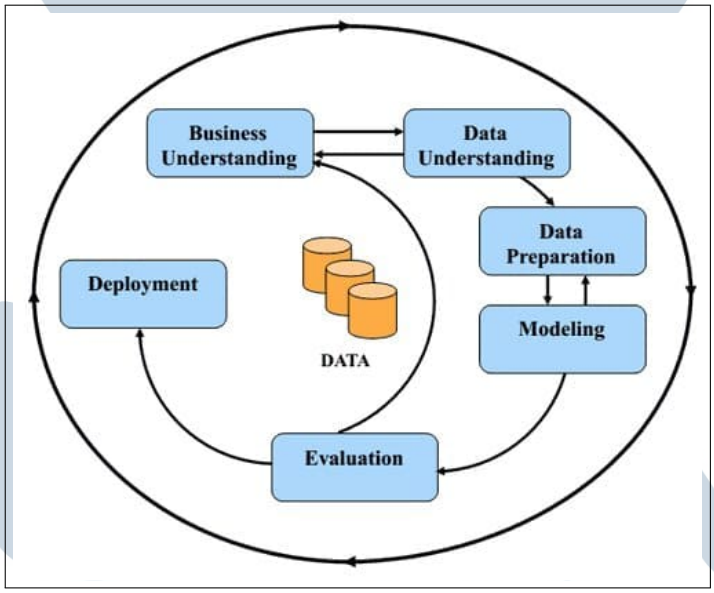
Gambar 2.19. Confusion Matrix.
Sumber: https://www.inpows.com/

2.8 Data Mining

Data Mining merupakan tahapan untuk menggali informasi serta pola bermakna dari kumpulan data berukuran besar. Proses ini meliputi pengumpulan data, ekstraksi informasi, analisis, serta pengolahan statistik untuk menemukan hubungan dan pola tersembunyi. Istilah data mining juga sering disebut sebagai *knowledge discovery*, *knowledge extraction*, *pattern analysis*, atau *information harvesting* karena tujuannya adalah mengubah data mentah menjadi pengetahuan yang dapat dimanfaatkan untuk pengambilan keputusan. Proses data mining terdapat dua metode yang bisa digunakan yaitu CRISP-DM (*Cross Industry Standard Process for Data Mining*) dan KDD (*Knowledge Discovery in Databases*).

2.8.1 CRISP-DM

Model *Cross Industry Standard Process for Data Mining* (CRISP-DM) merupakan kerangka metodologi standar yang paling banyak diadopsi oleh industri besar dalam pelaksanaan proyek data mining. CRISP-DM dirancang agar bersifat umum, dapat digunakan lintas industri, dan independen terhadap teknologi, sehingga menjadi satu kesatuan dalam proses data mining. Metode ini banyak digunakan dalam analisis sentimen yang terdiri dari enam fase, meliputi pemahaman bisnis (*business understanding*), pemahaman data (*data understanding*), pengolahan data (*data perparation)*, permodelan (*modelling*), evaluasi (*evaluation*), dan implementasi (*Deployment*). Sesuai dengan langkah dan fase yang telah ada diharapkan dapat menciptakan modelling yang paling sesuai dengan proses data mining melalui pemahaman data yang ada, sehingga meningkatkan efisiensi kinerja informasi yang dihasilkan.



Gambar 2.20. Diagram CRISP-DM.
Sumber: https://researchgate.net

Berikut adalah fase-fase dalam memproses data yang terdapat dalam kerangka kerja CRISP-DM [41], terdiri dari:

a. *Business Understanding*

Tahap *business understanding* merupakan langkah awal dalam proses data mining yang berfungsi untuk memahami dengan jelas permasalahan yang ingin diselesaikan. Sebelum melibatkan penggunaan data atau alat

analisis apapun, peneliti perlu menentukan terlebih dahulu tujuan utama yang ingin dicapai serta alasan mengapa tujuan tersebut penting untuk dicapai. Tahap *business understanding* terdiri dari empat kegiatan utama, yaitu mengidentifikasi tujuan bisnis, menilai situasi dan kondisi yang ada, menentukan tujuan penambangan atau analisis data, dan membuat rencana proyek.

b. *Data Understanding*

Tahap *data understanding* merupakan fase kedua setelah peneliti telah menentukan arah tujuan rencana penelitian. Pada tahap ini, peneliti mulai meninjau data yang akan digunakan untuk memastikan data tersebut relevan dan sesuai dengan kebutuhan penelitian. Proses ini mencakup kegiatan seperti mengumpulkan data, mendeskripsikan isi dan struktur data, dan eksplorasi awal seperti menemukan dan mengenali pola atau anomali, serta verifikasi data.

c. *Data Preprocessing*

Tahap *data preprocessing* merupakan fase yang paling banyak meyita waktu dalam proses *data mining*. Hal ini terjadi karena dilakukan proses pengolahan data mentah menjadi data yang siap digunakan untuk *modeling*. Data yang dikumpulkan pada umumnya masih mengandung *noise*, duplikasi, *Not a Number* (NaN), atau nilai tidak relevan, sehingga perlu dilakukan pembersihan dan transformasi. Hasil dari pembersihan tersebut yang kemudian dijadikan sebagai bahan untuk pembelajaran mesin.

d. *Modeling*

Tahap *modeling* merupakan proses ini dari proses *data mining*, dimana algoritma atau metod analisis diterapkan untuk membangun model yang dapat menggambarkan pola dari data yang telah disiapkan. Pada tahap ini dilakukan proses pemilihan teknik pemodelan yang sesuai berupa algoritma yang akan digunakan, merancang skenario pengujian, pelatihan dan pembangunan model, serta menilai hasil model.

e. *Evaluation*

Tahap *Evaluation* bertujuan untuk melakukan eksplorasi data dan menemukan pola sejauh mana model yang dibangun mampu memenuhi tujuan penelitian serta menghasilkan kinerja yang baik. Pada fase ini,

peneliti meninjau kembali hasil model untuk memastikan bahwa model tersebut bekerja dengan baik sesuai kebutuhan analisis. Evaluasi tidak hanya terhadap performa secara statistik, tetapi juga keseluruhan proses *data mining* yang digunakan untuk membangunnya. Pada tahap ini meliputi proses mengevaluasi hasil model, meninjau proses yang dilakukan, dan menentukan langkah selanjutnya yang akan dilakukan.

f. *Deployment*

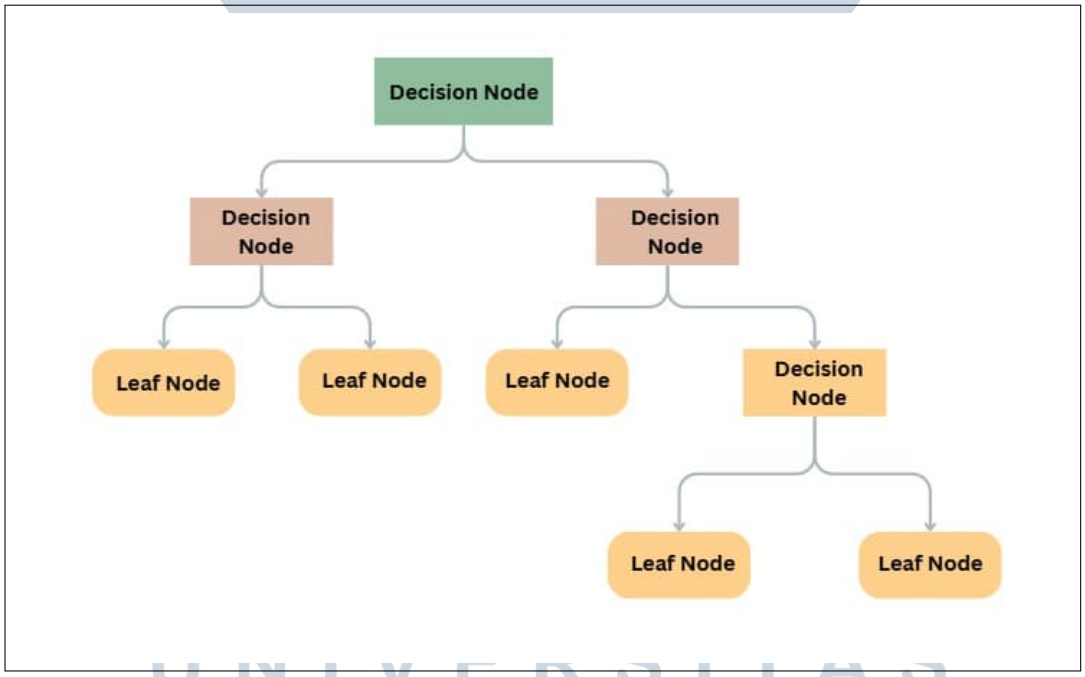
Tahap *Deployment* merupakan fase terakhir dalam proses kerja CRISP-DM, dimana model telah dievaluasi dan hasil dari proses *data mining* mulai dimanfaatkan secara nyata. Pada tahap ini, seluruh pengetahuan, pola, dan model decision tree yang telah dihasilkan sebelumnya diimplementasikan untuk mendukung pengambilan keputusan atau meningkatkan proses bisnis yang ada. Proses *deployment* juga meliputi dokumentasi model, pembuatan antarmuka sederhana, serta rencana pemantauan dan pemeliharaan, melaporkan hasil akhir, serta meninjau hasil akhir.

2.9 Algoritma

Algoritma merupakan serangkaian prosedur komputasi yang jelas dan terstruktur yang menerima satu atau beberapa nilai sebagai masukan (*input*) lalu menghasilkan satu atau beberapa nilai sebagai keluaran *(output)*. Dengan kata lain, ini adalah urutan langkah yang mengubah input dan output secara sistematis dan terstruktur berdasarkan metode yang digunakan untuk memecahkan masalah komputasi yang kompleks dengan efisien dan tepat sesuai kebutuhan pemecahan persoalan [42]. Dikutip dari *Microsoft Press Computer and Internet Dictionary (1998)*, algoritma dapat dipahami sebagai rangkaian operasi yang dirancang khusus untuk menyelesaikan masalah secara rasional dan terstruktur. Setiap langkah dieksekusi dengan logis dimana urutan yang jelas dan konsisten, tanpa lompatan atau ambiguitas, sehingga prosesnya dapat diikuti dan diverifikasi [43]. sebuah algoritma yang baik juga menekankan determinisme, ketuntasan (*finiteness*), dan ketepatan spesifikasi terhadap masalah yang diselesaikan. Contohnya dalam konteks deteksi *phishing*, masukan berupa sebuah URL (beserta fitur turunan lainnya seperti panjang URL, kebenaran protokol HTTPS, susunan subdomain, usia domain, entropi string, jumlah simbol,hingga pola penggunaan kata atau simbol mencurigakan).

2.9.1 Decision Tree

Algoritma *Decision Tree* (DT) merupakan teknik klasifikasi sederhana yang merepresentasikan proses pengambilan keputusan dalam bentuk struktur pohon, berupa node internal memuat kondisi pada fitur, cabang menggambarkan hasil pengujian, dan daun (*leaf*) berisi prediksi kelas. Model ini termasuk *supervised learning* karena dilatih menggunakan dataset berlabel dan melakukan pemisahan (*splitting*) data secara rekursif hingga kelompok pada node menjadi semakin homogen terhadap kelas target. Keunggulan decision tree antara lain yaitu interpretabilitas tinggi, waktu pelatihan relatif cepat, dan kemampuannya menangani data dengan ukuran besar dan fitur campuran (kategorikal dan numerik). Namun demikian, kelemahannya mencakup kecenderungan *overfitting* jika level pohon dilakukan hingga murni, sensitifitas terhadap perubahan data kecil, serta *bias* saat variabel memiliki banyak kategori [44].



Gambar 2.21. Decision Tree

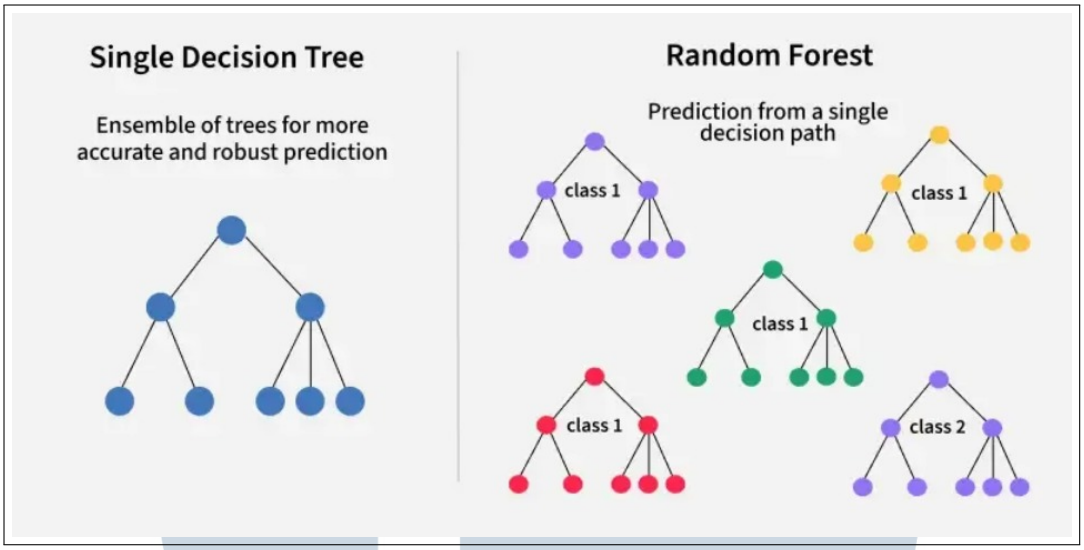
2.9.2 Classification and Regression Trees (CART)

Classification and Regression Trees (CART) merupakan salah satu algoritma pembelajaran mesin yang berpengaruh dalam menangani permasalahan klasifikasi maupun regresi melalui mekanisme pemisahan biner secara rekursif. Berbeda

dengan keluarga algoritma *Decision Tree* klasik seperti ID3 dan C4.5 yang mengandalkan *entropy* dan *information gain*, CART menggunakan ukuran ketidakmurnian berbasis indeks *Gini*. Penggunaan indeks *Gini* ini memberikan keuntungan efisiensi komputasi karena tidak memerlukan perhitungan logaritma yang kompleks seperti pada *entropy*, serta didukung oleh mekanisme *minimal cost-complexity pruning* untuk generalisasi model. Meskipun teknik *pruning* dirancang untuk mengendalikan kompleksitas model dan mengurangi risiko *overfitting*, model Decision Tree tunggal yang dibangun menggunakan CART cenderung memiliki variansi yang tinggi. Karakteristik inilah yang kemudian melatarbelakangi pengembangan metode ensemble, seperti Random Forest, guna meningkatkan stabilitas dan performa prediksi dengan menggabungkan banyak pohon CART. Dalam praktik modern, CART banyak diimplementasikan melalui pustaka `scikit-learn` dengan menggunakan kriteria `gini` sebagai standar pemilihan atribut dan dukungan parameter `ccp_alpha` untuk optimasi struktur pohon, sebagaimana dijelaskan dalam penelitian sebelumnya [45]. Indeks Gini digunakan untuk memilih fitur di setiap simpul internal dari pohon keputusan [46]

2.9.3 Random Forest (RF)

Random Forest (RF) merupakan algoritma ensemble learning yang dikembangkan dari metode *Decision Tree*. RF terdiri dari sekumpulan *Decision Tree*, di mana kumpulan pohon keputusan ini digunakan untuk mengklasifikasikan data ke dalam suatu kelas. Setiap *Decision Tree* dalam RF telah dilakukan pelatihan menggunakan sampel individu, dan pemisahan atribut pada setiap pohon dipilih antara subset atribut yang bersifat acak [47]. RF merupakan salah satu metode yang dapat meningkatkan hasil akurasi dalam membangkitkan atribut untuk setiap node yang dilakukan secara acak. Pohon keputusan dibuat dengan menentukan node akar dan berakhir dengan beberapa node daun untuk mendapatkan hasil akhir. Membentuk pohon keputusan pada metode RF memiliki proses yang sama dengan *Classification and Regression Tree* (CART), namun pada RF tidak dilakukan pruning (pemangkasan). Pada *Random Forest*, prediksi akhir diperoleh melalui mekanisme *majority voting*, di mana kelas yang paling banyak dipilih oleh seluruh pohon keputusan akan ditetapkan sebagai hasil klasifikasi. Pendekatan ini memungkinkan model menghasilkan keputusan yang lebih stabil dan mengurangi pengaruh kesalahan pada pohon individual.



Gambar 2.22. Pohon Keputusan Random Forest.
Sumber: https://www.geeksforgeeks.org/

2.10 Tools

Tools dalam konteks penelitian merupakan sekumpulan perangkat atau alat bantu yang digunakan untuk membantu peneliti melakukan proses pengolahan data, analisis, pembuatan model, maupun implementasi hasil penelitian (biasanya berupa *software* atau aplikasi).Penelitian ini memanfaatkan beberapa perangkat pendukung utama, yaitu *Python*, *Kaggle*, dan *Google Colab*.

2.10.1 Python

Python merupakan bahasa pemrograman tingkat tinggi yang dirancang untuk mendukung kemudahan pemahaman sintaks dan efisiensi penulisan kode. Diciptakan oleh Guido van Rossum dan dirilis pertama kali pada tahun 1991. Seiring berjalannya waktu, python telah berkembang menjadi salah satu bahasa pemrograman yang paling populer di dunia. Python bersifat multi-paradigma karena mendukung pemrograman berorientasi objek (OOP), imperatif, dan fungsional, sehingga fleksibel dalam berbagai kebutuhan pengembangan perangkat lunak dan analisis data. Salah satu keunggulannya terletak pada ketersediaan pustaka (*library*) yang sangat luas seperti NumPy, Pandas, dan Scikit-learn yang mendukung implementasi *machine learning* dan *data science*. NumPy membantu dalam komputasi numerik yang efisien, Pandas memudahkan pengolahan serta pembersihan data dalam bentuk tabel, sedangkan Scikit-learn menyediakan

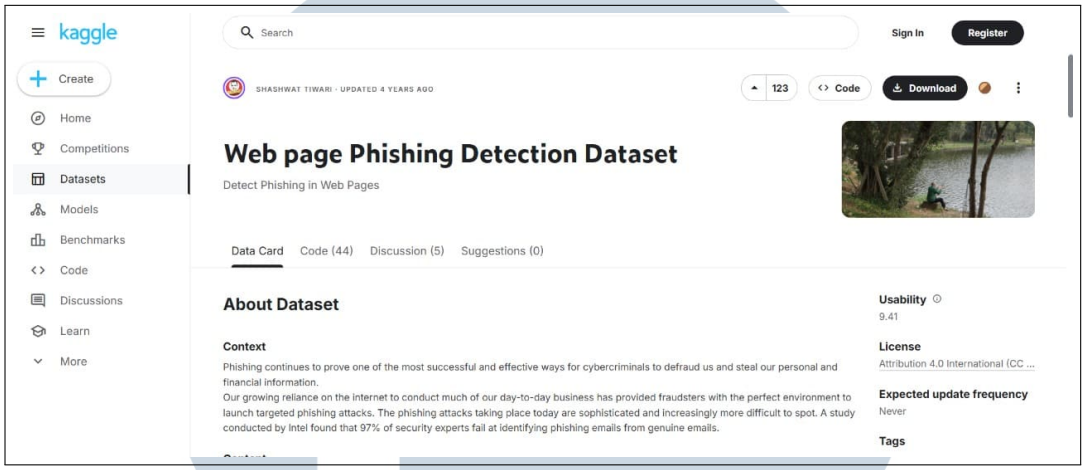
berbagai algoritma dan fungsi untuk membangun serta mengevaluasi model. Kombinasi ketiga *library* ini memungkinkan proses analisis dan pemodelan data dilakukan secara terstruktur, cepat, serta mudah untuk direproduksi. Selain itu, Komunitas pengguna *python* yang besar menjadikan bahasa ini terus berkembang dengan dukungan dokumentasi dan update berkelanjutan, menjadikannya ideal untuk penelitian berbasis komputasi modern [48].

Gambar 2.23. Logo Bahasa Pemrograman Python

2.10.2 Kaggle

Kaggle merupakan platform daring berbasis komunitas yang dirancang untuk mendukung kolaborasi dalam bidang *data science* dan *machine learning*. Platform ini memungkinkan peneliti dan praktisi untuk mengakses berbagai dataset publik, membangun model prediktif, serta berpartisipasi dalam kompetisi ilmiah yang menantang kemampuan analisis data secara praktis. Selain sebagai penyedia data, Kaggle juga menawarkan lingkungan pemrograman berbasis cloud yang disebut *Kaggle Notebook*, di mana pengguna dapat menulis, menjalankan, dan membagikan kode Python atau R tanpa perlu melakukan instalasi perangkat lunak tambahan di komputer lokal [49]. Ekosistem Kaggle memberikan peluang pembelajaran kolaboratif karena pengguna dapat mengamati, mengadaptasi, dan mengembangkan model dari notebook publik yang telah dibuat oleh anggota lain, sehingga mendorong pertukaran pengetahuan secara terbuka. Selain itu, fitur discussion forum dan leaderboard memperkuat motivasi belajar berbasis kompetisi yang efektif dalam meningkatkan keterampilan analisis data. Dengan demikian,

Kaggle tidak hanya berperan sebagai repositori dataset, tetapi juga sebagai sarana edukatif interaktif yang mendukung riset dan eksperimen *machine learning* secara global [50].



Gambar 2.24. Platform Dataset Kaggle.

2.10.3 Google Colab

Google Colaboratory atau Google Colab merupakan platform komputasi awan (*cloud-based notebook environment*) yang dikembangkan oleh Google untuk mendukung kegiatan penelitian dan pembelajaran di bidang *machine learning* dan analisis data. Colab berbasis teknologi *Jupyter Notebook* yang memungkinkan pengguna menulis, mengeksekusi, dan mendokumentasikan kode Python secara interaktif melalui antarmuka berbasis web. Platform ini menyediakan sumber daya komputasi gratis berupa CPU, GPU, dan TPU, sehingga sangat mendukung eksperimen *deep learning* dan analisis numerik tanpa memerlukan perangkat keras lokal yang kuat dan mudah dibagikan [51].

Menurut Carneiro dkk. [51], Google Colab menunjukkan performa komputasi yang sebanding dengan perangkat keras dedikasi seperti *Linux* server yang dilengkapi GPU Tesla K40, khususnya pada akselerasi pelatihan *Convolutional Neural Network* (CNN). Studi tersebut menegaskan bahwa Colab layak digunakan sebagai sarana penelitian berbasis GPU karena mendukung pustaka populer seperti *TensorFlow*, *Keras*, *Matplotlib*, dan *OpenCV*, serta terintegrasi langsung dengan *Google Drive* untuk kemudahan kolaborasi.

Sementara itu, penelitian Dewi dkk. [52] menunjukkan bahwa Google Colab efektif digunakan untuk menyelesaikan persoalan matematis dan numerik

seperti Metode Bagi Dua (*bisection method*) pada persamaan nonlinear dengan memanfaatkan Teorema Nilai Antara, yang menyatakan bahwa jika sebuah fungsi kontinu memiliki nilai berlawanan tanda pada dua titik interval (a,b), maka ada setidaknya satu akar di antara kedua titik tersebut. Dibandingkan dengan *Google Spreadsheet*, Colab memberikan efisiensi waktu yang lebih tinggi, fleksibilitas dalam pemrograman, serta kemampuan otomatisasi iterasi yang tidak terbatas. Implementasi *Python* di Colab memungkinkan visualisasi hasil secara dinamis menggunakan pustaka *Matplotlib* dan meminimalkan kesalahan manual dalam perhitungan.

Secara keseluruhan, Google Colab dapat dipandang sebagai lingkungan komputasi kolaboratif yang efisien dan terstandarisasi untuk eksperimen *machine learning* dan analisis data. Platform ini tidak hanya meningkatkan produktivitas peneliti dan mahasiswa, tetapi juga memperluas akses terhadap teknologi GPU bagi institusi pendidikan yang memiliki keterbatasan sumber daya perangkat keras.

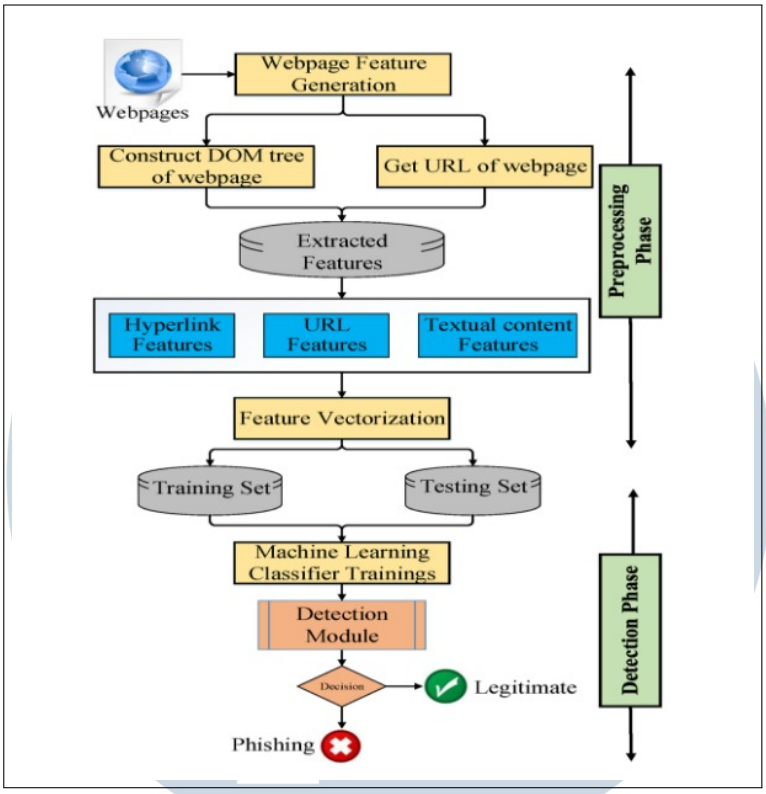
Gambar 2.25. Logo Platform Google Colab.

2.11 Penelitian Terdahulu

Penelitian terdahulu berperan sebagai dasar pembanding dan landasan pengembangan penelitian ini. Kajian terhadap penelitian sebelumnya membantu peneliti memahami metode, pemilihan dataset, perancangan fitur, serta proses pre-processing dan pemodelan dalam deteksi URL phishing. Studi literatur menunjukkan bahwa sebagian besar penelitian berfokus pada evaluasi performa algoritma dan tingkat akurasi, namun masih memiliki keterbatasan dalam pemanfaatan karakteristik URL dan pengujian pada dataset yang beragam. Oleh karena itu, kajian ini digunakan sebagai dasar dalam merumuskan celah penelitian dan kontribusi penelitian berbasis pembelajaran mesin.

2.11.1 Aljofey et al. (2022)

Penelitian berjudul "*An Effective Detection Approach for Phishing Websites Using URL and HTML Features*" yang dilakukan oleh Aljofey et al. (2022) mengusulkan pendekatan deteksi phishing berbasis *machine learning* dengan mengombinasikan fitur URL dan HTML untuk mengatasi keterbatasan metode *blacklist/whitelist* dalam mendeteksi serangan phishing baru (*zero-day attacks*). Pendekatan ini bersifat *client-side* dan tidak bergantung pada layanan pihak ketiga, dengan memanfaatkan urutan karakter URL, fitur hubungan *hyperlink*, serta fitur TF-IDF pada level karakter yang diekstraksi dari *plaintext* dan bagian *noisy HTML* halaman web. Evaluasi dilakukan menggunakan dataset yang dibangun sendiri dengan total 60.252 halaman web, yang terdiri dari 32.972 halaman *benign* (*legitimate*) dan 27.280 halaman *phishing*. Berbagai algoritma klasifikasi diuji, dan hasil eksperimen menunjukkan bahwa XGBoost memberikan kinerja terbaik ketika seluruh fitur digabungkan, dengan akurasi 96,76% dan *false positive rate* 1,39% pada dataset internal, serta akurasi 98,48% dengan *false positive rate* 2,09% pada dataset pembanding. Meskipun menghasilkan performa yang tinggi, pendekatan ini memiliki keterbatasan karena memerlukan akses ke *source code* HTML halaman web dan melibatkan fitur berbasis konten yang meningkatkan kompleksitas pemrosesan, sehingga membuka peluang penelitian lanjutan yang lebih ringan, seperti deteksi phishing berbasis karakteristik URL saja [53].

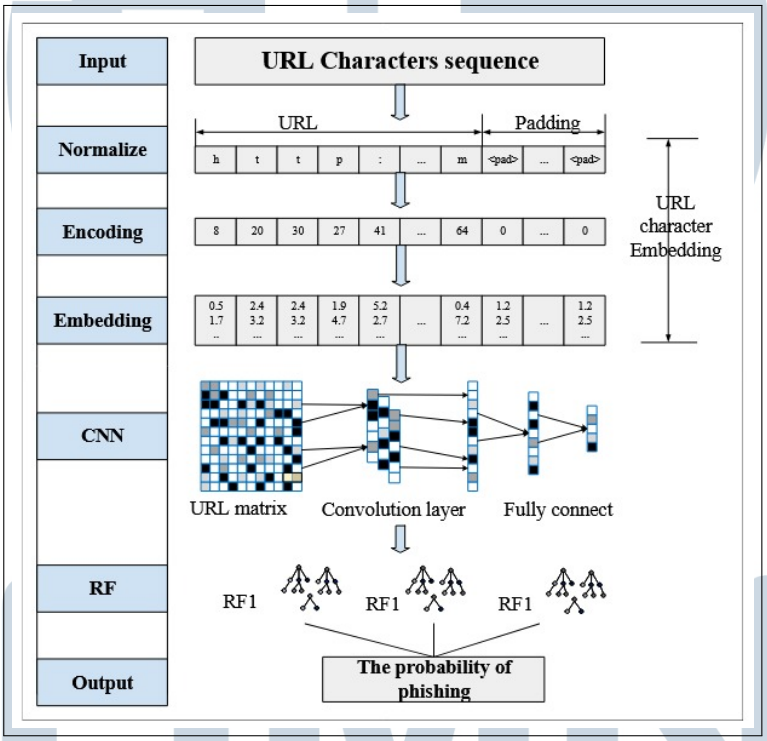


Gambar 2.26. Framework Penelitian Aljofey et al. (2022).

2.11.2 Yang et al. (2021)

Penelitian oleh Yang et al. (2021) melalui paper berjudul "*Phishing Website Detection Based on Deep Convolutional Neural Network and Random Forest Ensemble Learning*" yang dipublikasikan pada jurnal Sensors (MDPI) mengusulkan metode deteksi *phishing* berbasis URL dengan mengombinasikan *character-level Convolutional Neural Network (CNN)* dan *Random Forest* (RF). Penelitian ini berfokus pada analisis karakter URL tanpa memanfaatkan konten web maupun layanan pihak ketiga, sehingga proses deteksi dapat dilakukan secara lebih cepat dan efisien. URL direpresentasikan dalam bentuk matriks berdimensi tetap menggunakan teknik *character embedding*, kemudian CNN digunakan untuk mengekstraksi fitur URL pada beberapa tingkat (*multi-level*). Selanjutnya, fitur-fitur tersebut diklasifikasikan menggunakan beberapa model Random Forest dengan pendekatan *winner-take-all* untuk menentukan hasil akhir klasifikasi. Dataset yang digunakan terdiri dari 47.210 URL, dengan 22.491 URL *phishing* yang diperoleh dari PhishTank dan 24.719 URL *legitimate* dari Alexa, serta satu dataset pembanding yang berjumlah 83.857 URL. Hasil pengujian menunjukkan bahwa

metode yang diusulkan mampu mencapai akurasi sebesar 99,35% pada dataset utama dan 99,26% pada dataset benchmark, yang lebih tinggi dibandingkan beberapa metode baseline berbasis machine learning dan deep learning. Meskipun demikian, penelitian ini masih memiliki keterbatasan, seperti waktu pelatihan model yang relatif lama serta ketidakmampuan sistem dalam mendeteksi URL yang tidak aktif dan phishing yang tidak meniru domain tertentu, sehingga membuka peluang pengembangan lebih lanjut dengan penambahan fitur lain, misalnya fitur konten web, kode HTML, atau ikon situs [54].



Gambar 2.27. Framework Penelitian Yang et al. (2021).

2.11.3 Ryan Putra Ramadhan dan Teti Desyani (2023)

Ryan Putra Ramadhan dan Teti Desyani (2023) dalam jurnal penelitian berjudul "*Implementasi Algoritma J48 Untuk Identifikasi Website Phishing*" yang dipublikasikan melalui Biner: Jurnal Ilmu Komputer, Teknik dan Multimedia. Pembahasan mengenai penerapan algoritma J48 yaitu implementasi terhadap algoritma *C4.5 Decision Tree* untuk mengklasifikasi website *phishing* menggunakan aplikasi web WEKA. Penelitian ini menggunakan pendekatan berbasis URL dengan menganalisis karakteristik domain website seperti penggunaan https, jumlah *slash*, keberadaan alamat IP, jumlah titik (dot), panjang

URL, serta penggunaan simbol "@". Dataset yang digunakan diperoleh dari Kaggle berjumlah 50 data website, yang kemudian diolah melalui perhitungan *entropy* dan *information gain* (*gini*) untuk membangun pohon keputusan J48. Hasil penelitian menunjukkan bahwa algoritma J48 mampu menghasilkan model klasifikasi yang mudah dipahami dan efektif dalam mendeteksi website phishing, dimana atribut slash memiliki nilai gain tertinggi dan menjadi node akar pada *decision tree*. Namun demikian, penelitian ini memiliki beberapa keterbatasan, mulai dari ukuran dataset yang relatif kecil, penggunaan satu algoritma klasifikasi saja tanpa dilakukan pembanding dengan metode lain, serta jumlah fitur yang masih terbatas. Evaluasi kinerja model belum dilakukan secara komprehensif pada dataset yang lebih besar dan beragam. Oleh karena itu, masih terdapat beberapa celah penelitian untuk mengembangkan sistem deteksi *phishing* berbasis URL dengan memanfaatkan dataset yang lebih besar, penambahan fitur, serta perbandingan dengan algoritma klasifikasi lain guna meningkatkan akurasi dan generalisasi model [13].

2.11.4 Fatiha et al. (2024)

Fatiha et al. (2024) dalam penelitian berjudul "*Optimisasi Sistem Deteksi Phishing Berbasis Web Menggunakan Algoritma Decision Tree*" yang dipublikasikan pada Jurnal Ilmiah IT CIDA mengembangkan sistem deteksi phishing berbasis web dengan memanfaatkan algoritma Decision Tree dan metode pengembangan perangkat lunak *Rapid Application Development* (RAD). Penelitian ini menggunakan dataset publik dari Kaggle yang terdiri dari 11.055 data dengan 26 atribut, mancakup informasi URL, konten halaman web, *metadata*, serta informasi sertifikat keamanan. Tahapan penelitian meliputi proses *pre-processing* data seperti penghapusan duplikasi, penanganan *missing value*, dan normalisasi, kemudian data dilakukan pembagian 80% data *training* dan 20% data *testing* serta pengevaluasian menggunakan *k-fold cross-validation*. Hasil pengujian menunjukkan bahwa model *Decision Tree* mampu mencapai akurasi sebesar 95,07%, yang menunjukkan efektivitas algoritma tersebut dalam mengidentifikasi website *phishing*. Selain itu, penerapan metode RAD dinilai mampu mendukung pengembangan sistem yang cepat dan efisien. Namun demikian, penelitian ini masih memiliki beberapa keterbatasan, antara lain penggunaan satu algoritma klasifikasi saja tanpa perbandingan dengan metode lain, evaluasi performa yang masih berfokus pada akurasi tanpa pembahasan metrik lain seperti *precision*, *recall*, dan *F1-score* secara mendalam, serta ketergantungan pada fitur berbasis

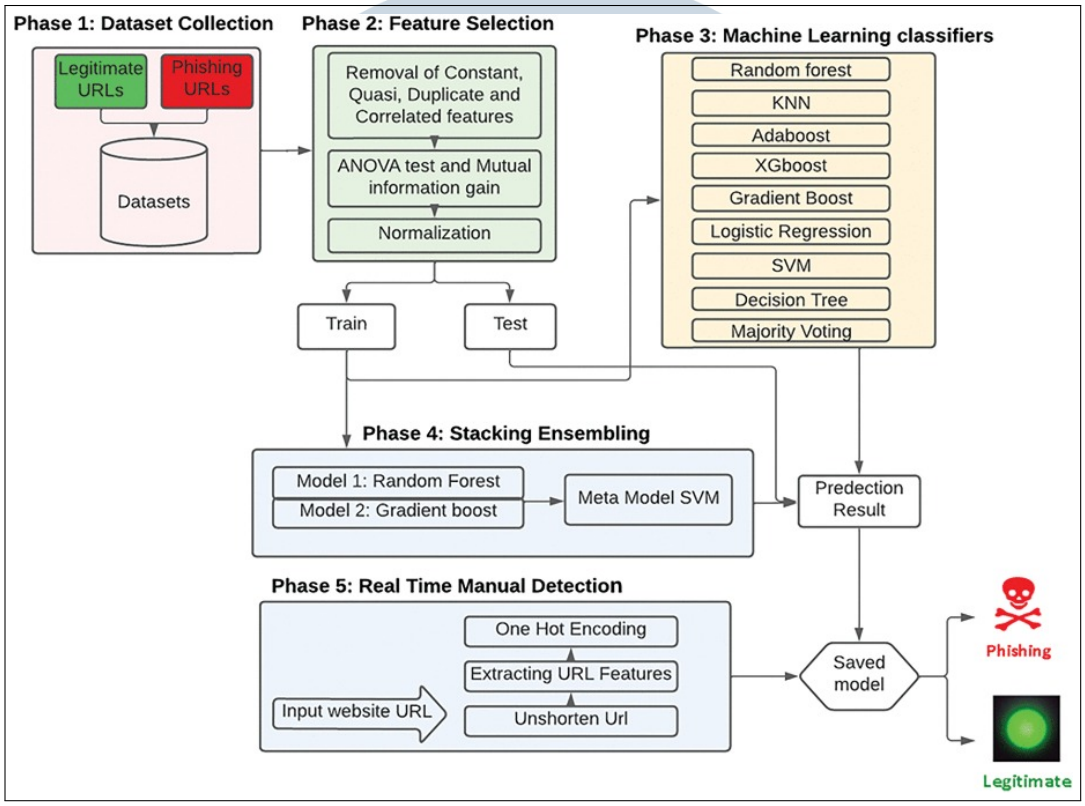
web yang dapat meningkatkan kompleksitas sistem. Oleh karena itu, terdapat celah penelitian untuk mengembangkan deteksi phishing dengan perbandingan atau kombinasi algoritma lain seperti *Random Forest*, penggunaan fitur yang lebih selektif, serta pendekatan yang lebih ringan dan efisien untuk meningkatkan akurasi dan generalisasi model [55].

Gambar 2.28. Framework Penelitian Fatiha et al. (2024).

2.11.5 Bari et al. (2025)

Penelitian oleh Bari et al. (2025) yang berjudul "*A Filter-Based Feature Selection Framework to Detect Phishing URLs Using Stacking Ensemble Machine Learning*" dipublikasikan pada jurnal *Engineering Application of Artificial Intelligence (ScienceDirect/Elsevier)* membahas pengembangan sistem deteksi *phishing* berbasis *machine learning* dengan fokus utama pada penurunan *false positive rate* yang masih menjadi permasalahan pada banyak metode deteksi *phishing* sebelumnya. Penelitian ini menekankan pentingnya proses *feature selection*, karena penggunaan fitur yang tidak relevan atau berlebihan dapat menurunkan kinerja model. Oleh karena itu, penulis menerapkan beberapa tahapan seleksi fitur, seperti penghapusan fitur konstan, duplikat, dan berkorelasi, serta penggunaan metode ANOVA dan Mutual Information untuk memilih fitur yang paling berpengaruh. Pengujian dilakukan menggunakan tiga dataset publik dengan karakteristik yang berbeda, yaitu dataset berukuran kecil hingga besar dengan jumlah fitur yang bervariasi. Setelah proses seleksi fitur, beberapa algoritma klasifikasi seperti *Random Forest*, *Decision Tree*, *SVM*, dan *XGBoost* diuji dan dikombinasikan menggunakan pendekatan *ensemble* seperti *stacking* dan *majority voting*. Hasil eksperimen menunjukkan bahwa model *ensemble* yang diusulkan mampu mencapai tingkat akurasi yang tinggi dengan nilai *false positive rate* yang lebih rendah dibandingkan metode sebelumnya. Meskipun demikian, pendekatan ini masih memiliki keterbatasan karena melibatkan jumlah fitur dan struktur model yang cukup kompleks, sehingga berpotensi meningkatkan beban komputasi. Hal ini membuka peluang penelitian lanjutan untuk mengembangkan metode deteksi

phishing berbasis URL yang lebih sederhana dan efisien, namun tetap mampu mempertahankan kinerja deteksi yang baik [56].

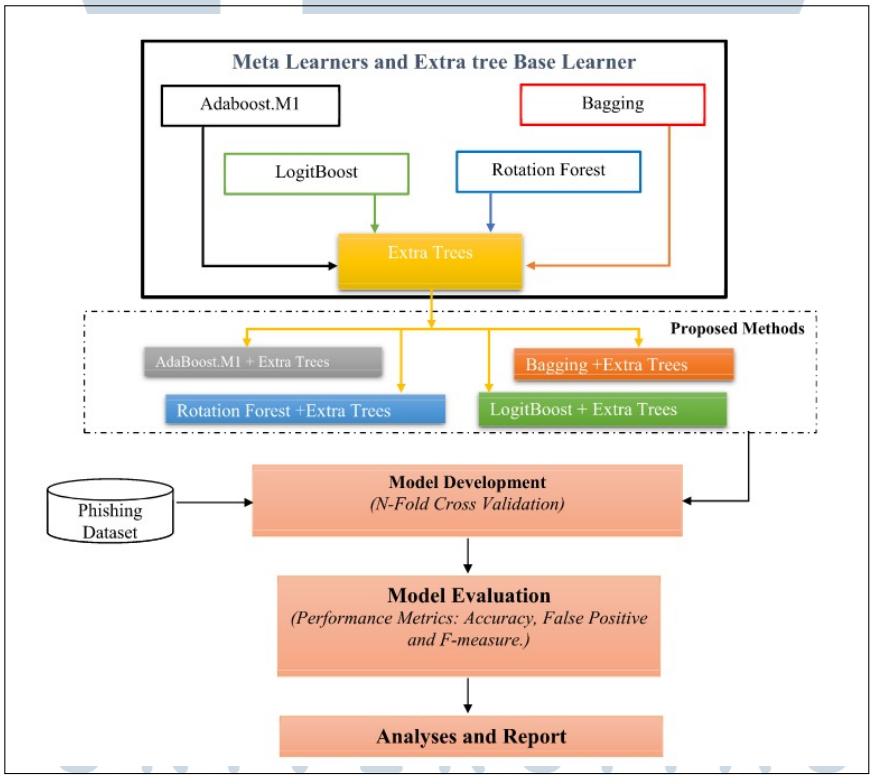


Gambar 2.29. Framework Penelitian Bari et al. (2025).

2.11.6 Alsariera et al. (2020)

Alsariera et al. (2020) dalam penelitian berjudul "*AI Meta-Learners and Extra-Trees Algorithm for the Detection of Phishing Websites*" yang dipublikasikan pada jurnal IEEE Access mengusulkan pendekatan deteksi phishing berbasis *artificial intelligence* dengan memanfaatkan meta-learner dan algoritma Extra Trees sebagai *base classifier*. Penelitian ini dilatarbelakangi oleh kelemahan metode klasifikasi tunggal yang umumnya memiliki tingkat false positive dan false negative yang cukup tinggi dalam mendeteksi website phishing. Oleh karena itu, penulis mengembangkan empat model meta-learner, yaitu AdaBoost–Extra Tree (ABET), Bagging–Extra Tree (BET), Rotation Forest–Extra Tree (RoFBET), dan LogitBoost–Extra Tree (LBET) untuk meningkatkan kinerja deteksi phishing. Dataset yang digunakan berasal dari UCI Phishing Website Dataset dengan jumlah 11.055 data dan 30 fitur, yang mencakup fitur berbasis URL, domain, serta

karakteristik HTML dan JavaScript. Proses evaluasi dilakukan menggunakan cross validation dengan metrik akurasi, *false positive rate*, *false negative rate*, dan *F-measure*. Hasil pengujian menunjukkan bahwa seluruh model meta-learner yang diusulkan mampu mencapai akurasi di atas 97% dengan tingkat kesalahan yang relatif rendah, di mana salah satu model menghasilkan *false positive rate* sekitar 0,018. Meskipun menunjukkan performa yang baik, penelitian ini masih memiliki keterbatasan karena pengujian dilakukan pada dataset statis dan belum diterapkan pada lingkungan *real-time*. Selain itu, penggunaan beberapa model meta-learner membuat kompleksitas sistem menjadi lebih tinggi. Oleh karena itu, masih terdapat peluang penelitian lanjutan untuk mengembangkan metode deteksi phishing yang lebih sederhana dan efisien, namun tetap memiliki tingkat akurasi dan keandalan yang tinggi.



Gambar 2.30. Framework Penelitian Alsariera et al. (2020).

Tabel 2.1. Ringkasan Penelitian Terdahulu Deteksi Website Phishing

No	Peneliti (Tahun)	Metode / Algoritma	Fitur yang Digunakan	Dataset	Hasil dan Keterbatasan
1	Aljofey et al. (2022)	XGBoost dan beberapa algoritma ML	URL character sequence, hyperlink features, TF–IDF karakter dari HTML dan plaintext	60.252 halaman (32.972 benign; 27.280 phishing) + benchmark	Akurasi 96,76% (FPR 1,39%) dan 98,48% (FPR 2,09%); perlu akses HTML dan kompleksitas pemrosesan meningkat.
2	Yang et al. (2021)	Character-level CNN + Random Forest (ensemble)	Karakter URL berbasis character embedding dan fitur multi-level CNN	47.210 URL (22.491 phishing; 24.719 legitimate) + benchmark 83.857 URL	Akurasi 99,35% dan 99,26%; pelatihan relatif lama dan tidak mengecek URL tidak aktif.
3	Ramadhan & Desyani (2023)	J48 (C4.5 Decision Tree)	Karakteristik URL (https, slash, IP address, panjang URL, simbol khusus)	Kaggle: 50 data website	Model mudah dipahami; dataset kecil dan belum ada perbandingan algoritma serta evaluasi pada data yang lebih beragam.
4	Fatiha et al. (2024)	Decision Tree + RAD	URL, konten web, metadata, dan sertifikat keamanan	Kaggle: 11.055 data (26 atribut)	Akurasi 95,07%; evaluasi dominan akurasi dan belum membahas metrik lain secara mendalam serta belum dibandingkan dengan model lain.
5	Bari et al. (2025)	Stacking ensemble (RF, SVM, dll.) + feature selection	Seleksi fitur (ANOVA, Mutual Information) dan filter-based cleaning	Tiga dataset publik (ukuran bervariasi)	FPR lebih rendah dan akurasi tinggi; model kompleks sehingga beban komputasi meningkat.
6	Alsariera et al. (2020)	AI meta-learners berbasis Extra Trees	Fitur URL, domain, HTML, dan JavaScript	UCI: 11.055 data (30 fitur)	Akurasi >97% dan FPR rendah; kompleksitas tinggi dan belum diuji pada skenario real-time.