BAB II TINJAUAN PUSTAKA

2.1 Penelitian Sebelumnya

Sebagai pendukung atas penelitian ini dilaksanakan pengkajian dalam penelitian sebelumnya agar memberi pemahaman yang mendalam terkait *Penetration Testing* yang mampu dipergunakan. Berikut ini penelitian-penelitian terdahulu mengenai "*Penetration Testing*".

1. ***Web Application Penetration Testing Using SQL Injection Attack***[6]

   Penelitian ini membahas mengenai pentingnya keamanan aplikasi web dan metode untuk melakukan pengujian keamanan guna mencegah serangan siber. Metode yang dipergunakan pada penelitian ini yakni metode pengujian penetrasi mempergunakan pendekatan kotak hitam (*black-box*) untuk menguji keamanan aplikasi web berdasarkan serangan terbanyak yang terdaftar pada *Open Web Application Security Project* (OWASP), khususnya serangan *SQL Injection*. Temuan dari penelitian ini memperlihatkan jika sebesar 80% dari website yang diuji mengalami kerentanan terhadap serangan *SQL Injection*.

   Penelitian ini menyimpulkan bahwa serangan *SQL Injection* masih merupakan ancaman serius bagi aplikasi web. Serangan ini memungkinkan penyerang untuk mengakses informasi rahasia seperti *database* melalui kerentanan pada aplikasi web. Hal ini memberikan kesempatan bagi penyerang untuk secara langsung mengambil informasi dari *database*. Dengan demikian, penelitian ini menekankan betapa pentingnya mengamankan aplikasi web dan melakukan pengujian keamanan secara teratur untuk mencegah serangan *SQL Injection* dan melindungi data rahasia yang tersimpan dalam *database* aplikasi web[6].

2. ***Vulnerability Testing Analysis of XYZ Regional Government Site Using PTES***[7]

   Penelitian ini mengkaji tentang analisis pengujian kerentanan pada situs pemerintah daerah XYZ mempergunakan PTES (*Penetration Testing Execution Standard*). Fokus penelitian ini adalah pentingnya menjaga keamanan sistem informasi dalam era kemajuan teknologi, terutama dalam

konteks akses web. Situs web pemerintah daerah XYZ dipilih sebagai objek penelitian untuk memperkuat keamanan layanan pemerintahan terintegrasi di wilayah tersebut[7].

Persoalan yang dikaji pada penelitian ini yakni seperti apa mengevaluasi tingkat risiko pada situs layanan pemerintah daerah XYZ dan mengidentifikasi kerentanan yang ada di dalamnya. Penulis menggunakan PTES sebagai standar pelaksanaan pengujian kerentanan dan menggunakan berbagai alat untuk melakukan serangan pada situs web tersebut. Hasil penelitian menunjukkan bahwa situs pemda XYZ memiliki beberapa kerentanan, seperti *Cross-Site Scripting* (XSS), HTML *form* tanpa perlindungan CSRF, *Clickjacking*, *insecure cookies*, *password guessing attack*, dan lain sebagainya[7].

3. ***Security Analysis On Websites Using The Information System Assessment Framework (ISSAF) And Open Web Application Security Version 4 (Owaspv4) Using The Penetration Testing Method***[8]

   Dalam penelitian ini, permasalahan yang dibahas adalah bagaimana melakukan analisis keamanan pada website menggunakan kerangka kerja ISSAF dan OWASPv4 dengan metode pengujian penetrasi. Penelitian ini bertujuan untuk mengidentifikasi kerentanan keamanan pada website dan memberikan rekomendasi untuk meningkatkan keamanannya. Hasil penelitian menunjukkan bahwa penggunaan kerangka kerja ISSAF dan OWASPv4 dengan metode pengujian penetrasi dapat membantu mengidentifikasi kerentanan keamanan pada website[8].

   Dalam penelitian ini, beberapa kerentanan keamanan berhasil diidentifikasi dan dijelaskan secara rinci. Selain itu, penelitian ini juga memberikan rekomendasi untuk meningkatkan keamanan website, seperti menginstal patch keamanan, mengubah kata sandi secara berkala, dan memperbarui perangkat lunak. Dengan demikian, penelitian ini dapat membantu meningkatkan kesadaran tentang pentingnya keamanan website dan memberikan panduan praktis untuk meningkatkan keamanannya[8].

4. **Analisis Keamanan Sistem Informasi Akademik Menggunakan *Open Web Application Security Project Framework***[9]

   Penelitian ini bertujuan untuk menganalisis keamanan sistem informasi akademik menggunakan *Open Web Application Security Project* (OWASP) *Framework* dan menemukan kerentanan pada sistem. Hasil penelitian menunjukkan bahwa terdapat 12 kerentanan pada sistem informasi akademik yang dianalisis menggunakan OWASP *Framework*. Dari 12 kerentanan tersebut, empat kerentanan berada pada *level medium*, enam kerentanan pada *level low*, dan dua kerentanan pada *level informational*[9].

5. **Implementasi *Penetration Testing Execution Standard* Untuk Uji Penetrasi Pada Layanan *Single Sign-On***[4]

   Penelitian ini berfokus pada layanan *Single Sign-On* (SSO) di sebuah instansi dan membahas permasalahan keamanan sistem serta pentingnya melakukan uji penetrasi untuk mengidentifikasi kerentanan dalam sistem tersebut. Metode yang dipergunakan pada penelitian ini yakni *Penetration Testing Execution Standard* (PTES). Hasil penelitian ini menunjukkan berhasilnya uji penetrasi yang dilakukan pada sistem informasi kampus yang menggunakan layanan SSO[4].

   Dalam analisis kerentanan, ditemukan beberapa kerentanan yang perlu diperhatikan, terutama yang berkaitan dengan kebocoran informasi sensitif. Selain itu, hasil pengumpulan informasi menyediakan pemahaman yang lebih baik terkait organisasi dan layanan yang diuji. Kesimpulan dari penelitian ini adalah pentingnya melaksanakan uji penetrasi guna mendeteksi kerentanan dalam sistem. Dengan melakukan uji penetrasi, kerentanan dapat diidentifikasi dan langkah-langkah pengelolaan risiko dapat dirancang untuk mencegah kebocoran informasi[4].

6. ***Autonomous Security Analysis and Penetration Testing***[10]

   Penelitian ini membahas tentang bagaimana mengembangkan sebuah *framework* yang dapat melakukan analisis keamanan dan pengujian penetrasi secara otomatis pada jaringan yang besar. Masalah yang dihadapi adalah teknik pengujian penetrasi saat ini menggunakan kombinasi alat

pemindaian otomatis dan eksploitasi manual dari masalah keamanan untuk mengidentifikasi ancaman yang mungkin terjadi pada jaringan. Namun, solusi ini tidak efektif pada jaringan yang besar dan kompleks[10].

Hasil dari penelitian ini adalah pengembangan sebuah *framework* yang disebut *Autonomous Security Analysis and Penetration Testing (ASAP)* yang dapat melakukan analisis keamanan dan pengujian penetrasi secara otomatis pada jaringan yang besar dan kompleks. *Framework* ini menggunakan algoritma *reinforcement* learning berbasis *Deep-Q Network* (DQN) untuk mengidentifikasi kebijakan optimal dalam melakukan pengujian penetrasi. Selain itu, ASAP juga menggabungkan matriks transisi dan pemodelan *reward* yang spesifik untuk domain untuk menangkap pentingnya kerentanan keamanan dan kesulitan yang melekat dalam mengeksploitasi mereka[10].

7. ***Analysis And Comparative Studies Of Software Penetration Testing Methods***[3]

Topik penelitian ini adalah analisis kerentanan perangkat lunak dan metode identifikasi kerentanan perangkat lunak. Fokus penelitian ini adalah pentingnya keamanan informasi dalam organisasi dan dampak yang dapat terjadi akibat kebocoran informasi. Beberapa permasalahan yang dibahas meliputi kelemahan dalam perangkat lunak, jenis serangan yang sering digunakan oleh penyerang siber, dan kekurangan dalam metode identifikasi kerentanan yang saat ini ada[3].

Pada penelitian ini, metode yang dipergunakan ialah analisis perbandingan terhadap beberapa metode identifikasi kerentanan perangkat lunak. Beberapa metode yang dibandingkan meliputi OSSTMM (*Open Source Security Testing Methodology Manual*), OWASP (*Open Web Application Security Project*) *Testing Guide*, PTES (*Penetration Testing Execution Standard*), NIST *Special Publication* 800-115, BSI (*Study A Penetration Testing Model*), dan ISSAF (*Information System Security Assessment Framework*). Hasil penelitian ini berupa analisis dan perbandingan terhadap beberapa metode identifikasi kerentanan perangkat lunak[3].

8. **Analisis Celah Keamanan Pada Website Dengan Menggunakan Metode *Penetration Testing* Dan *Framework* Issaf Pada Website SMK Al-Kautsar**[11]

   Penelitian ini bertujuan untuk mengidentifikasi celah keamanan pada sistem informasi berbasis website SMK Al-Kautsar Purwokerto dengan menggunakan *Framework* ISSAF. Hasil penelitian menunjukkan bahwa website SMK Al-Kautsar Purwokerto rentan terhadap serangan DDoS (*Distributed Denial of Service*). Hal ini terbukti saat dilakukan pengujian menggunakan tools LOIC (*Low Orbit Ion Cannon*), website tidak dapat diakses selama serangan DDoS berlangsung. Serangan DDoS bertujuan untuk membuat server sibuk dengan permintaan dari client[11].

9. ***Security test MOODLE: a penetration testing case study***[12]

   Rumusan masalah dalam penelitian ini adalah bagaimana mengidentifikasi kerentanan keamanan dalam Moodle dan menguji keefektifan metode yang diusulkan dalam menemukan kerentanan tersebut. Metode yang digunakan dalam jurnal ini adalah kombinasi dari analisis kode sumber statis dan dinamis, serta pengujian penetrasi aplikasi web. Analisis kode sumber statis digunakan untuk mengidentifikasi titik masuk potensial dan implementasi yang rentan, sedangkan analisis dinamis digunakan untuk menyaring fungsi-fungsi kritis yang mungkin rentan[12].

   Pengujian penetrasi aplikasi web digunakan untuk memvalidasi temuan dari fase sebelumnya. Selain itu, digunakan juga alat otomatis seperti RIPS untuk mendeteksi kerentanan tipe taint dalam kode sumber. Hasil dari penelitian ini adalah penemuan sembilan kerentanan keamanan dalam Moodle 2.6 menggunakan metode yang diusulkan. Namun, jurnal ini tidak memberikan detail yang cukup tentang kerentanan yang ditemukan, sehingga sulit untuk mengevaluasi efektivitas metodologi yang diusulkan[12].

10. ***Information Technology Risk Management Using ISO 31000 Based on ISSAF Framework Penetration Testing (Case Study: Election Commission of X City)***[13]

    Rumusan masalah dalam jurnal ini adalah bagaimana manajemen risiko teknologi informasi dapat diterapkan pada Komisi Pemilihan Umum di Kota X menggunakan kerangka kerja ISSAF dan standar ISO 31000. Penulis juga ingin mengetahui bagaimana teknik *penetration testing* dapat digunakan untuk mengidentifikasi dan mengevaluasi risiko keamanan informasi pada situs web organisasi. Hasil penelitian menunjukkan bahwa manajemen risiko teknologi informasi dapat membantu meningkatkan keamanan informasi pada situs web organisasi[13].

Tabel 2.1 Penelitian Sebelumnya

No	Judul	Comparing	Contrasting	Criticize	Synthesize	Summarize
1	*Web Application Penetration Testing Using SQL Injection Attack*[6]	Melakukan penelitian terkait keamanan aplikasi web dengan menggunakan OWASP[6].	Pada penelitian ini menggunakan metode OWASP sedangkan penelitian yang akan dilakukan menggunakan ISSAF[6].	Pada penelitian ini hanya membahas SQL *injection* sebagai ancaman utama pada aplikasi web, tanpa membahas ancaman lainnya seperti *cross-site scripting* (XSS) atau *cross-site request forgery* (CSRF)[6].	Pada penelitian ini membahas pentingnya keamanan aplikasi web dan bagaimana melakukan *penetration testing* menggunakan metode *black-box* dengan fokus pada SQL *injection*. Penulis melakukan pengujian pada beberapa website pemerintah, sekolah, dan komersial dengan teknik SQL *injection*. Hasil pengujian menunjukkan bahwa 80% dari website yang diuji memiliki kelemahan terhadap serangan SQL *injection*[6].	Penulis merekomendasikan untuk melakukan pengujian keamanan secara teratur dan memperbarui sistem keamanan untuk mencegah serangan SQL *injection*[6].
2	*Vulnerability Testing Analysis of XYZ Regional Government Site Using PTES*[7]	Melakukan penelitian terkait pengujian kerentanan pada situs pemerintah daerah XYZ dengan menggunakan PTES (*Penetration Testing Execution Standard*)[7]	Pada penelitian ini menggunakan metode PTES sedangkan penelitian yang akan dilakukan menggunakan ISSAF[7].	Meskipun penelitian ini menggunakan beberapa *tools* untuk melakukan pengujian kerentanan, namun masih banyak *tools* lain yang dapat digunakan untuk melakukan pengujian kerentanan yang lebih komprehensif[7].	Penelitian ini menekankan pentingnya melakukan pengujian kerentanan pada website, terutama pada website pemerintah yang memiliki data sensitif. Dalam hal ini, pengujian kerentanan dapat membantu meningkatkan keamanan website dan	Hasil pengujian menunjukkan bahwa website layanan terpadu memiliki jenis kerentanan dan tingkat risiko yang berbeda-beda sesuai dengan *tools* yang digunakan. Penelitian ini memberikan beberapa

					mencegah serangan yang dapat merugikan pengguna website[7].	rekomendasi untuk meningkatkan keamanan website layanan terpadu pemerintahan daerah XYZ. Rekomendasi tersebut meliputi pembaruan sistem operasi, pembaruan aplikasi, dan peningkatan keamanan jaringan[7].
3	*Security Analysis On Websites Using The Information System Assessment Framework (ISSAF) And Open Web Application Security Version 4 (Owaspv4) Using The Penetration Testing Method*[8]	Melakukan penelitian terkait keamanan pada website menggunakan kerangka kerja ISSAF dan OWASPv4 dengan metode pengujian penetrasi[8].	Pada penelitian ini menggunakan 2 metode yaitu ISSAF dan OWASPv4 sedangkan penelitian yang akan dilakukan hanya menggunakan satu metode yaitu ISSAF[8].	Penelitian ini tidak menyebutkan metodologi atau teknik khusus yang digunakan dalam kerangka ISSAF dan OWASP untuk melakukan pengujian penetrasi[8].	Penelitian ini memberikan ikhtisar komprehensif tentang kerangka kerja ISSAF dan OWASPv4 dan bagaimana mereka dapat digunakan untuk analisis keamanan situs web. Penulis juga menjelaskan pentingnya pengujian penetrasi dan memberikan panduan langkah demi langkah tentang cara melakukannya menggunakan kerangka kerja ini[8].	Penelitian ini membahas penggunaan *framework* ISSAF dan OWASPv4 untuk analisis keamanan situs web dan memberikan wawasan tentang cara melakukan pengujian penetrasi menggunakan *framework* ini. Penulis menekankan pentingnya *cyber security* dalam pengembangan website dan memberikan rekomendasi untuk meningkatkan keamanan website[8].
4	Analisis Keamanan Sistem Informasi Akademik Menggunakan *Open Web Application*	Melakukan penelitian terkait keamanan sistem informasi akademik menggunakan *Open*	Pada penelitian ini menggunakan metode OWASP sedangkan penelitian yang akan	Meskipun penelitian ini memberikan analisis yang cukup detail tentang kerentanan sistem informasi akademik	Hasil Penelitian menunjukkan bahwa terdapat 12 kerentanan pada sistem, dengan empat kerentanan pada *level*	Penelitian ini juga memberikan gambaran tentang skenario serangan pada sistem dan tabel yang

	Security Project Framework[9]	*Web Application Security Project* (OWASP) *Framework*[9].	dilakukan menggunakan ISSAF[9].	menggunakan OWASP *Framework*, namun tidak ada pembahasan tentang bagaimana mencegah atau mengatasi kerentanan yang ditemukan[9].	*medium*, enam pada *level low*, dan dua pada *level informational*[9].	menjelaskan alat yang digunakan dan fungsionalitasnya. Namun, jurnal ini tidak membahas tentang bagaimana mencegah atau mengatasi kerentanan yang ditemukan dan hanya membahas satu kasus studi[9].
5	Implementasi *Penetration Testing Execution Standard* Untuk Uji Penetrasi Pada Layanan *Single Sign-On*[4]	Melakukan penelitian terkait permasalahan keamanan sistem pada layanan *Single Sign-On* (SSO) di sebuah instansi dengan menggunakan PTES[4].	Pada penelitian ini menggunakan metode PTES sedangkan penelitian yang akan dilakukan menggunakan ISSAF[4].	Penulis berhasil mengimplementasikan PTES untuk melakukan uji penetrasi pada layanan *Single Sign-On*. Namun, tidak dijelaskan secara rinci mengenai bagaimana penulis memilih teknik dan alat yang digunakan dalam uji penetrasi tersebut. Selain itu, penulis juga tidak memberikan penjelasan yang cukup mengenai hasil uji penetrasi yang dilakukan[4].	Penulis berhasil mengimplementasikan PTES untuk melakukan uji penetrasi pada layanan *Single Sign-On*. Penulis melakukan uji penetrasi dengan tujuan untuk mengidentifikasi kerawanan dan menguji keamanan sistem dengan mengeskploitasi kerawanan tersebut[4].	Penulis melakukan uji penetrasi dengan tujuan untuk mengidentifikasi kerawanan dan menguji keamanan sistem dengan mengeskploitasi kerawanan tersebut. Dari tujuh tahap uji penetrasi yang dilakukan, berhasil teridentifikasi 12 kerawanan yang terdiri dari 3 kerawanan kategori sedang, 6 kerawanan kategori rendah dan 3 kerawanan kategori informasi. Enam serangan siber telah dilakukan untuk mengeksploitasi kerawanan dengan hasil 3 serangan berhasil dan 3 serangan gagal[4].

6	*Autonomous Security Analysis and Penetration Testing*[10]	Melakukan penelitian terkait pengujian penetrasi dalam konteks keamanan jaringan[10].	Pada penelitian ini melakukan analisis terkait keamanan jaringan sedangkan penelitian yang akan dilakukan yaitu mengenai keamanan web[10].	Penelitian ini menggunakan kerangka kerja yang sangat bergantung pada algoritma pembelajaran penguatan, yang mungkin tidak selalu menjadi pendekatan yang paling efisien atau efektif untuk mengidentifikasi kerentanan keamanan. Selain itu, framework mungkin tidak cocok untuk semua jenis jaringan atau lingkungan keamanan, dan mungkin memerlukan kustomisasi dan konfigurasi yang signifikan untuk mencapai hasil yang optimal[10].	Mengidentifikasi kebijakan optimal untuk melakukan pengujian pentesting. Dengan membuat peta ancaman keamanan dan kemungkinan jalur serangan dalam jaringan menggunakan grafik serangan, kerangka kerja ini dapat memberikan wawasan berharga tentang potensi kerentanan dan membantu organisasi melindungi sistem dan data mereka dengan lebih baik[10].	*Autonomous Security Analysis and Penetration Testing* (ASAP) adalah pendekatan mutakhir untuk penilaian keamanan yang memanfaatkan algoritma pembelajaran penguatan dan grafik serangan untuk mengidentifikasi potensi kerentanan dalam jaringan. Dengan mengotomatiskan banyak tugas yang terkait dengan pengujian pentesting, *framework* ini dapat membantu organisasi melindungi sistem dan data mereka secara lebih efisien dan efektif[10].
7	*Analysis And Comparative Studies Of Software Penetration Testing Methods*[3]	Melakukan penelitian terkait kerentanan perangkat lunak dengan menggunakan 6 metode yaitu OSSTMM, OWASP, PTES, NIST, BSI dan ISSAF[3].	Pada penelitian ini membandingkan 6 metode yaitu OSSTMM, OWASP, PTES, NIST, BSI dan ISSAF. Sedangkan penelitian yang akan dilakukan yaitu hanya menggunakan	Meskipun penelitian ini memberikan perbandingan yang baik antara metodologi pengujian penetrasi yang berbeda, namun tidak ada penjelasan yang cukup tentang bagaimana metodologi ini dapat diterapkan pada	Penelitian ini memberikan perbandingan yang komprehensif antara beberapa metodologi pengujian penetrasi yang berbeda. Hasil penelitian menunjukkan bahwa penggunaan metodologi yang kompleks dan mempertimbangkan risiko	Penelitian ini membahas pentingnya pengujian penetrasi dalam menjaga keamanan informasi organisasi. Selain itu, jurnal ini juga membahas beberapa jenis serangan dan intrusi yang paling

			1 metode, ISSAF[3].	lingkungan produksi yang sebenarnya. Selain itu, jurnal ini tidak membahas secara rinci tentang bagaimana mengatasi masalah yang muncul selama pengujian penetrasi[3].	keamanan yang ada dapat meningkatkan efektivitas pengujian penetrasi[3].	sering digunakan oleh penjahat siber[3].
8	Analisis Celah Keamanan Pada Website Dengan Menggunakan Metode *Penetration Testing* Dan *Framework* Issaf Pada Website SMK Al-Kautsar[11]	Melakukan penelitian terkait keamanan dalam website sistem informasi pada SMK Al-Kautsar Purwokerto dengan menggunakan ISSAF[11].	Pada penelitian ini tahapan yang dipakai dalam *Framework* ISSAF hanya bagian *Assessment*, sedangkan penelitian yang akan dilakukan menggunakan 3 tahapan, *planning & preparation*, *assessment* dan *clean up & Destroy Artifacts*[11].	Pada penelitian ini hanya dilakukan uji keamanan menggunakan satu serangan yaitu DDoS. Akan lebih komprehensif dan berguna untuk menilai keamanan situs web terhadap berbagai serangan yang lebih luas, seperti injeksi SQL, *clickjacking*, dan *brute force*. Dengan membatasi penilaian pada serangan DDoS, postur keamanan situs web secara keseluruhan mungkin belum dievaluasi secara memadai[11].	Pengujian penetrasi yang dilakukan menggunakan *framework* ISSAF mengidentifikasi kerentanan terhadap serangan DDoS di situs web SMK Al-Kutsar, potensi ancaman lain seperti SQL *injection*, *clickjacking*, dan **brute force** tidak dinilai. Penting bagi organisasi untuk melakukan penilaian keamanan yang komprehensif untuk memastikan perlindungan keseluruhan situs web dan data pengguna mereka[11].	Hasil penelitian dengan menggunakan *tools* LOIC yaitu, website SMK Al Kautsar tidak bisa diakses selama proses serangan ddos. Serangan ddos ini bertujuan untuk membuat server sibuk dengan permintaan dari *client*. Akan tetapi website ini terhindar dari serangan XSS, serta serangan yang memanfaatkan *port* yang terbuka yaitu *port* 21. Pada pengujian xss serta *port* 21, penulis gagal mendapatkan akses ke dalam website tersebut[11].
9	*Security test MOODLE: a penetration testing case study*[12]	Melakukan penelitian terkait kerentanan keamanan dalam *Moodle* dengan	Pada penelitian ini menggunakan metode kombinasi dari analisis kode	Penelitian ini tidak memberikan detail yang cukup tentang kerentanan khusus yang ditemukan di	Penelitian menyajikan metodologi pengujian keamanan yang komprehensif untuk	Secara keseluruhan, metodologi menyediakan pendekatan sistematis

		menggunakan kombinasi dari analisis kode sumber statis dan dinamis, serta pengujian penetrasi aplikasi web[12].	sumber statis dan dinamis, sedangkan penelitian yang akan dilakukan menggunakan ISSAF[12].	*Moodle* 2.6, sehingga sulit untuk menilai keefektifan metodologinya. Selain itu, penelitian ini tidak membahas dampak atau tingkat keparahan dari kerentanan yang ditemukan, yang sangat penting untuk memahami potensi risiko terhadap *Moodle*[12].	*Moodle*, menggabungkan analisis kode sumber statis dan dinamis dengan pengujian penetrasi aplikasi web. Analisis statis membantu mengidentifikasi titik masuk potensial dan implementasi yang rentan, sedangkan analisis dinamis menyaring fungsi-fungsi kritis yang mungkin rentan[12].	untuk mengidentifikasi dan memvalidasi kerentanan keamanan di *Moodle*[12].
10	*Information Technology Risk Management Using* ISO 31000 *Based on* ISSAF *Framework Penetration Testing (Case Study: Election Commission of X City)*[13]	Melakukan penelitian terkait bagaimana manajemen risiko teknologi informasi dapat diterapkan pada Komisi Pemilihan Umum di Kota X menggunakan kerangka kerja ISSAF dan standar ISO 31000[13].	Pada penelitian ini dilakukan penerapan manajemen risiko teknologi dengan menggunakan ISSAF dan Standar ISO 3100, sedangkan penelitian yang akan dilakukan yaitu analisis keamanan web dengan ISSAF[13].	Studi kasus penelitian ini hanya dilakukan pada satu lembaga dan tidak mencakup variasi yang cukup dalam jenis organisasi yang berbeda[13].	Penelitian ini menyajikan sebuah studi kasus tentang bagaimana manajemen risiko teknologi informasi dapat diterapkan pada Komisi Pemilihan Umum di Kota X menggunakan kerangka kerja ISSAF dan standar ISO 31000. Penulis menggunakan teknik penetration testing untuk mengidentifikasi dan mengevaluasi risiko keamanan informasi pada situs web Komisi Pemilihan Umum[13].	Hasil penelitian menunjukkan bahwa manajemen risiko teknologi informasi dapat membantu meningkatkan keamanan informasi pada situs web organisasi[13].

2.2 Dasar Teori

2.2.1 Website

Sebuah website yakni sekumpulan halaman yang terkait dan mampu diakses dengan Internet. Halaman-halaman ini umumnya tersusun atas teks, gambar, video, dan elemen lain yang membentuk tampilan yang terstruktur dan dirancang untuk memberikan informasi atau fungsi tertentu. Website mempunyai sebuah alamat unik yang dikenal dengan URL (*Uniform Resource Locator*), yang dipergunakan guna mengaksesnya melalui peramban web. Setiap halaman dalam website memiliki URL yang berbeda, tetapi semua halaman tersebut terhubung satu sama lain melalui tautan atau *hyperlink*[14].

2.2.2 CMS

CMS (*Content Management System*) dapat didefinisikan sebagai sistem yang digunakan untuk mengelola konten dalam bahasa Indonesia. CMS adalah suatu perangkat lunak yang memungkinkan pengguna agar dengan mudahnya membuat, mengedit, mengatur, dan mengelola konten di situs web tanpa membutuhkan pengetahuan teknis yang kompleks. CMS menyediakan antarmuka pengguna yang mudah dipahami dan berbasis web, sehingga memungkinkan pengguna yang tidak memiliki keahlian dalam pemrograman atau desain web untuk dengan mudah mengelola dan mempublikasikan konten[15].

Dengan menggunakan CMS, pengguna dapat membuat dan mengedit halaman, mengunggah gambar dan video, membuat posting blog, mengatur tata letak situs, dan menjalankan berbagai tugas terkait konten. Terdapat beberapa contoh CMS yang populer, seperti WordPress, Joomla, Drupal, dan Magento. CMS memungkinkan pemilik situs web atau pengelola konten untuk dengan cepat mengubah dan memperbarui konten mereka, sehingga mempermudah dalam pemeliharaan dan pengelolaan situs web[15].

2.2.3 Wordpress

WordPress adalah salah satu CMS yang terkenal dan populer yang sering digunakan untuk membuat dan mengelola situs web. Sebagai

perangkat lunak sumber terbuka yang gratis, WordPress awalnya dirancang sebagai alat yang memudahkan pembuatan dan pengelolaan blog. Namun, seiring berjalannya waktu, *platform* ini telah mengalami perkembangan menjadi solusi yang mampu mengatasi berbagai jenis situs web[16].

Dengan antarmuka pengguna yang intuitif dan berbasis web, WordPress memungkinkan pengguna dengan berbagai tingkat pengetahuan teknis untuk dengan mudah membuat dan mengedit konten. Platform ini dilengkapi dengan fitur dan fungsi yang beragam, termasuk pembuatan halaman, penulisan posting blog, pengelolaan media seperti gambar dan video, pemasangan tema dan *plugin*, pengaturan tata letak, manajemen komentar, dan masih banyak lagi[16].

2.2.4 *Penetration Testing*

Penetration Testing ialah metode pengujian yang dipergunakan guna menguji tingkat keamanan sistem atau jaringan komputer dengan melakukan simulasi serangan. Bertujuan untuk mendeteksi potensi kerentanan yang mungkin terjadi akibat kelemahan dalam sistem, pengaturan yang tidak tepat, atau kesalahan operasional. Hasil laporan dari *Penetration Testing* memberikan informasi kepada pemilik sistem tentang celah keamanan yang ada dalam sistem mereka, sehingga dapat dievaluasi dan dilakukan langkah perbaikan yang tepat. Hal ini memungkinkan tindakan pencegahan yang lebih dini untuk menambal celah keamanan yang teridentifikasi dalam sistem komputer yang sedang digunakan[17], [18].

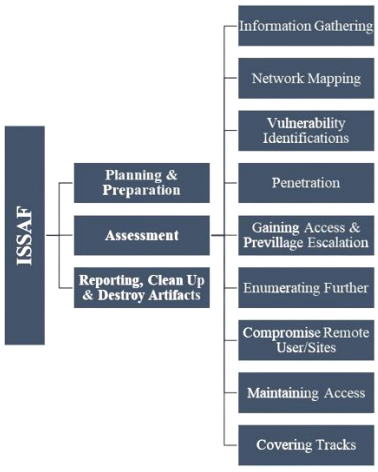
Banyak orang sering keliru mengartikan *Penetration Testing* sebagai *Vulnerability Analysis*. Dalam *Vulnerability Analysis*, dilaksanakan pemeriksaan sistem guna memastikan adanya potensi celah keamanan. Namun demikian, *Penetration Testing* melibatkan simulasi serangan yang menyerupai tindakan seorang peretas untuk secara aktif mencari dan mengeksploitasi celah keamanan yang ada. Atas dasar tersebut, mampu ditarik kesimpulan jika *Penetration Testing*

merupakan tahap yang lebih lanjut setelah *Vulnerability Analysis* dilakukan[17], [18].

2.2.5 ISSAF

Information Sistem Security Assessment Framework (ISSAF) adalah sebuah kerangka kerja yang terstruktur dan terarah yang digunakan untuk mengelompokkan informasi, mengevaluasi, dan melaporkan hasil pengujian keamanan sistem terhadap domain-domain yang diuji. Kerangka kerja ini juga menyediakan analisis terhadap hasil pengujian tersebut. Kerangka kerja ini menyediakan metode uji penetrasi yang dirancang khusus untuk mengevaluasi jaringan, sistem, dan kontrol aplikasi[1], [5], [19], [20], [21], [22], [23].

Dalam melakukan Penetration Testing atau uji penetrasi, terdapat sejumlah metodologi yang dapat dipilih, seperti *Information Systems Security Assessment Framework* (ISSAF), *Penetration Testing Execution Standard* (PTES), *Open Web Application Security Project* (OWASP), dan *Open Source Security Testing Methodology Manual* (OSSTMM). Namun, dibandingkan dengan metode yang lainnya ISSAF memiliki nilai yang tinggi dalam melindungi infrastruktur dengan mengevaluasi kontrol keamanan yang ada terhadap kerentanan yang kritis. ISSAF juga memiliki struktur yang jelas yang dapat mengarahkan penguji dengan tahapan-tahapan penilaian yang kompleks, sehingga uji penetrasi dapat dilakukan secara efektif, lengkap, dan akurat



Gambar 2.1 Framework ISSAF

1. Fase *planning and preparation*

   Fase pertama ISSAF yang terdiri dari tahap persiapan dan pengumpulan informasi dari web target yang akan diuji melalui *penetration testing*[24], [25].

2. Fase *Assessment*

   Fase *Assessment* merupakan tahap di mana sistem informasi yang telah disepakati pada Fase *Planning and Preparation* diuji coba. Fase *Assessment* ini terdiri dari sembilan sub-fase yang dapat dikembangkan, yaitu[24], [25]:

   a. Pengumpulan Informasi (*Information Gathering*)

      Tahap pengumpulan informasi umum pada target, juga dikenal sebagai tahap *information gathering*, merupakan langkah awal dalam memperoleh berbagai informasi terkait dengan target yang sedang dituju. Informasi tersebut mencakup data IP tujuan, informasi pendaftar dan *administrator*, detail reverse DNS, pencarian IP, dan informasi umum lainnya.

   b. Pemetaan Jaringan (*Network Mapping*)

      Pemetaan Jaringan adalah langkah di mana data rinci tentang jaringan di lokasi yang dituju dikumpulkan. Salah satu jenis data yang dikumpulkan pada tahap ini adalah informasi tentang *port* TCP dan UDP yang ada pada sistem target.

   c. Identifikasi Keterangan (*Vulnerability Identification*)

      Tahap identifikasi kerentanan (*vulnerability identification*) adalah proses pemindaian website target dengan tujuan untuk mengidentifikasi kelemahan keamanan yang ada di dalamnya.

   d. Penetrasi (*Penetration*)

      Fase penetrasi merupakan tahapan di mana dilakukan simulasi serangan terhadap web target dengan tujuan untuk menemukan kerentanan dalam sistem keamanannya.

   e. Mendapatkan Akses dan Peningkatan Hak Istimewa (*Gaining Access and Privilege Escalation*)

Fase ini adalah langkah pengujian yang dilakukan dengan tujuan untuk mendapatkan akses ke sistem target. Dalam penelitian ini, dilakukan berbagai jenis akses, termasuk akses ke sistem pengguna, akses ke sistem administrator, dan akses ke sistem lainnya.

f. Perencanaan Lebih Lanjut (*Enumerating Further*)

Perencanaan Lanjutan (*Enumerating Further Phase*) adalah tahap dalam pengujian di mana dilakukan analisis dan pengolahan lebih lanjut terhadap semua informasi terkait dengan kata sandi yang diperoleh dari sumber-sumber web.

g. Kompromi Pengguna/Situs Jarak Jauh (*Compromise Remote User/Sites*)

Fase ini merupakan tahap pengujian di mana dilakukan eksploitasi untuk mencapai akses pengguna *root* melalui koneksi web jarak jauh.

h. Mempertahankan Akses (*Maintaining Access*)

Tahap *maintaining access* adalah tahap pengujian di mana *backdoor* ditanamkan ke dalam sistem website target.

i. Covering Tracks

Pada tahap ini, penguji akan melakukan tindakan untuk menghilangkan jejak-jejak yang ada dengan cara menyembunyikan file dan menghapus *file log* yang dihasilkan pada tahap sebelumnya.

3. Fase *Reporting, Clean Up and Destroy Artifacts*

Fase ini merupakan tahap terakhir di mana dilakukan penghapusan sistem informasi yang digunakan dalam penelitian. Tujuannya adalah untuk mencegah penyalahgunaan di luar konteks penelitian dan persetujuan yang telah diberikan pada tahap *Planning and Preparation*. Fase *Clean Up and Destroy Artifacts* dapat dibagi menjadi dua kelompok, yaitu:

a. *Reporting*

Pada tahap ini, penguji akan menyusun laporan yang menjelaskan hasil pengujian beserta rekomendasi dan tindakan yang direkomendasikan untuk mengatasi temuan-temuan yang ditemukan.

b. *Clean Up and Destroy Artifacts*

Pada tahap ini, semua informasi yang telah dibuat atau dimasukkan ke dalam sistem harus dihapuskan. Jika tidak memungkinkan dilakukan secara langsung pada sistem jarak jauh, pihak yang sedang diuji harus diberitahu sehingga tim TI dapat menghapus informasi tersebut setelah menerima laporan.

2.2.6 *Tracer Study*

Tracer study merupakan metode penelitian yang dilakukan untuk melacak dan menganalisis jejak karir dan kinerja lulusan suatu institusi pendidikan atau program pelatihan. Tujuan dari *tracer study* adalah untuk memperoleh informasi tentang keberhasilan lulusan dalam memasuki dunia kerja, tingkat keterampilan yang dimiliki, relevansi pendidikan dengan pekerjaan yang dijalani, serta pemenuhan kebutuhan pasar kerja[26].

2.2.7 Kali Linux

Linux adalah sebuah sistem operasi dengan sifat terbuka atau *open source*, yang berarti bahwa siapapun dapat mengembangkan sistem operasi ini. Sistem operasi ini awalnya dikembangkan oleh seorang *hacker* bernama Linus Benedict Torvalds. Salah satu varian dari Linux adalah Kali Linux, yang sering digunakan untuk melakukan *penetration testing* terhadap website dan jaringan komputer. Kali Linux dikembangkan oleh *Offensive Security*[23], [27].

2.2.8 Keamanan Informasi

Keamanan informasi merupakan upaya dan langkah-langkah yang dilakukan untuk melindungi informasi dari akses, penggunaan, perubahan, atau pengungkapan yang tidak sah. Tujuannya adalah

menjaga kerahasiaan, integritas, dan ketersediaan informasi agar tetap terlindungi dari ancaman dan risiko yang mungkin terjadi.

2.2.9 Jenis-Jenis Ancaman Keamanan Informasi

Serangan atau gangguan pada keamanan informasi dibagi menjadi empat kategori, yaitu[28]:

1. *Interception*

   Proses serangan atau gangguan sistem keamanan di mana pihak yang tidak berwenang mampu mengakses dan mengambil informasi yang tersedia atau terjadi pencurian informasi, serupa dengan penyadapan. contohnya yaitu, penyerang dapat memanfaatkan teknik seperti *sniffing* (memantau data yang dikirim melalui jaringan) atau serangan *Man-in-the-Middle* (MITM) untuk mendapatkan akses ke informasi sensitif[28].

2. *Interruption*

   Interruption atau Interupsi Proses serangan atau gangguan pada sistem keamanan di mana informasi yang diminta menjadi tidak tersedia atau tidak dapat diakses. Contohnya yaitu serangan DDoS yang mengakibatkan sistem terbanjiri oleh lalu lintas yang berlebihan, serta serangan fisik yang merusak perangkat keras sistem[28].

3. *Fabrication*

   Proses serangan atau gangguan pada sistem keamanan di mana pihak yang tidak mempunyai wewenang melakukan penyisipan objek palsu melalui pemalsuan informasi yang ada dan menargetkan pengguna atau pengguna lain. Contohnya yaitu, *faked mails* dan *spoofing*[28].

4. *Modification*

   Proses serangan atau gangguan pada sistem keamanan di mana informasi yang terdapat pada sistem mampu diubah oleh individu yang tidak mempunyai wewenang, mengakibatkan perubahan pada integritas informasi tersebut. Contohnya meliputi

serangan *man-in-the-middle*, pengelolaan (*cracking*) *file*, dan virus[28].

2.2.10 Macam-Macam Bentuk Serangan pada Keamanan Website

Berikut adalah beberapa bentuk serangan umum pada keamanan website:

1. *Local File Inclusion* (LFI) & *Remote File Inclusion* (RFI)

   Local File Inclusion (LFI) yakni suatu kerentanan yang terjadi pada situs web yang memungkinkan seseorang untuk mengakses *file-file* pada *server* hanya melalui URL. Dalam LFI, serangan terjadi pada *level* lokal, di mana penyerang dapat memanipulasi URL untuk mendapatkan akses ke file-file yang seharusnya tidak dapat diakses. Sementara itu *Remote File Inclusion* (RFI) yakni kerentanan di mana suatu situs web memungkinkan seseorang untuk menyertakan atau memasukkan *file* dari *server* eksternal. Dalam RFI, serangan terjadi melalui penyisipan file dari sumber eksternal ke dalam situs web target. Hal ini dapat memungkinkan penyerang untuk menjalankan kode berbahaya dari *file* yang disertakan tersebut.

2. *Cross-Site Scripting* (XSS)

   Kadang-kadang, seorang peretas bisa memanfaatkan kelemahan pada sisi klien daripada menyerang *server* yang lebih sulit. Serangan ini sulit terdeteksi karena beroperasi dari sisi klien. Salah satu bentuk serangan yang menggunakan pendekatan ini adalah *Cross-Site Scripting* (XSS), yang bukan sama dengan CSS (*Cascading Style Sheets*) yang mana ialah bahasa pemrograman web guna mengatur tampilan sebuah situs web[29].

   XSS memanfaatkan kelemahan pada sisi klien untuk menyisipkan skrip berbahaya yang dieksekusi oleh browser pengguna. Hal ini,dapat mengubah tampilan dan perilaku situs web yang menggunakan bahasa pemrograman seperti HTML dan XHTML.

```
<script language="javascript">window.alert ('i love this site')</script>
```

Silahkan ubah kalimat "*I love this site*" dengan kata-kata pribadi. Jika muncul peringatan JavaScript di *browser*, itu menandakan bahwa situs ini rentan terhadap serangan XSS[2].

3. *Phishing*

   Dalam teknik hacking ini, tekniknya yaitu mencoba menjadikan seseorang mengunjungi situs yang salah dengan tujuan memperoleh informasi rahasia seperti *username*, *password*, atau informasi sensitif lainnya. Biasanya, pelaku membuat situs dengan nama domain yang mirip dengan situs aslinya. Istilah *phishing* sering dikaitkan dengan *web spoofing* dan *DNS spoofing*. Teknik *phishing* juga dikenal dengan istilah *fake login*, di mana seseorang memasukkan login pada halaman yang sebenarnya bukan halaman aslinya[30].

4. *SQL Injection*

   SQL injection yakni serangan keamanan yang dilakukan pada aplikasi klien dengan mengubah perintah atau sintaks SQL[2]. Terdapat dua jenis *SQL injection*:

   a. *Blind SQL Injection* adalah metode yang digunakan untuk memanipulasi sintaks SQL dalam suatu situs web yang mempunyai kerentanan keamanan, dengan tujuan mengakses dan melihat isi dari *database* SQL yang terkait[2].

   b. *Advanced SQL Injection* adalah metode yang lebih canggih dalam *SQL Injection*, di mana tidak hanya memungkinkan akses ke *database*, tetapi juga memungkinkan pembuatan *shell* ataupun *backdoor* pada situs target. Dalam teknik ini, seorang peretas nantinya mencoba melangsungkan serangan *SQL Injection* dan memasang *shell* pada situs mempergunakan sintaks SQL[2].

5. *Defacing*

   Defacing yakni tindakan peretasan yang melakukan perubahan tampilan dari suatu situs web. Metode yang diterapkan

dapat beragam, termasuk *SQL Injection*, pencarian *password*, dan teknik lainnya.

6. MITM (*Man In The Middle*) *Attack*

   Serangan MITM (*Man In The Middle*) merupakan serangan keamanan dimana seorang peretas berusaha menyusup di antara dua pihak yang sedang berkomunikasi, seperti pengguna dan *server*, dengan tujuan mengambil kendali atas komunikasi tersebut. Tujuan dari serangan ini adalah untuk memonitor, memodifikasi, atau mencuri data yang sedang dikirimkan di antara keduanya. Penyerang dalam serangan MITM memiliki kemampuan untuk memanipulasi komunikasi dan bahkan mengirimkan data palsu kepada kedua belah pihak, sehingga menciptakan ilusi jika mereka berkomunikasi dengan langsung antara satu dengan yang lainnya, padahal sebenarnya ada pihak tengah yang terlibat[31].

7. *Session Hijacking*

   Session Hijacking yakni serangan yang dilaksanakan oleh seorang penyerang melalui maksud mengambil alih sesi yang sedang berlangsung antara pengguna dan *server*. Pada serangan ini, penyerang berupaya untuk memperoleh informasi penting seperti ID sesi, *cookie*, atau token otentikasi yang digunakan untuk mengidentifikasi dan mengautentikasi pengguna. *Session hijacking* melibatkan penggunaan metode seperti pemantauan (*capturing*), serangan *brute force*, atau rekayasa balik (*reverse engineering*) untuk mendapatkan ID sesi yang digunakan. Setelah mendapatkan ID sesi, penyerang dapat mengambil alih kendali atas sesi yang dimiliki oleh pengguna lain sewaktu sesi tersebut berlangsung.

8. DDOS (*Distributed Denial of Service*)

   DDoS (*Distributed Denial of Service*) yakni serangan yang bertujuan guna mengganggu ataupun menghambat ketersediaan layanan pada sebuah sistem atau jaringan. Dalam serangan DDoS, penyerang menggunakan sejumlah besar mesin atau perangkat yang terinfeksi (biasanya dalam *botnet*) untuk secara bersamaan

mengirimkan lalu lintas yang sangat tinggi ke target yang ingin diserang. Tujuannya adalah untuk melampaui kapasitas sistem atau jaringan target sehingga menyebabkan penurunan kinerja, penolakan akses, atau bahkan kegagalan sistem secara keseluruhan. Serangan DDoS mengakibatkan gangguan layanan dan dapat menyebabkan kerugian finansial, reputasi, dan kehilangan data bagi korban serangan[2].

9. CSRF (*Cross-Site Request Forgery*)

   CSRF (*Cross-Site Request Forgery*)adalah serangan yang memanfaatkan kepercayaan antara pengguna dan situs web yang sedang dikunjungi. Dalam serangan CSRF, penyerang berusaha memaksa pengguna yang telah diperdaya untuk melakukan tindakan yang tidak diinginkan di dalam situs web yang sedang mereka kunjungi.

   Biasanya, serangan CSRF terjadi ketika pengguna yang telah masuk ke suatu situs web mengunjungi halaman web berbahaya yang dikendalikan oleh penyerang. Halaman web tersebut kemudian secara otomatis mengirimkan permintaan ke situs web yang dituju, memanfaatkan kredensial dan kepercayaan yang telah diperoleh dari pengguna.

   Dalam skenario serangan CSRF, penyerang dapat melakukan berbagai tindakan berbahaya atas nama pengguna, seperti mengubah data, melakukan transaksi keuangan yang tidak diinginkan, atau menghapus informasi penting. Serangan ini dapat menyebabkan kerugian bagi pengguna dan pemilik situs web.

10. *Brute Force*

    Brute force adalah sebuah teknik yang digunakan untuk secara berurutan mencoba semua kemungkinan kombinasi dalam harapan menemukan solusi yang diinginkan. Dalam bidang keamanan komputer, *brute force* sering kali digunakan untuk mencoba memecahkan kata sandi atau mendapatkan akses ke

informasi terbatas dengan mencoba semua kemungkinan kombinasi yang ada sampai ditemukan yang tepat[32], [33], [34].

Dalam serangan *brute force*, penyerang akan berulang kali mencoba semua kemungkinan kata sandi atau kunci enkripsi guna memperoleh akses yang tidak sah pada sistem atau data yang dilindungi. Metode ini seringkali memerlukan waktu dan sumber daya komputasi yang besar, tergantung pada kompleksitas kata sandi atau kunci yang harus dipecahkan[32], [33], [34].

2.2.11 Macam-Macam *Hacker*

Di bawah ini terdapat beberapa jenis-jenis hacker yang sering dikenal:

1. *White Hat Hacker*

   White Hat Hacker, juga dikenal sebagai "*ethical hacker*", yakni orang yang memiliki pengetahuan mendalam tentang keamanan komputer dan menggunakan keterampilan mereka untuk melakukan pengujian keamanan, mengidentifikasi kerentanan, dan membantu melindungi sistem dari serangan. Para *white hat hacker* ini bekerja secara legal dan etis dengan izin dari pemilik sistem yang sedang diuji[35].

2. *Black Hat Hacker*

   Black Hat Hacker, yang juga dikenal sebagai "*cracker*", adalah orang yang memanfaatkan kerentanan keamanan dalam sistem komputer dan jaringan untuk mendapatkan keuntungan pribadi atau mencapai tujuan jahat. Para *black hat hacker* ini terlibat dalam kegiatan ilegal seperti pencurian data, merusak sistem, dan melakukan serangan keamanan lainnya[35].

3. *Grey Hat Hacker*

   Grey Hat Hacker ialah kombinasi antara *white hat* dan *black hat hacker*. Mereka melakukan tindakan *hacking* tanpa izin, namun dengan niat yang tidak jahat. *Grey hat hacker* memiliki kemampuan untuk menemukan kerentanan dalam sistem dan memberitahu pemiliknya, meskipun mereka melakukannya tanpa izin resmi[35].

4. *Suicide Hacker*

   Suicide hacker adalah seorang *hacker* yang tidak merasa takut terhadap ancaman hukum, dan tujuannya hanya untuk menciptakan kekacauan sebanyak mungkin[35].

5. *Script Kiddie*

   Script Kiddie yakni seseorang yang mempunyai pengetahuan terbatas atau bahkan tidak memiliki pengetahuan *hacking*, namun menggunakan alat dan skrip yang telah dibuat oleh *hacker* lain untuk melancarkan serangan. Biasanya, mereka tidak memiliki pemahaman mendalam tentang teknik *hacking* yang mereka gunakan[35].

6. *Hacktivist*

   Hacktivist adalah orang atau kelompok yang memanfaatkan kemampuan *hacking* mereka untuk mendukung tujuan politik atau sosial tertentu. Mereka melakukan serangan terhadap situs web atau infrastruktur yang terkait dengan entitas atau pemerintah yang dianggap bertentangan dengan kepercayaan mereka[35].

7. *State-Sponsored Hacker*

   State-Sponsored Hacker adalah para *hacker* yang bekerja atas perintah pemerintah atau badan intelijen negara. Tujuan mereka seringkali melibatkan aktivitas spionase, peretasan sistem militer, industri, atau lembaga pemerintah asing[35].