Dinamika
PROSIDING-DISEMINASI NASIONAL VIII
e-issn : 3063-4091
hal. 359-377

CRYPTOGRAPHIC FAILURES PADA WEBSITE: SYSTEM LITERATURE REVIEW

Sri Anita[1], Sunu Aditya Mahadany[2], Widya Lelisa Army[3], Namora[4]

Faculty of Business and Technology Pertiwi Universit Bekasi, Indonesia, Research and Innovation PT Integra Solusi Teknotama Bandung, Indonesia, Faculty of Business and Technology Pertiwi University Bekasi, Indonesia, Faculty of Business and Technology Pertiwi University Bekasi, Indonesia

sri.anita@pertiwi.ac.i, monstm@gmail.com, Widya.lelisa@pertiwi.ac.id, Namora@pertiwi.ac.id

ABSTRACT

The transformation of digital transactions is currently experiencing very rapid development, which has given rise to many organizations using websites for transactions, both internal and external. This has given rise to attacks where there are actors who want to steal accounts in order to control the legitimate website application owner. Cryptographic failures are one of the causes of vulnerabilities that occur, which are published by the OWASP top 10, which summarizes the 10 most common attacks that occur in the world, including Indonesia. Cryptographic failures cause sensitive data exposure, which is very detrimental to account owners or application owners, and the impact of which can damage reputation and data theft. This article will discuss the causes of websites being attacked, how to prevent them, and proposed scenarios to mitigate them. The method used to research is the Literature Review system method, which has been carried out by previous researchers in reviews and scientific articles. The results obtained from this article are proposed scenarios that apply the encryption method or Hash SHA-2 526. The importance of protecting websites for both website owners and account users because if they have been attacked by cryptographic failures, one of the losses is exposure of personal data, data theft, and ,of course, the reputation that will be at stake.

***Keywords**: Cryptographic Failures, Types of website attacks, Sensitive Data Exposure, Data Privacy, SHA2.*

ABSTRAK

Transformasi transaksi digital saat ini mengalami perkembangan yang sangat pesat, hal ini memunculkan banyak organisasi menggunakan website untuk transaksi baik urusan internal dan eksternal. hal ini memunculkan adanya serangan dimana ada aktor yang menginginkan mencuri akun untuk dapat menguasai pemilik aplikasi website yang sah. Cryptographic failures salah satu penyebab kerentanan yang terjadi yang dipublikasi oleh OWASP top 10 yang merangkum 10 serangan terbanyak yang terjadi didunia artinya termasuk Indonesia. Cryptographic failures menyebabkan sensitive data exposure yang sangat merugikan bagi pemilik akun atau pemilik aplikasi, yang dampaknya dapat merusak reputasi dan pencurian data. Dalam artikel ini akan dibahas mengenai penyebab website terkena serangan, cara mencegah, dan usulan skenario untuk memitigasinya. Metode yang dilakukan untuk meneliti adalah metode system Literature Review yang telah dilakukan peneliti sebelumnya dalam ulasan dan artikel ilmiah. Hasil yang diperoleh dari artikel ini usulan skenario dan menerapkan metode enkripsi atau Hash SHA 2 526. Pentingnya melindungi website baik bagi pemilik website dan pengguna akun sebab jika sudah terserang cryptographic failures kerugiannya salah satunya adalah terekseposnya data pribadi, pencurian data, dan tentunya reputasi yang akan dipertaruhkan.

Katakunci: *Cryptographic Failures, Types of website attacks, Sensitive Data Exposure, Privasi Data, SHA2.*

PENDAHULUAN

Dalam pemanfaatan teknologi menawarkan kemudahan baik secara akses terhadap kemajuan bisnis, maupun berkaitan dengan pemasaran produk dan jasa untuk mencapai tujuan bisnis sukses. Transformasi teknologi terutama di Indonesia sudah dimulai di berbagai sektor industri baik kalangan pemerintahan dan swasta dewasa ini. Namun apakah transformasi teknologi yang menawarkan kemudahan, modernisasi, transparasi, dan lainnya tersebut di imbangi dengan faktor risiko ancaman yang mengintai hal tersebut. Tentu dengan hasil penelusuran sebanyak 609.393.852 serangan siber yang dirilis oleh BSSN melalui laporan honeynet tahun 2024[1]. Serangan yang muncul tidak hanya berasal dari Indonesia, namun juga dari India, Amerika Serikat, China, Vietnam, Bangladesh, Filipina, Rusia, Pakistan, dan Singapura. Jenis serangan juga berbagai macam jenisnya, salah satunya adalah serangan jenis cryptographic failures.

Dalam serangan yang muncul pada platform digital indonesia adalah meliputi sektor pemerintahan, keuangan, organisasi swasta baik profit dan non profit[1]. Dalam hal ini sasaran serangan siber tentunya mempunyai maksud yaitu salah satunya adalah pengambilan data yang ada di dalam platform digital tersebut [2], [3]. Hal yang akan terjadi jika data yang ada di platform digital di curi dan disalah gunakan, tentu akan sangat merugikan bagi pemilik platform dan pemilik data pribadi. Jika data sudah di curi akan menjadi ancaman yang serius yaitu menyangkut reputasi perusahaan dan organisasi. Selain itu pencurian identitas atau

informasi pribadi dapat menjadi potensi pelanggaran hukum berat seperti pada peraturan (GDPR dan UU PDP), selain itu ada potensi penyalahgunaan akun atau keuangan yang dapat merugikan financial [4]. Kasus terberat yang pernah terjadi pada kasus cryptographic failures yaitu terjadi pada tahun 2016 yaitu kasus Panama Papers, menyerang website firma hukum Mossack Fonseca menggunakan software yang tidak di enkripsi dengan baik, termasuk menggunakan wordpress dengan plugin yang usang dan tidak menggunakan HTTPS. Dampak kejadian ini adalah 11.5 juta dokumen bocor, sehingga data keuangan rahasia milik politisi, selebritis, dan orang kaya dunia terekspos. Dalam kejadian tersebut tidak hanya merugikan organisasi firma hukum Mossack Fonseca namun klien yang ada di dalam firma hukum tersebut[5].

Cryptographic Failures adalah salah satu kerentanan serangan siber pada aplikasi website yang sering terjadi tentunya menimbulkan kerugian dan risiko yang besar tidak hanya reputasi melainkan kerugian financial [4], [5]. Jenis serangan yang seharusnya dapat diantisipasi dan penting untuk diketahui dan wajib dipahami oleh organisasi yang memiliki aplikasi website. Diharapkan dengan adanya artikel ini dapat memberikan pedoman, dan rujukan teknis untuk memahami apa saja yang perlu dilakukan dalam manajemen aplikasi website untuk mengurangi kerentanan melalui celah cryptographic failures [4], [6]–[8]. Dalam artikel ini akan dibahas mengenai teknik dan langkah apa saja yang perlu dilakukan untuk terhindar dari cryptographic failures dari pengalaman pribadi yang sudah dilakukan dan hasil dari system literature review dari berbagai artikel yang terakreditasi baik nasional dan internasional, serta situs web pemerintah dan organisasi profesional.

e-issn : 3063-4091
hal. 359-377

LITERATURE REVIEW

A. SHA 2 (Secure Hash Algorithm 2)

SHA2-512/256 merupakan fungsi *hash* kriptografi yang dirancang oleh Badan Keamanan Nasional Amerika Serikat. Fungsi *SHA2-512/256* dibangun menggunakan struktur **Merkle-Damgard** dari fungsi kompresi satu arah. Fungsi kompresi yang digunakan menggunakan struktur **Davies-Meyer** dari chipher blok yang dirahasiakan . *SHA 2-512/256* menggantikan fungsi pendahulunya yaitu *SHA1* yang sudah digunakan secara luas dalam sistem keamanan kriptografi, *SHA 2* ini diterapkan untuk mengatasi kekurangan yang ada di *SHA 1*. *SHA 2* versi terbaru ini diakui untuk sekarang lebih aman dan fleksibel sehingga menjadi pilihan yang paling tepat dalam perkembangan teknologi saat ini [9].

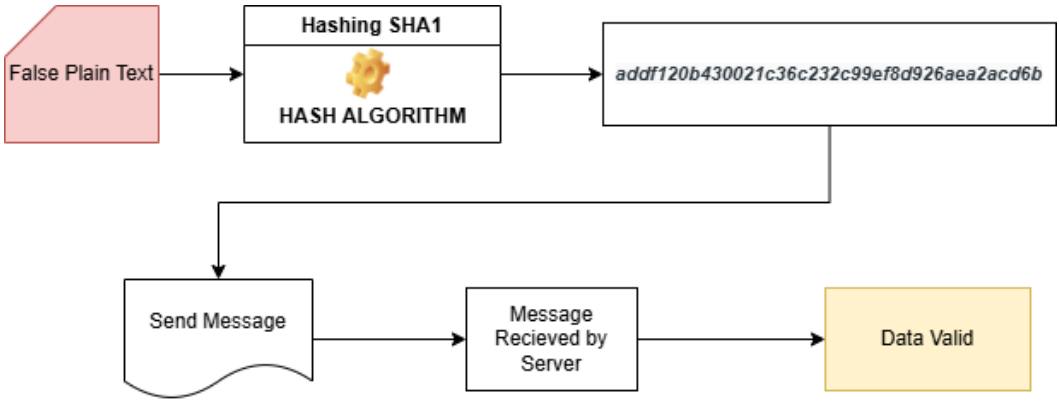
SHA 2 menghasilkan nilai hash 224, 256,384, atau 512 bit. *SHA2* memiliki sertifikat yang lebih baik, merupakan penerus *SH 1* dan pendahulu *SHA 3* dan *Hash* yang dihasilkan *SHA 2* kuat. Berikut ini merupakan hasi yang ada dalam *SHA 2*: *86d755349c6b9f95f365c6ffe7734f25bf2b00cabe8c6bc5f2b8b746c1aac332* [4], [9], [10].

B. Cryptographic Failures

Jenis kerentanan yang terjadi pada *website* yang masuk ke dalam *Top* 10 *OWASP* pada peringkat ke-2. *Cryptographic failures* biasa dikenal dengan istilah *Sensitive Data Exprosure*, yang merupakah gejala dari akar penyebab kegagalan yang terkait dengan kriptografi [4], [7], [11], [12]. Kriptografi sangat penting untuk perlindungan data, tetapi jika terjadi kegagalan dalam enkripsinya dapat

menyebabkan terungkapnya informasi atau data sensitive. Kegagalan kriptografi dapat disebabkan oleh berbagai kelemahan seperti kata sandi yang dikodekan secara keras dan entropi yang tidak memadai. Dampaka kegagalan kriptografi meliputi pencurian data, pelanggaran, dan kerusakan reputasi. Untuk mengurangi kegagalan kriptografi gunakan kunci enkripsi yang kuat seperti teknik pengkodean Hash SHA2 dan dilakukan pengujian penetrasi secara berkala.

Cryptographic failures yang merupakan serangan yang menyerang data sensitif, dinyatakan dalam gambar 1 dibawah ini:



Gambar 1. Pesan palsu dalam Hash SHA1

Skenario yang dilakukan penyerang adalah dengan cara memasukan data palsu (privasi data) yang akan digunakan sebagai validasi data privasi asli (sesungguhnya), data di enkripsi menggunakan algoritma yaitu teknik hashing SHA1, dimana pesan yang dimasukan apapun akan berubah menjadi 512 bit. Kemudian data akan dikirim ke server aplikasi dan tentunya server aplikasi akan menerima sebagai data asli (sesungguhnya) karena sudah tersamarkan datanya. Proses dalam cryptographic failures terletak pada saat hashing SHA1. Dalam teknologi keamanan terbaru sudah tidak lagi disarankan menggunakan SHA1

e-issn : 3063-4091
hal. 359-377

sebagai algoritma untuk enkripsi data karena sudah dapat ditembus dengan cara brute force. Disarankan untuk menggunakan algoritma SHA2 yang akan dijelaskan pada bab analisa.

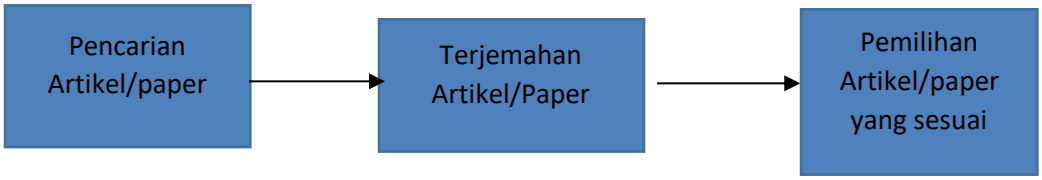
METODOLOGI PENELITIAN

Systematic literature review merupakan salah satu jenis penelitian yang metode penelitian yang cukup terstruktur untuk melakukan identifikasi, evaluasi, penafsiran bukti-bukti penelitian sebelumnya yang relevan dan tervalidasi untuk mencari kebenaan serta pertanyaan-pertanyaan penelitian terdahulu [13]. Disebut sebagai jenis penelitain yang tervalidasi karena artikel yang digunakan merujuk pada artikel yang sudah terbit di jurnal yang terindeks baik nasional dan internasional, dan merupakan karya ilmiah laporan penelitian serta website onlin yang membahas secara deail keilmuwan tertentu di internet.

Tahapan penelitian

Tahapan-tahapan dalam penelitian ini diilustrasikan pada gambar2, merupakan penjelasan langkah pencarian sumber untuk mencari artikel dan laporan karya ilmiah yang relevan terhadap *study case cryptographic failures*. Dengan menggunakan *keyword* "*OWASP top 10*" dan "*information technology security*" pada beberapa journal *sciencedirect*, *IEEE*, *google scholar*, dan pencarian *google search engine*. Setelah menemukan artikel yang relevan dengan penulisan dengan tema *cryptographic failures* artikel tersebut di unduh untuk diterjemahkan dan dikumpulkan data dan informasi terkait dengan *study case* yang sedang diteliti.

Tahap terakhir setelah membaca artikel yaitu tahap pemilihan paper dan artikel yang terpilih berkaitan dan relevan digunakan untuk *study case*.



Gambar 2. Tahapan Penelitian

Hasil Penelitian dan Temuan

A.Hasil Pencarian *System Literature Review*

Tabel 1. Pencarian *System Literature Review*

Jenis Ancaman Aplikasi Website	**Referensi**
SHA1, MD5	[4], [9], [14]
SHA2-512/256	[4], [7], [9]
OWASP Top 10	[4]
Cryptographic Failures	[4], [6]–[8], [11]–[13]
Teknologi Mencegah Cryptographic Failures	[4], [7]

B. *Cryptographic Failures*

Dalam artikel ini dijelaskan mengenai faktor-faktor yang menyebabkan kerentanan pada aplikasi *website* karena *cryptographic failures* dan hal apa saja yang diperlukan untuk mitigasi terhadap kerentanan *cryptographic failures*.

e-issn : 3063-4091
hal. 359-377

1. Penyebab Kerentanan

Sebelum mengetahui tentang apa saja yang menyebabkan, hal utama yang dilakukan adalah menentukan tentang kebutuhan perlindungan data saat transit dan saat tidak aktif. Contohnya seperti kata sandi, nomor kartu kredit, catatan kesehatan, informasi pribadi, dan rahasia bisnis yang memerlukan perlindungan lebih, terutama jika data tersebut termasuk dalam undang-undang privasi, misalkan Peraturan Perlindungan Data Umum (*GDPR*) uni eropa dan Peraturan Undang-undang Perlindungan Data (UU PDP) Nomor 27 Tahun 2022, untuk semua data tersebut [4], [7], [9]:

- Pada saat pengiriman data apakah dalam pengiriman data ada data yang dikirimkan dalam bentuk teks biasa? hal ini menyangkut protokol seperti *HTTP*, *SMTP*, *FTP*, yang juga menggunakan teknologi pemutakhiran *TLS* seperti *STARTTLS*. Kemudian lintas internet eksternal yang berbahaya. Verifikasi semua lalu lintas internal, misalnya antara penyeimbang beban, *server website*, atau *back-end system*.
- Masih menggunakan algoritma atau protokol kriptografi lama atau lemah yang dipakai baik secara *default* atau di dalam kode lama.
- Menggunakan kunci kriptografi lemah atau menggunakan kembali, atau tidak menerapkan manajemen atau rotasi kunci yang tepat, dan tidak memeriksa kunci kriptografi ke dalam repositori kode sumber.

- Enkripsi tidak diterapkan, misalnya tidak ada arahan atau pemberitahuan keamanan atau header *HTTP* (*browser*) yang hilang.
- Sertifikat *server* yang diterima dan rantai kepercayaan tidak diperiksa dan divalidasi dengan benar.
- Pengabaian vektor inisialisasi, menggunakan kembali, dan dibuat tidak cukup standar keamanan untuk mode operasi kriptografi. Menggunakan mode operasi yang tidak aman seperti *ECB*. Tidak menggunakan autentikasi enkripsi yang tepat.
- Menggunakan kata sandi sebagai kunci kriptografi tanpa adanya fungsi derivasi kunci dasar kata sandi.
- Menerapkan keacakan digunakan untuk tujuan kriptografi yang tidak dirancang untuk memenuhi persyaratan kriptografi yang terstandar, bahkan apabila fungsi ini digunakan, apakah pengembang perlu menyamakannya, jika tidak, apakah pengembang telah menimpa fungsionalitas penyemaian yang kuat untuk dibangun di dalamnya dengan bibit yang tidak memiliki entropi/ketidakpastian yang cukup memenuhi standar.
- Masih menggunakan teknik *MD5* atau *SHA1*, dan menerapkan fungsi hash non-kriptografi saat fungsi hash kriptografi dibutuhkan.
- Masih menerapkan metode *padding* kriptografi seperti *PKCS* nomor 1 v1.5.
- Tidak menerapkan pesan atau pemberitahuan kesalahan kriptografi atau informasi saluran samping dapat dieksploitasi, seperti dalam bentuk serangan *oracle padding*.

e-issn : 3063-4091
hal. 359-377

2. Cara Mencegah

Langkah-langkah preventif yang dapat dapat dilakukan untuk meminimalkan dampak adalah sebagai berikut [4], [9]:

- Mengklasifikasikan data yang diproses, disimpan, atau dikirimkan oleh aplikasi. Mengidentifikasi data mana yang sensitif menurut undang-undang perlindungan data dan privasi, persyaratan peraturan, atau kebutuhan bisnis.
- Jangan menyimpan data sensitif jika tidak diperlukan, membuang data tersebut segera mungkin atau menggunakan tokenisasi atau bahkan pemotongan yang sesuai dengan Payment Card Industry Data Security Standard (PCI DSS).
- Memastikan semua data sensitif yang tidak aktif di enkripsi dengan tepat.
- Memastikan algoritma, protokol, dan kunci standar yang mutakhir dan kuat tersedia, dan menggunakan manajemen kunci yang tepat.
- Enkripsikan semua data yang sedang dikirim dengan protokol aman seperti TLS dengan sandi(ciphers) forward secrecy (FS), prioritas sandi oleh server, dan parameter aman.
- Menerapkan enkripsi menggunakan arahan seperti HTTP Strict Transport Security (HSTS).
- Lakukan non-aktifkan caching untuk respon yang berisi data sensitif.

- Menerapkan kontrol keamanan yang diperlukan sesuai dengan klasifikasi data.
- Jangan menggunakan protokol lama seperti FTP(File Transfer Protocol) dan SMTP(Simple Mail Transfer Protocol) untuk mengangkut data sensitif.
- Simpanlah kata sandi menggunakan fungsi hash adaptif dan salted yang kuat dengan kerja (faktor penundaan), seperti argon2, scrypt,bcrypt, atau PBKDF2.
- Vektor inisialisasi harus dipilih sesuai dengan mode operasi. Untuk banyak mode, ini berarti menggunakan CSPRNG (cryptographically secure pseudo random number generator) merupakan generator angka acak semu yang aman secara kriptografi. Untuk mode yang memerlukan nonce, maka vektor inisialisasi (IV) tidak memerlukan CSPRNG. Dalam berbagai kasus, IV tidak boleh digunakan dua kali untuk kunci tetap.
- Selalu menggunakan enkripsi yang diautentikasi, bukan hanya enkripsi.
- Kunci harus dibuat secara acak menerapkan kriptografi dan disimpan dalam memori sebagai array byte, jika kata sandi digunakan, maka kata sandi tersebut harus diubah menjadi kata kunci melalui fungsi derivasi kunci dasar kata sandi yang sesuai.
- Memastikan bahwa keacakan kriptografi digunakan jika sesuai, dan tidak disemai dengan cara yang dapat diprediksi atau dengan entropi rendah. Sebagian besar API modern tidak mengharuskan pengembang untuk menyemai CSPRNG untuk mendapatkan keamanan.

e-issn : 3063-4091
hal. 359-377

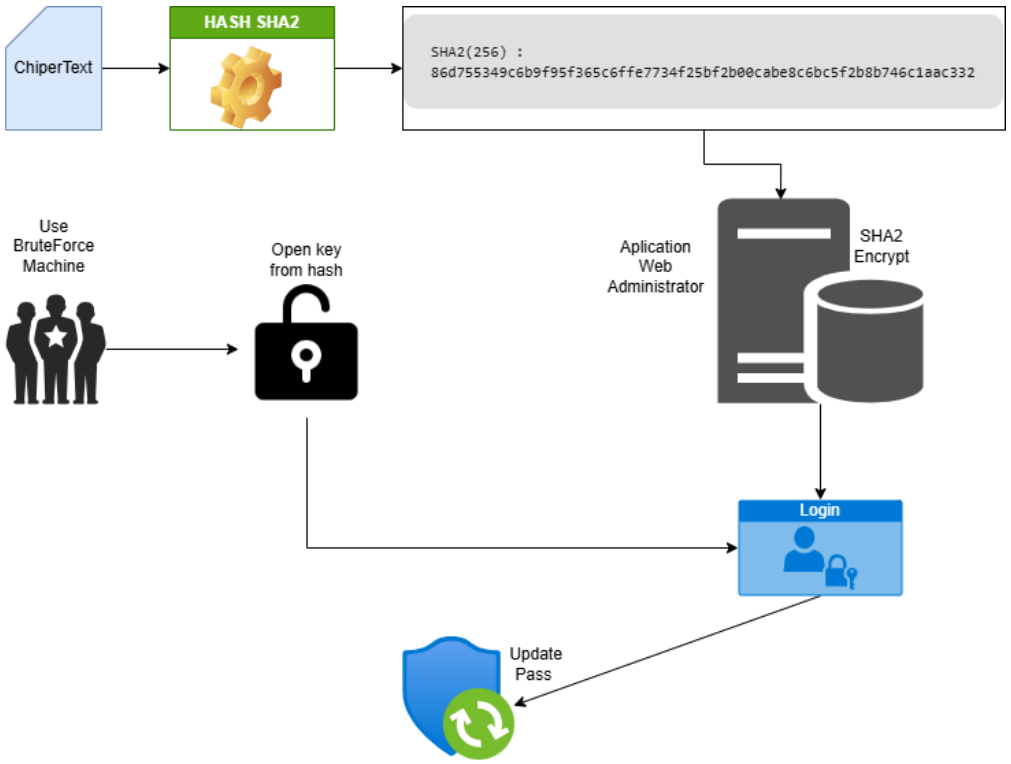
- Menghindari fungsi kriptografi dan skema padding yang sudah tidak digunakan lagi, seperti MD5, SHA1, PKCS nomor 1 v1.5.
- Memverifikasi secara independen efektivitas konfigurasi dan pengaturan.

C. Hasil Analisa

Berdasarkan pengamatan dan penelusuran mengenai mitigasi bencana cryptographic failures yang sudah dilakukan sebelumnya terdapat usulan skenario untuk mencegah terjadinya kerentanan pada cryptographic failures, skenario yang dapat dilakukan sebagai pengguna aplikasi website adalah sebagai berikut:

1. Skenario Skema Pencegahan Dalam Penggunaan Aplikasi Website

Berikut diilustrasikan pencegahan yang dapat dilakukan untuk kerentanan cryptographic failures pengguna aplikasi website, disebutkan dalam gambar 3 dibawah ini:



Gambar 3. Skenario Pencegahan Cryptographic Failure

Penjelasan singkat dari skenario pencegahan pada gambar 3 sebagai berikut:

Pada kasus pembuatan website untuk sekarang ini masih banyak yang menggunakan teknologi MD5 atau SHA1 atau masih menggunakan keduanya didalam aplikasi. Tentu hal ini sebagai seorang pengguna aplikasi, sulit mengetahui apakah website yang kita gunakan sudah menerapkan SHA2. MD5 dan SHA1 saat ini sudah harus ditinggalkan karena kodenya sudah dapat dipecahkan dengan mudah oleh penyusup, kebanyakan kasus sudah menggunakan mesin brute force untuk mendapatkan kunci atau password yang sudah di enkripsi menggunakan MD5 dan SHA1.

Pada gambar 3 dijelaskan skenario mengenai pencegahan yang bisa dilakukan sebagai pengguna aplikasi website, yaitu dengan minimal kita melakukan login

pada aplikasi dan merubah password setidaknya 3(tiga) bulan sekali untuk mencegah kerentanan. Pada gambar 3 disebutkan kunci yang sudah di enkripsi menggunakan SHA2-*512/256* disimpan didalam aplikasi untuk digunakan sebagai autentikasi pengguna website ketika akan login dan melakukan transaksi. Kemudian dengan melakukan login dan merubah password. Kenapa hal ini disarankan pada gambar 3, karena pada praktiknya aplikasi website sekarang masih banyak yang menggunakan metode enkripsi MD5 dan SHA1 yang sudah ditinggalkan seharusnya. Ketika kita melakukan login pada aplikasi website itu berarti lebih memudahkan pengembang aplikasi melakukan update hash dengan metode enkripsi yang terbaru. Sulit jadinya apabila aplikasi website sudah melakukan update dengan metode SHA2 namun dari pengguna aplikasi website tidak melakukan login sehingga tidak dapat dilakukan autentikasi oleh pengembang, yang dimana autentikasi ini akan digunakan untuk melakukan update hash yang disimpan didalam database kunci login didalam aplikasi. Salah satu faktor mengapa update enkripsi SHA2 ini masih sulit dilakukan karena tidak ada autentikasi dari pengguna.

Di Dalam gambar 3 disebutkan bahwa ada skenario yang akan menggantikan login pengguna asli yaitu aktor brute force. Saat ini sudah banyak digunakan mesin brute force untuk menemukan kunci dari hash, ketika kita akan melakukan login dibutuhkan username (nickname atau email) dan kunci/password untuk masuk ke dalam aplikasi website. Aktor brute force sudah memiliki password/nickname dan hash kunci dan password namun memang tidak bisa digunakan untuk login, yang dibutuhkan untuk login adalah

plaintext dari kunci, yang dimana aktor akan mencoba mendapatkan kunci dengan berbagai cara dari mesin deteksi. Setidaknya akan dicoba semua karakter, huruf, dan angka yang digunakan. Tentu hal ini meski menggunakan mesin dibutuhkan waktu sampai menemukan kunci yang tepat sesuai dengan yang dimiliki pemilik akun yang sah/sesungguhnya. Gambar 4 mengilustrasikan mesin yang mencari kunci dengan mesin brute force sebagai berikut:

Gambar 4. Ilustrasi menemukan password menggunakan mesin brute force

Pada saat pengguna sah/sesungguhnya melakukan login dan merubah password setiap maksimal 3 bulan sekali, akan mencegah ketika mesin sudah mencapai password yang sesungguhnya, tetapi pengguna sudah mengganti kunci/password yang baru. Dengan begitu mesin brute force akan mengulang lagi dari awal dan seterusnya. Jika pengguna tidak merubah password secara berkala setiap 3(tiga) bulan sekali akan menjadi celah mesin brute force menemukan kunci sah/sesungguhnya dari mendeskripsikan hash password, dan

akun login pengguna beralih kepada aktor dan dengan tanpa pengetahuan pengguna sah/sesungguhnya aktor menguasai akun.

Dalam skenario jika masih menggunakan MD5 dan SHA1 akan mudah ditemukan kunci duplikat dari kunci yang sah/sesungguhnya. Pada gambar 5 dibawah ini akan diperlihatkan dimana kunci asli dan kunci bilangan heksa berbeda namun memiliki kunci hash enkripsi MD5 yang sama, yaitu sebagai berikut:

Gambar 5. Ilustrasi bilangan heksa berbeda namun memiliki Hash MD5 sama

Dimana heksa yang pertama dan kedua perbedaannya diberi warna merah, ketika data tersebut di enkripsi menggunakan MD5 memiliki nilai hash yang sama yaitu 79054025255fb1a26e4bc422aef54eb4. Penelitian ini dilakukan pada tahun Xiaoyun Wang and Hongbo Yu of Shandong University in China [14].

2. Hasil Kinerja Skenario

Pada beberapa percobaan dengan melakukan skenario pencegahan yang dapat dilakukan pengguna aplikasi website ini dapat mengurangi kerentanan yang ada didalam cryptographic failures. Salah satunya yang menjadi acuan yaitu pada aturan

Sistem Manajemen Keamanan Informasi ISO/IEC 27001 diwajibkan untuk melakukan perubahan kunci/password pada aplikasi dan website minimal 3(tiga) bulan sekali, dengan tingkat wajib. Tentunya aturan tersebut juga tidak diatur tanpa adanya referensi kejadian berdasarkan kerentanan aplikasi dan website yang sudah terjadi sebelumnya.

Pada kasus setiap harinya pengguna aplikasi dan website contoh seperti Google Mail dan Yahoo Mail dan lainnya, ketika dalam jangka waktu tertentu pengguna tidak melakukan login akan diberi notifikasi kepada email pemulihan untuk mengingatkan pengguna melakukan login kedalam aplikasi dan website. Hal ini tentu tidak dilakukan dengan tanpa ada alasan. Salah satu langkah untuk melakukan pembaharuan teknologi keamanan di sisi enkripsi kunci dan password yang menjadi hash dan disimpan di database aplikasi website mempermudah pengembang melakukan update enkripsi ke metode hash terbaru dan termutakhir.

Pada kasus dimana pemilik aplikasi website perlu pemahaman yang baik mengenai serangan yang ada dan dapat melihat rujukan pada artikel yang diterbitkan oleh OWASP, dan tentunya terkait kerentanan cryptographic failures, untuk saat ini yang disarankan untuk melakukan enkripsi menggunakan metode Hashing SHA2.

KESIMPULAN

Cryptographic failures adalah kerentanan yang masuk dalam daftar top 10 OWASP dan menempati posisi ke-2. Daftar OWASP merilis serangan yang paling sering terjadi didunia dan dirangkum 10 terbesar dalam rentang waktu 2017-2021, dan akan dilakukan 5 tahun sekali. Hal ini menunjukan banyaknya jumlah serangan yang muncul pada aplikasi website di Indonesia. Tentunya cryptographic failures

perlu dijaga untuk setiap pemilik aplikasi website karena dapat menimbulkan risiko berdampak sangat merugikan yaitu sensitive data exposure. Di Sisi pengguna perlu adanya mitigasi untuk mengatasi dampaknya itu dengan melakukan login aplikasi website secara berkala untuk memastikan, dan tentunya dari sisi pengembang aplikasi dapat terbantu untuk melakukan update enkripsi hash kunci/password terbarukan dengan metode hash terbaru.

Harapan kedepannya dengan adanya pengetahuan umum dari artikel kerentanan cryptographic failures dapat memberi dampak positif bagi pemilik dan pengguna aplikasi website untuk lebih waspada dan lebih bijak pada pengelolaan kunci/password. Diharapkan akan dapat dilakukan penelitian lebih mendalam untuk membuat model framework dan pedoman lebih detail terkait dengan resep dalam teknik enkripsi untuk mencegah kerentanan lainnya yang ada dalam daftar Top 10 OWASP, serta memberi dampak positif pada aplikasi website di Indonesia.

REFERENCES

[1] T. Redaksi, L. Tni, P. Drs, N. Sulistyo, and M. Han, “Laporan tahunan honeynet bssn 2024,” 2024.

[2] C. N. Ganesh, S. Ahamad, V. Veeraiah, T. K. O, S. P. Patil, and A. Namdev, “Quantum Computing: The Future of Secure Data Encryption and Problem Solving,” *2024 IEEE 11th Uttar Pradesh Sect. Int. Conf. Electr. Electron. Comput. Eng.*, pp. 1–6, Nov. 2024, doi: 10.1109/UPCON62832.2024.10983883.

[3] A. Majeed and S. O. Hwang, “When AI Meets Information Privacy: The Adversarial Role of AI in Data Sharing Scenario,” *IEEE Access*, vol. 11, pp. 76177–76195, 2023, doi: 10.1109/ACCESS.2023.3297646.

[4] “A02 Cryptographic Failures - OWASP Top 10:2021.” https://owasp.org/Top10/A02_2021-Cryptographic_Failures/ (accessed May 30, 2025).

[5] “Posisi Firma Hukum dalam Kasus Panama Papers - LK2 FHUI.”

https://lk2fhui.law.ui.ac.id/posisi-firma-hukum-dalam-kasus-panama-papers/ (accessed May 30, 2025).

[6] R. J. Kikani, K. Verma, R. Navalakhe, G. Shrivastava, and V. Shrivastava, "Cryptography: Recent research trends of encrypting mathematics," *Mater. Today Proc.*, vol. 56, pp. 3247–3253, Jan. 2022, doi: 10.1016/J.MATPR.2021.09.378.

[7] P. Nannipieri *et al.*, "SHA2 and SHA-3 accelerator design in a 7 nm technology within the European Processor Initiative," *Microprocess. Microsyst.*, vol. 87, p. 103444, Nov. 2021, doi: 10.1016/J.MICPRO.2020.103444.

[8] G. Markowsky, "Was the 2006 Debian SSL Debacle a system accident?," *Proc. 2013 IEEE 7th Int. Conf. Intell. Data Acquis. Adv. Comput. Syst. IDAACS 2013*, vol. 2, pp. 624–629, 2013, doi: 10.1109/IDAACS.2013.6663000.

[9] "Difference Between SHA1 and SHA2 | GeeksforGeeks." https://www.geeksforgeeks.org/difference-between-sha1-and-sha2/ (accessed May 30, 2025).

[10] Y. Chen and Y. Zhao, "Sign-then-encrypt with security enhancement and compressed ciphertext," *Theor. Comput. Sci.*, vol. 1027, p. 115006, Feb. 2025, doi: 10.1016/J.TCS.2024.115006.

[11] Y. Zhu, H. Li, J. Cui, and Y. Ma, "Verifiable Subgraph Matching with Cryptographic Accumulators in Cloud Computing," *IEEE Access*, vol. 7, pp. 169636–169645, 2019, doi: 10.1109/ACCESS.2019.2955243.

[12] V. Grozov, M. Budko, A. Guirik, and M. Budko, "Efficiency Comparison of Pseudorandom Number Generators Based on Strong Cryptographic Algorithms," *Int. Congr. Ultra Mod. Telecommun. Control Syst. Work.*, vol. 2018-November, Jul. 2018, doi: 10.1109/ICUMT.2018.8631223.

[13] M. Althamir, A. Alabdulhay, and M. M. Yasin, "A Systematic Literature Review on Symmetric and Asymmetric Encryption Comparison Key Size," *Proc. - 2023 3rd Int. Conf. Smart Data Intell. ICSMDI 2023*, pp. 110–117, 2023, doi: 10.1109/ICSMDI57622.2023.00027.

[14] "Peter Selinger: MD5 Collision Demo." https://www.mscs.dal.ca/~selinger/md5collision/ (accessed May 30, 2025).