Polygon: Jurnal Ilmu Komputer dan Ilmu Pengetahuan Alam
Volume. 4 Nomor. 1 Januari 2026
e-ISSN: 3046-5419; p-ISSN: 3032-6249, Hal 65-73
DOI: https://doi.org/10.62383/polygon.v4i1.926
Tersedia: https://journal.arimsi.or.id/index.php/Polygon

Perbandingan Waktu Pemecahan *Password* Menggunakan Algoritma *Hash* MD5, SHA-256, dan SHA-512 pada Serangan *Brute Force*

Nur Bainatun Nisa[1*], Noni Fauzia Rahmadani[2], Aulia Kartika Dewi[3], Luftia Rahma Nasution[4], Dzilhulaifa Siregara[5], Rifdah Syahputri[6], Ibnu Rusydi[7]
[1-7]Program Studi Ilmu Komputer, Universitas Islam Negeri Sumatera, Indonesia
**Penulis Korespondensi : nurbainatunnisa@gmail.com*

Abstract. *Password security is a critical component in protecting information systems, as passwords are often the primary target of various attacks, particularly brute force attacks. A brute force attack works by systematically attempting all possible character combinations until the correct password corresponding to a stored hash value is found. Therefore, the choice of an appropriate hash algorithm plays a significant role in determining a system's resistance to such attacks. This study aims to analyze and compare the password cracking time of MD5, SHA-256, and SHA-512 hash algorithms under brute force attack scenarios. The research methodology involves generating hash values from a set of test passwords using each hash algorithm, followed by conducting brute force attacks to recover the original passwords based on the generated hash values. The collected data are analyzed by measuring the time required to crack passwords for each algorithm. The results indicate that MD5 has the fastest cracking time compared to SHA-256 and SHA-512, indicating a lower level of resistance to brute force attacks. SHA-256 demonstrates better security than MD5 but remains less resistant when compared to SHA-512. The SHA-512 algorithm requires the longest cracking time, reflecting the highest level of resistance to brute force attacks among the tested algorithms. In conclusion, hash algorithms with larger bit lengths provide stronger protection against brute force attacks and are more suitable for secure password storage in information systems.*

Keywords: *Brute Force; Information Security; MD5; Password Hash; SHA-256.*

Abstrak. Keamanan *password* merupakan komponen penting dalam perlindungan sistem informasi karena *password* sering menjadi target utama berbagai jenis serangan, salah satunya adalah serangan *Brute Force*. Serangan *Brute Force* dilakukan dengan mencoba seluruh kemungkinan kombinasi karakter hingga menemukan *password* yang sesuai dengan nilai *hash* yang tersimpan. Oleh karena itu, pemilihan algoritma *hash* yang tepat sangat berpengaruh terhadap tingkat ketahanan sistem terhadap serangan tersebut. Penelitian ini bertujuan untuk menganalisis dan membandingkan waktu pemecahan *password* menggunakan algoritma *hash* MD5, SHA-256, dan SHA-512 pada skenario serangan *Brute Force*. Metode penelitian dilakukan dengan menghasilkan nilai *hash* dari sejumlah *password* uji menggunakan masing-masing algoritma *hash*, kemudian dilakukan proses *Brute Force* untuk menemukan kembali *password* asli berdasarkan nilai *hash* tersebut. Data yang diperoleh dianalisis dengan mengukur waktu yang dibutuhkan untuk memecahkan *password* pada setiap algoritma. Hasil penelitian menunjukkan bahwa algoritma MD5 memiliki waktu pemecahan paling cepat dibandingkan SHA-256 dan SHA-512, yang menandakan tingkat ketahanan yang lebih rendah terhadap serangan *Brute Force*. SHA-256 menunjukkan tingkat keamanan yang lebih baik dibandingkan MD5, namun masih lebih rentan jika dibandingkan dengan SHA-512. Algoritma SHA-512 membutuhkan waktu pemecahan paling lama, sehingga menunjukkan tingkat ketahanan tertinggi terhadap serangan *Brute Force*. Kesimpulan dari penelitian ini adalah bahwa algoritma *hash* dengan panjang bit yang lebih besar memberikan perlindungan yang lebih kuat terhadap serangan *Brute Force* dan lebih direkomendasikan untuk digunakan dalam sistem penyimpanan *password*.

Kata kunci: *Brute Force; Hash Password*; Keamanan Informasi; MD5; SHA-256.

1. LATAR BELAKANG

Keamanan *password* merupakan aspek fundamental dalam menjaga kerahasiaan dan integritas data pada sistem informasi *modern*. *Password* masih menjadi mekanisme autentikasi yang paling banyak digunakan dalam berbagai aplikasi, mulai dari sistem akademik, layanan perbankan, hingga *platform* media sosial. Namun, penggunaan *password* yang lemah serta praktik penyimpanan yang kurang aman masih sering ditemukan pada berbagai sistem

Naskah Masuk: 23 November 2025; Revisi: 21 Desember 2025; Diterima: 18 Januari 2026; Terbit: 23 Januari 2026

informasi (Fachri, 2023). Kondisi ini meningkatkan risiko keberhasilan serangan terhadap akun pengguna, terutama ketika data hash *password* berhasil diakses oleh pihak yang tidak berwenang. Untuk mengurangi risiko kebocoran data, *password* umumnya disimpan dalam bentuk nilai *hash* menggunakan algoritma kriptografi satu arah. Meskipun demikian, meningkatnya kemampuan komputasi dan ketersediaan perangkat lunak peretas menyebabkan teknik serangan terhadap *password* semakin berkembang, salah satunya melalui serangan *brute force* yang mencoba seluruh kemungkinan kombinasi karakter hingga menemukan *password* yang sesuai (Simangunsong et al., 2025).

Berbagai algoritma *hash* telah dikembangkan dan digunakan secara luas untuk pengamanan *password*, di antaranya MD5, SHA-256, dan SHA-512. MD5 merupakan algoritma *hash* generasi awal yang memiliki kecepatan tinggi namun panjang *bit* yang relatif pendek, sedangkan SHA-256 dan SHA-512 merupakan bagian dari keluarga *Secure Hash Algorithm* yang menawarkan tingkat kompleksitas dan panjang *bit* yang lebih besar. Sejumlah penelitian sebelumnya menunjukkan bahwa MD5 memiliki berbagai kelemahan dan lebih rentan terhadap serangan kriptografi, sementara algoritma SHA dengan panjang *bit* lebih besar dinilai lebih aman (Santoso, 2021). Namun, sebagian besar kajian terdahulu lebih menekankan pada aspek teoritis atau jenis serangan tertentu, seperti *collision attack*, tanpa memberikan perbandingan eksperimental yang jelas terkait waktu pemecahan *password* menggunakan serangan *brute force* (Dafi et al., 2025).

Keterbatasan penelitian sebelumnya tersebut menunjukkan adanya celah penelitian yang perlu dikaji lebih lanjut, khususnya dalam konteks analisis empiris terhadap ketahanan algoritma *hash* populer menggunakan pendekatan *brute force* (Sitorus et al., 2024). Perbandingan waktu pemecahan *password* menjadi indikator penting untuk menilai seberapa efektif suatu algoritma *hash* dalam menahan serangan tersebut. Oleh karena itu, penelitian ini memiliki urgensi untuk memberikan gambaran nyata mengenai perbedaan tingkat ketahanan antara MD5, SHA-256, dan SHA-512 berdasarkan hasil pengujian langsung, sehingga dapat menjadi referensi praktis bagi pengembang sistem (Kurniawan et al., 2022).

Berdasarkan latar belakang tersebut, tujuan dari penelitian ini adalah untuk menganalisis dan membandingkan waktu pemecahan *password* menggunakan algoritma *hash* MD5, SHA-256, dan SHA-512 pada skenario serangan *brute force*. Hasil penelitian ini diharapkan dapat memberikan kontribusi dalam pemilihan algoritma *hash* yang lebih aman untuk penyimpanan *password* serta memperkaya kajian empiris di bidang keamanan informasi.

2. KAJIAN TEORITIS

Password merupakan mekanisme autentikasi yang paling umum digunakan dalam sistem informasi untuk memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses sumber daya tertentu (Mardhatillah et al., 2025). Meskipun telah banyak dikembangkan metode autentikasi alternatif, *password* masih menjadi pilihan utama karena kemudahan implementasi dan penggunaannya. Namun, penggunaan *password* yang lemah serta praktik pengelolaan *password* yang kurang tepat dapat meningkatkan risiko terjadinya pelanggaran keamanan. Oleh karena itu, diperlukan mekanisme tambahan untuk melindungi *password* agar tidak mudah disalahgunakan apabila terjadi kebocoran data (Marcius et al., 2024).

Algoritma *Hash* Kriptografi

Algoritma *hash* kriptografi adalah fungsi satu arah yang digunakan untuk mengubah data masukan, seperti *password*, menjadi representasi nilai hash dengan panjang tetap. Sifat satu arah dari algoritma *hash* membuat nilai *hash* tidak dapat dikembalikan secara langsung menjadi data aslinya. Selain itu, algoritma *hash* yang baik dirancang untuk menghasilkan nilai hash yang unik untuk setiap masukan yang berbeda dan memiliki tingkat kompleksitas komputasi tertentu. Dalam konteks keamanan *password*, algoritma *hash* berperan penting dalam mencegah penyerang memperoleh *password* asli meskipun berhasil mengakses basis data penyimpanan *hash* (Arisandi et al., 2025).

Penyimpanan *password* menggunakan algoritma *hash* bertujuan untuk mengurangi risiko kebocoran *password* dalam bentuk teks asli. Pada proses ini, *password* pengguna akan di-*hash* sebelum disimpan di dalam basis data. Ketika pengguna melakukan autentikasi, sistem akan membandingkan nilai hash dari *password* yang dimasukkan dengan nilai *hash* yang tersimpan. Mekanisme ini memberikan perlindungan dasar terhadap *password*, namun tetap memiliki keterbatasan apabila algoritma *hash* yang digunakan memiliki kecepatan komputasi tinggi dan tidak dilengkapi dengan mekanisme tambahan seperti *salt* (Hutagalung et al., 2023).

Serangan *Brute Force*

Serangan *brute force* merupakan teknik serangan yang dilakukan dengan mencoba seluruh kemungkinan kombinasi karakter secara sistematis hingga menemukan *password* yang sesuai. Serangan ini tidak memanfaatkan kelemahan tertentu pada algoritma *hash*, melainkan mengandalkan jumlah percobaan yang besar dan kecepatan komputasi. Keberhasilan serangan *brute force* sangat dipengaruhi oleh kompleksitas *password* dan algoritma *hash* yang digunakan. Algoritma *hash* yang cepat memungkinkan penyerang melakukan lebih banyak

percobaan dalam waktu singkat, sehingga meningkatkan peluang keberhasilan serangan (Wijaya et al., 2021).

Algoritma *Hash* MD5

MD5 adalah algoritma *hash* kriptografi yang menghasilkan nilai *hash* sepanjang 128 *bit* dan dirancang untuk kecepatan pemrosesan yang tinggi. Pada awalnya, MD5 banyak digunakan dalam berbagai aplikasi keamanan. Namun, seiring perkembangan penelitian kriptografi, MD5 diketahui memiliki berbagai kelemahan, terutama terkait ketahanan terhadap serangan *collision* dan *brute force*. Kecepatan komputasi MD5 yang tinggi justru menjadi kelemahan dalam konteks keamanan password, karena memungkinkan penyerang melakukan percobaan hash dalam jumlah besar dalam waktu yang relatif singkat (Fachri, 2023).

Algoritma *Hash* SHA-256

SHA-256 merupakan algoritma *hash* yang termasuk dalam keluarga *Secure Hash Algorithm* dan menghasilkan nilai hash sepanjang 256 *bit*. Algoritma ini dirancang untuk memberikan tingkat keamanan yang lebih tinggi dibandingkan MD5 dengan meningkatkan kompleksitas komputasi. SHA-256 banyak digunakan dalam sistem keamanan *modern*, termasuk pada aplikasi web dan teknologi *blockchain*. Dengan tingkat kompleksitas yang lebih tinggi, SHA-256 memberikan ketahanan yang lebih baik terhadap serangan *brute force* dibandingkan algoritma *hash* generasi lama (Dafi et al., 2025; Santoso, 2021).

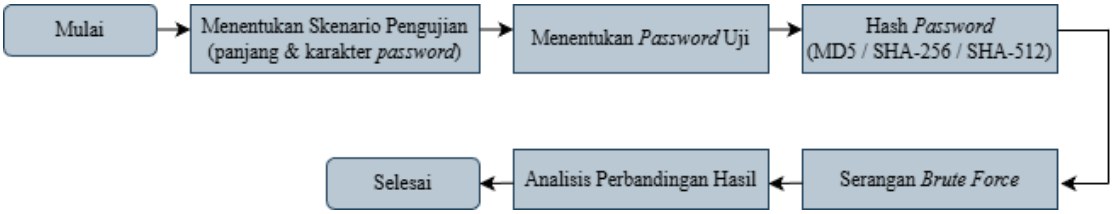
Algoritma *Hash* SHA-512

SHA-512 merupakan varian *Secure Hash Algorithm* dengan panjang *hash* sebesar 512 *bit* yang menawarkan tingkat keamanan lebih tinggi dibandingkan SHA-256. Algoritma ini dirancang untuk menghadapi ancaman serangan kriptografi yang semakin kompleks dengan meningkatkan ruang kemungkinan hash dan beban komputasi (Andilala et al., 2023). Dalam konteks serangan *brute force*, SHA-512 membutuhkan waktu pemrosesan yang lebih lama untuk setiap percobaan *hash*, sehingga secara signifikan memperlambat proses pemecahan *password* dan meningkatkan ketahanan sistem terhadap serangan (Fajrin & Baharuddin, 2025). Beberapa penelitian terdahulu telah membahas perbandingan tingkat keamanan berbagai algoritma *hash* terhadap serangan kriptografi. Hasil penelitian tersebut umumnya menunjukkan bahwa algoritma hash dengan panjang *bit* yang lebih besar memiliki tingkat ketahanan yang lebih baik terhadap serangan *brute force* (Nurjaman & Turnip, 2024). Selain itu, penelitian sebelumnya juga menyarankan untuk tidak lagi menggunakan MD5 sebagai algoritma pengamanan *password* pada sistem modern. Meskipun demikian, masih terbatas penelitian yang secara khusus membandingkan waktu pemecahan *password* menggunakan serangan

brute force pada MD5, SHA-256, dan SHA-512 dalam satu kerangka eksperimen yang sama (Setiadi et al., 2025).

3. METODE PENELITIAN

Penelitian ini menggunakan metode eksperimen komputasi untuk membandingkan waktu pemecahan *password* pada algoritma *hash* MD5, SHA-256, dan SHA-512 menggunakan serangan *brute force*. Metode ini dipilih karena memungkinkan pengujian langsung terhadap performa masing-masing algoritma *hash* dalam lingkungan yang terkontrol, sehingga hasil yang diperoleh dapat dianalisis secara objektif dan terukur.



Gambar 1. Tahapan Penelitian.

Tahap awal penelitian dimulai dengan penentuan skenario pengujian, yang meliputi pemilihan karakter *password* dan panjang *password* yang akan diuji. *Password* uji ditentukan secara terkontrol agar setiap algoritma *hash* diuji menggunakan input yang sama. Selanjutnya, *password* tersebut diproses menggunakan algoritma *hash* MD5, SHA-256, dan SHA-512 untuk menghasilkan nilai *hash* yang akan menjadi target dalam proses serangan *brute force*.

Proses pengumpulan data dilakukan dengan menjalankan serangan *brute force* terhadap setiap nilai *hash* yang dihasilkan. Serangan *brute force* dilakukan dengan mencoba kombinasi karakter secara sistematis hingga ditemukan *password* yang sesuai dengan nilai *hash* target. Seluruh proses pengujian dijalankan pada lingkungan perangkat keras dan perangkat lunak yang sama untuk menjaga konsistensi hasil. Data yang dikumpulkan berupa waktu yang dibutuhkan untuk berhasil memecahkan *password* pada masing-masing algoritma *hash*.

Analisis data dilakukan dengan menggunakan analisis deskriptif untuk membandingkan waktu pemecahan *password* antar algoritma *hash*. Hasil pengujian disajikan dalam bentuk tabel dan grafik untuk menunjukkan perbedaan performa secara jelas. Validitas hasil dijaga dengan memastikan bahwa setiap pengujian dilakukan menggunakan konfigurasi dan parameter yang sama, sedangkan reliabilitas ditunjukkan melalui konsistensi hasil waktu pemecahan pada pengujian yang dilakukan secara berulang.

Model penelitian dalam penelitian ini menggambarkan hubungan antara jenis algoritma *hash* sebagai variabel independen dan waktu pemecahan *password* sebagai variabel dependen. Algoritma *hash* MD5, SHA-256, dan SHA-512 diuji dalam skenario serangan *brute force* yang sama untuk melihat pengaruh perbedaan karakteristik algoritma terhadap waktu pemecahan *password*.

4. HASIL DAN PEMBAHASAN

Pengujian dilakukan dengan menerapkan serangan *brute force* terhadap *password* yang telah di-*hash* menggunakan algoritma MD5, SHA-256, dan SHA-512. Proses pengujian dilakukan pada lingkungan komputasi yang sama untuk memastikan konsistensi hasil. *Password* yang diuji terdiri dari beberapa variasi panjang dan kompleksitas karakter, yaitu abc, a1b2, dan abc123. Parameter utama yang diamati adalah waktu yang dibutuhkan untuk menemukan *password* asli dari nilai *hash* yang dihasilkan.

Hasil pengujian waktu pemecahan *password* disajikan dalam bentuk tabel untuk memudahkan proses analisis dan perbandingan antar algoritma *hash*. Tabel 1. menunjukkan waktu pemecahan *password* untuk setiap algoritma *hash* berdasarkan variasi *password* yang diuji.

Tabel 1. Waktu Pemecahan *Password* Menggunakan Serangan *Brute Force*.

Password	**MD5 (detik)**	**SHA-256 (detik)**	**SHA-512 (detik)**
abc	0.0014	0.0018	0.0018
a1b2	0.0893	0.0949	0.1296
abc123	69.0437	84.4074	93.0641

Berdasarkan Tabel 1. terlihat bahwa waktu pemecahan *password* meningkat seiring dengan bertambahnya panjang dan kompleksitas *password*. Selain itu, terdapat perbedaan waktu pemecahan yang cukup signifikan antara algoritma hash MD5, SHA-256, dan SHA-512. Untuk memperjelas tren ketahanan masing-masing algoritma *hash*, dilakukan perhitungan rata-rata waktu pemecahan *password* dari seluruh skenario pengujian. Ringkasan hasil tersebut disajikan pada Tabel 2.

Tabel 2. Rata-rata Waktu Pemecahan *Password.*

Algoritma *Hash*	**Rata-rata Waktu (detik)**
MD5	23.0448
SHA-256	28.1680
SHA-512	31.0652

Tabel 2. menunjukkan bahwa algoritma SHA-512 memiliki rata-rata waktu pemecahan paling tinggi, diikuti oleh SHA-256 dan MD5. Hasil ini mengindikasikan bahwa semakin besar kompleksitas algoritma *hash*, semakin lama waktu yang dibutuhkan untuk melakukan serangan *brute force*.

Analisis Perbandingan Algoritma Hash

Hasil pengujian menunjukkan bahwa password dengan panjang pendek dan karakter sederhana, seperti *abc*, dapat dipecahkan dalam waktu yang sangat singkat oleh ketiga algoritma hash. Hal ini menunjukkan bahwa penggunaan *password* sederhana tetap memiliki risiko keamanan yang tinggi, meskipun telah dilindungi menggunakan algoritma *hash* yang relatif kuat. Ketika kompleksitas *password* meningkat, seperti pada *password* a1b2 dan abc123, waktu pemecahan meningkat secara signifikan, terutama pada algoritma *hash* dengan kompleksitas komputasi yang lebih tinggi.

Algoritma MD5 secara konsisten menunjukkan waktu pemecahan paling cepat pada seluruh skenario pengujian. Hal ini disebabkan oleh karakteristik MD5 yang memiliki kecepatan komputasi tinggi dan panjang *hash* yang lebih pendek. SHA-256 membutuhkan waktu pemecahan yang lebih lama dibandingkan MD5 karena memiliki kompleksitas dan panjang hash yang lebih besar. Sementara itu, SHA-512 menunjukkan waktu pemecahan paling lama, terutama pada *password* dengan panjang dan kompleksitas lebih tinggi, seperti **abc123**. Temuan ini menunjukkan bahwa peningkatan panjang *bit* pada algoritma *hash* berbanding lurus dengan peningkatan ketahanan terhadap serangan *brute force*.

Hasil penelitian ini sejalan dengan konsep dasar kriptografi yang menyatakan bahwa algoritma *hash* dengan panjang *bit* dan kompleksitas komputasi yang lebih besar memiliki ketahanan yang lebih baik terhadap serangan *brute force.* Temuan ini juga konsisten dengan hasil penelitian sebelumnya yang menyimpulkan bahwa algoritma *hash* generasi lama seperti MD5 tidak lagi direkomendasikan untuk pengamanan *password* karena lebih rentan terhadap berbagai jenis serangan. Tidak ditemukan hasil yang bertentangan dengan penelitian terdahulu, melainkan penelitian ini memperkuat temuan sebelumnya melalui pendekatan eksperimental berbasis waktu pemecahan password.

Penelitian ini memberikan bukti empiris bahwa perbedaan karakteristik algoritma *hash* berpengaruh signifikan terhadap waktu pemecahan *password* pada serangan *brute force*. Secara praktis, hasil penelitian ini menunjukkan bahwa penggunaan algoritma *hash* dengan panjang *bit* lebih besar, seperti SHA-512, lebih disarankan untuk meningkatkan keamanan penyimpanan *password*. Selain itu, hasil ini juga menegaskan pentingnya penggunaan

password yang panjang dan kompleks sebagai lapisan keamanan tambahan dalam sistem autentikasi.

5. KESIMPULAN DAN SARAN

Penelitian ini menyimpulkan bahwa algoritma *hash* MD5 memiliki waktu pemecahan paling cepat dibandingkan SHA-256 dan SHA-512 pada seluruh variasi *password* yang diuji, sedangkan SHA-512 menunjukkan ketahanan tertinggi terhadap serangan brute force dengan waktu pemecahan paling lama, terutama pada *password* yang lebih kompleks. Hasil ini menunjukkan bahwa tingkat kompleksitas algoritma *hash* berpengaruh signifikan terhadap ketahanan sistem keamanan *password*, sehingga tujuan penelitian dapat tercapai.

Namun, hasil penelitian ini terbatas pada skenario pengujian tertentu dengan variasi *password* dan lingkungan komputasi yang terbatas, sehingga generalisasi hasil perlu dilakukan secara hati-hati. Penelitian selanjutnya disarankan untuk melibatkan variasi *password* yang lebih luas, penggunaan teknik pengamanan tambahan seperti salt atau algoritma *hashing* khusus *password*, serta pengujian pada lingkungan komputasi yang berbeda untuk memperoleh hasil yang lebih komprehensif.

DAFTAR REFERENSI

Andilala, Hidayah, A. K., Mahfuzy, A. R. W., & Oki, M. (2023). Implementasi kombinasi enkripsi Base64 dengan hashing SHA-1 dan MD5 pada aplikasi perpustakaan Universitas Muhammadiyah Bengkulu. *Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD*, *6*, 694–703. https://doi.org/10.53513/jsk.v6i2.8546

Arisandi, D., Hartati, S., & Vrabora, G. (2025). Otentikasi dua faktor menggunakan TOTP dengan SHA-512 untuk sistem pemilihan presiden mahasiswa. *EXPLORER Journal of Computer Science and Information Technology*, *5*(1), 14–25. https://doi.org/10.47065/explorer.v5i1.17

Dafi, R., Azhar, A., & Widiati, I. S. (2025). Evaluasi keamanan penyimpanan password menggunakan algoritma hash: MD5, SHA-1, dan Bcrypt. *Seminar Nasional Teknologi Informasi Dan Bisnis (SENATIB) 2025*, 1302–1305. https://doi.org/10.47701/q885nj69

Fachri, F. (2023). Optimasi keamanan web server terhadap serangan brute-force menggunakan penetration testing. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIIK)*, *10*(1), 51–58. https://doi.org/10.25126/jtiik.2023105872

Fajrin, A. M., & Baharuddin, F. (2025). Analisis performa algoritma BLAKE2b dan SHA-256 pada implementasi blockchain. *KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)*, *6*(2), 451–460. https://doi.org/https://tunasbangsa.ac.id/pkm/index.php/kesatria/article/view/588

Hutagalung, J., Ramadhan, P. S., Sihombing, S. J., & Korespondensi, P. (2023). Keamanan data menggunakan secure hashing algorithm (SHA)-256 dan Rivest Shamir Adleman

(RSA) pada digital signature. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIIK)*, *10*(6). https://doi.org/10.25126/jtiik.2023107319

Kurniawan, F., Kusyanti, A., & Nurwarsito, H. (2022). Analisis dan implementasi algoritma SHA-1 dan SHA-3 pada sistem autentikasi Garuda Training Cost. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *1*(9), 803–812.

Marcius, B., Wijaya, E., & Sutanto, F. A. (2024). Implementasi sistem autentikasi JSON Web Token pada aplikasi Fieldrent menggunakan algoritme SHA-512. *Progresif: Jurnal Ilmiah Komputer*, *19*(2), 901–911. https://doi.org/10.35889/progresif.v19i2.1367

Mardhatillah, S., Sriwinar, & Armiady, D. (2025). Optimasi algoritma SHA-256 dan metode salt untuk pengamanan akun calon santri baru pesantren Almuslim. *Jurnal Ilmu Komputer Aceh*, *2*, 1–7. https://doi.org/https://jurnal.fikompublisher.com/ilka/article/view/22

Nurjaman, A. R., & Turnip, A. T. (2024). Kombinasi algoritma kriptografi AES-256 dan SHA3-512 untuk meningkatkan keamanan dokumen PDF. *JITTER - Jurnal Ilmiah Teknologi Dan Komputer*, *11*(1), 46–54. https://doi.org/10.33197/jitter.vol11.iss1.2024.2346

Santoso, M. H. (2021). Perbandingan algoritma kriptografi hash MD5 dan SHA-1. *SEMANTIKA Prosiding Seminar Nasional Teknologi Informatika*, *2*, 54–59.

Setiadi, I., Widianti, S., & Kayuan, I. P. P. (2025). Implementasi kriptografi pengamanan data soal ujian di lingkungan. *Jurnal Penelitian Rumpun Ilmu Teknik*. https://doi.org/10.55606/juprit.v3i4.4569

Simangunsong, V., Hutasoit, Y. R., & Siallagan, D. (2025). Analisis terhadap keamanan password menggunakan hash SHA-256. *Jurnal Quancom*, *3*(1). https://doi.org/10.62375/jqc.v3i1.431

Sitorus, N., Sharon, J., Sinaga, G., & Samosir, S. L. (2024). Analisis kinerja algoritma hash pada keamanan data: Perbandingan. *Jurnal Quancom*, *2*(2). https://doi.org/10.62375/jqc.v2i2.432

Wijaya, R., Miharja, S., & Wilson. (2021). Implementasi algoritma AES-128 dan SHA-256 dalam perancangan aplikasi pengamanan file dokumen. *Jurnal TIMES*, *10*(2), 80–87. https://doi.org/10.51351/jtm.10.2.2021666