Implementasi *Secret Sharing* Skema Shamir pada Pola Generatif untuk Autentikasi Visual Berbasis *Similarity*

Vanson Kurnialim - 13522049[1,2]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13522049@std.stei.itb.ac.id, [2]vansonkurnialim@gmail.com

***Abstract*— Penelitian ini mengatasi risiko privasi dan penyimpanan pada autentikasi visual konvensional dengan mengusulkan penggunaan pola generatif unik sebagai kunci rahasia. Dibangkitkan secara deterministik melalui fungsi *hash* SHA-256 dan PRNG, metode ini meniadakan kebutuhan basis data citra fisik karena kunci diciptakan secara langsung di memori. Pola tersebut didistribusikan menggunakan skema *Shamir's Secret Sharing* berbasis *Finite Field* GF(251) untuk menjamin presisi matematis tanpa distorsi data. Validasi integritas menggunakan *Structural Similarity Index Measure* (SSIM) membuktikan sistem memiliki sifat *tamper-evident* yang kuat. Gangguan minimal pada *shares* menyebabkan kegagalan rekonstruksi, sehingga menjamin keamanan akses sekaligus melindungi privasi pengguna dari pencurian data biometrik.**

***Kata kunci*—Pola Generatif, Keamanan Deterministik, Shamir's Secret Sharing, GF(251), SSIM.**

I. Latar Belakang

Dalam ekosistem digital modern, keamanan data telah bertransformasi menjadi kebutuhan fundamental seiring dengan meningkatnya ancaman siber dan risiko kegagalan sistem penyimpanan. Mekanisme pengamanan konvensional umumnya bergantung pada manajemen kunci kriptografi terpusat, di mana akses terhadap informasi sensitif dilindungi oleh satu kunci privat tunggal. Pendekatan ini menciptakan kerentanan fatal berupa *Single Point of Failure*. kegagalan dalam mengelola kunci baik karena kehilangan, kerusakan perangkat, maupun pencurian dapat mengakibatkan hilangnya akses data secara permanen tanpa opsi pemulihan. Oleh karena itu, terdapat kebutuhan mendesak untuk mengembangkan mekanisme autentikasi yang mampu mendistribusikan kepercayaan (*distributed trust*) ke beberapa entitas, sehingga keamanan sistem tidak lagi bergantung mutlak pada satu titik penyimpanan saja.

Salah satu solusi yang berkembang untuk mengatasi masalah sentralisasi kunci adalah skema pembagian rahasia (*secret sharing scheme*) yang diimplementasikan pada media visual. Metode ini memungkinkan sebuah kunci akses disembunyikan dan dipecah ke dalam beberapa citra digital, yang kemudian dapat disatukan kembali untuk memulihkan informasi asli. Namun, penerapan skema kriptografi pada media citra menghadapi tantangan besar terkait kompatibilitas tipe data. Algoritma kriptografi umumnya bekerja pada domain bilangan yang presisi, sedangkan format citra digital standar memiliki keterbatasan ruang penyimpanan pada setiap pikselnya. Ketidakselarasan ini sering kali menyebabkan degradasi data saat proses penyisipan informasi, di mana hasil rekonstruksi citra mengalami penurunan kualitas atau distorsi (*noise*) yang signifikan. Dalam konteks autentikasi keamanan tinggi, distorsi sekecil apa pun tidak dapat ditoleransi karena dapat mengubah makna data secara keseluruhan.

Permasalahan integritas data tersebut sering kali diselesaikan dengan menyertakan berkas metadata tambahan atau menggunakan format penyimpanan khusus yang tidak standar, yang pada akhirnya mengurangi efisiensi dan kepraktisan sistem. Selain itu, penggunaan citra visual sebagai kunci juga menimbulkan isu privasi ketika pengguna diharuskan menggunakan foto pribadi atau biometrik wajah, yang rentan terhadap penyalahgunaan identitas. Oleh karena itu, diperlukan sebuah pendekatan sistematis yang mampu menjembatani kesenjangan antara presisi algoritma kriptografi dengan keterbatasan format media penyimpanan, sekaligus menjamin privasi pengguna melalui penggunaan pola visual yang tidak bersifat personal.

Makalah ini bertujuan untuk merancang dan membangun sebuah sistem autentikasi visual yang mampu mengatasi masalah degradasi data tersebut melalui pendekatan aritmatika modular yang presisi. Sistem yang diusulkan dirancang untuk menjamin rekonstruksi citra yang bersifat *lossless* (tanpa penurunan kualitas), sehingga validitas kunci dapat diverifikasi secara akurat. Selain itu, penelitian ini juga mengusulkan penggunaan pola generatif abstrak sebagai media penyimpanan rahasia untuk menjawab permasalahan

privasi dan keunikan kunci. Melalui pengembangan ini, diharapkan tercipta sebuah mekanisme pengamanan data yang tidak hanya tangguh dan terdesentralisasi, tetapi juga praktis dan memiliki integritas validasi yang tinggi.

II. LANDASAN TEORI

A. Representasi Citra Digital

Citra digital secara fundamental dapat didefinisikan sebagai fungsi dua dimensi, f(x, y), di mana x dan y merupakan koordinat spasial bidang datar. Agar dapat diolah oleh komputer, citra harus melalui proses digitalisasi yang melibatkan dua tahapan pemetaan nilai intensitas ke dalam nilai diskrit. Hasil dari proses ini adalah sebuah matriks elemen-elemen diskrit yang dikenal sebagai elemen gambar piksel.

Makalah ini menggunakan format citra berwarna yang mengadopsi model ruang warna RGB. Dalam model ini, setiap warna yang terlihat merupakan hasil superposisi aditif dari tiga warna primer cahaya yaitu merah (*red*), hijau (*green*), dan biru (*blue*). Oleh karena itu, sebuah piksel pada citra berwarna tidak hanya menyimpan satu nilai tunggal, melainkan sebuah vektor yang terdiri dari tiga komponen intensitas:

$$p(x,y) = [R(x,y), G(x,y), B(x,y)]$$

Dalam implementasi komputasi, citra RGB sering diperlakukan sebagai tumpukan tiga matriks terpisah, di mana operasi pengolahan citra dilakukan secara independen pada masing-masing kanal tersebut.

Aspek paling krusial dalam kriptografi visual adalah kedalaman bit atau tipe data yang digunakan untuk menyimpan intensitas setiap kanal warna. Standar citra digital yang paling umum menggunakan kedalaman 8-bit per kanal. Ini berarti setiap komponen warna (R, G, atau B) dialokasikan memori sebesar 8 bit atau 1 byte. Maka dari itu, terdapat keterbatasan rentang nilai dari 0 sampai 255.

B. Skema Pembagian Rahasia Shamir

Shamir's Secret Sharing (SSS) adalah sebuah algoritma kriptografi yang diperkenalkan oleh Adi Shamir pada tahun 1979. Skema ini bertujuan untuk memecah sebuah data rahasia (S) menjadi n bagian berbeda yang disebut sebagai *shares*. Fitur fundamental dari skema (k, n) ini adalah properti ambang batas (*threshold*), di mana rahasia asli hanya dapat dipulihkan jika dan hanya jika sejumlah minimal k *share* digabungkan kembali.

Landasan matematis dari algoritma Shamir adalah interpolasi polinomial. Teorema dasar aljabar menyatakan bahwa diperlukan minimal k titik koordinat unik untuk mendefinisikan sebuah polinomial berderajat k-1 secara tunggal. Untuk menyembunyikan sebuah rahasia S, sistem akan membangkitkan sebuah polinomial acak f(x) dengan derajat k-1. Konstruksi polinomial tersebut didefinisikan sebagai berikut:

$$f(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{k-1}x^{k-1} \pmod p$$

Setelah polinomial terbentuk, sistem pembagi akan menghitung nilai fungsi f(x) pada titik-titik x yang berbeda. Setiap shareholder akan menerima pasangan koordinat (x, y) sebagai *share* mereka. Karena a_0 (rahasia) adalah konstanta (f(0)), dan koefisien lainnya acak, maka setiap nilai y_i akan tampak sebagai data acak yang tidak memiliki korelasi visual dengan citra asli. Kekuatan keamanan skema Shamir terletak pada sifat matematis polinomial. Jika seorang penyerang hanya memiliki k-1 titik, terdapat tak terhingga kemungkinan polinomial yang dapat melewati titik-titik tersebut, yang berarti terdapat tak terhingga kemungkinan nilai f(0) atau rahasia S. Hal ini membuat metode *brute force* menjadi tidak mungkin dilakukan.

C. Galois Field

Dalam aljabar abstrak, sebuah lapangan (*field*) adalah struktur aljabar di mana operasi penjumlahan, pengurangan, perkalian, dan pembagian (kecuali pembagian dengan nol) didefinisikan dan memenuhi sifat-sifat dasar aritmatika. *Finite Field* yang sering disebut sebagai *Galois Field* sdalah lapangan yang mengandung jumlah elemen yang terbatas. Implementasi algoritma kriptografi pada komputer, khususnya Shamir's Secret Sharing, memerlukan struktur matematis yang presisi. Perhitungan matematika yang digunakan pada komputasi memiliki kelemahan dalam representasi presisi (*floating point errors*), yang dapat menyebabkan kegagalan fatal dalam rekonstruksi data kriptografis. Oleh karena itu, digunakan *Galois Field* dengan orde bilangan prima.

Tantangan utama dalam menerapkan *Shamir's Secret Sharing* pada citra digital adalah menyesuaikan ruang bilangan algoritma dengan ruang warna citra. Format citra standar menggunakan kanal warna 8-bit yang menampung nilai integer [0, 255] seperti yang sudah dijelaskan pada bagian sebelumnya. Bilangan prima terbesar yang kurang dari atau sama dengan 255 adalah 251.

Oleh karena itu, pada makalah ini digunakan GF(251). Dengan membatasi ruang warna input ke rentang [0, 250], seluruh operasi kriptografi dapat berjalan sempurna di dalam struktur data citra standar. Piksel hasil enkripsi dijamin valid secara visual dan matematis, memungkinkan rekonstruksi citra yang bersifat *lossless* karena eliminasi total terhadap galat pembulatan (*rounding errors*).

D. Interpolasi Lagrange

Interpolasi Lagrange adalah metode matematis yang digunakan untuk menemukan kembali sebuah fungsi polinomial utuh hanya dengan mengetahui beberapa titik koordinat yang dilaluinya. Dalam sistem *Shamir's Secret Sharing,* metode ini berfungsi sebagai kunci pembuka untuk merekonstruksi data rahasia (S) dari kumpulan share yang dimiliki pengguna.

Tujuan interpolasi ini bukanlah menggambar ulang

seluruh grafik polinomial, melainkan hanya mencari nilai titik potong kurva terhadap sumbu-Y, yaitu saat x=0. Hal ini karena dalam fase enkripsi, data rahasia asli selalu diletakkan pada posisi f(0).

Oleh karena itu, rumus Lagrange dapat disederhanakan. Tidak perlu menghitung persamaan untuk seluruh nilai x, tetapi cukup menghitung penjumlahan berbobot dari *share-share* (y_i) yang ada untuk mendapatkan nilai di titik nol. Normalnya, rumus interpolasi menggunakan operasi pembagian pecahan yang rumit, namun karena sistem ini didasarkan pada GF(251), operasi pembagian diganti dengan modulo. Alhasil, hasil perhitungan rekonstruksi selalu berupa bilangan bulat yang presisi dan sesuai dengan format warna piksel, tanpa ada error pembulatan sedikitpun.

E. Structural Similarity Index Measure (SSIM)

Structural Similarity Index Measure (SSIM) adalah sebuah metode standar yang digunakan untuk mengukur tingkat kemiripan antara dua citra digital. Berbeda dengan metode tradisional seperti *Mean Squared Error* (MSE) yang hanya menghitung selisih nilai absolut antar piksel, SSIM dirancang untuk memodelkan persepsi sistem visual manusia yang sangat sensitif terhadap perubahan struktur informasi dalam sebuah gambar. Komponen utama evaluasi kualitas citra yang dilakukan oleh SSIM adalah luminansi, kontras, dan struktur. Secara matematis, SSIM dihitung dengan rumus sebagai berikut:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

Rentang nilai yang dihasilkan oleh SSIM adalah -1 hingga 1 dimana 1 berarti kedua citra memiliki kemiripan yang identik sempurna. Karena sistem menggunakan aritmatika modular yang sensitif, kesalahan sekecil apa pun pada input kunci (*shares*) akan memicu *snowball effect* yang mengubah hasil rekonstruksi menjadi penuh dengan *noise*. Maka dari itu, SSIM cocok untuk digunakan sebagai token validasi bagi sistem.

F. Pola Generatif Unik

Pola generatif adalah metode penciptaan citra digital secara prosedural menggunakan algoritma komputer, dimana visual yang terbentuk bukan hasil tangkapan kamera atau goresan tangan, melainkan hasil perhitungan matematis. Dalam sistem makalah ini, pola generatif berfungsi sebagai media penampung rahasia yang bersifat netral dan unik. Agar dapat digunakan sebagai kunci, pola gambar harus memiliki sifat paradoks: harus acak namun *reproducible* atau bisa dihasilkan kembali. Untuk mencapai hal ini, sistem memanfaatkan *Pseudo-Random Number Generator* (PRNG). PRNG adalah algoritma yang menghasilkan deret angka yang tampak acak, namun sebenarnya ditentukan sepenuhnya oleh *seed* yang didapatkan dari hasil *hash*. Jika *seed*-nya sama, urutan angka acak yang dihasilkan akan selalu sama persis. Hasil dari PRNG inilah yang menjadi parameter dalam pembentukan pola citra.

Pada pembuatan *seed* yang unik bagi setiap pengguna, digunakan fungsi *hashing* kriptografi, yaitu SHA-256. Fungsi ini mengubah input teks (seperti *User ID* atau kata sandi) menjadi deretan karakter alfanumerik dengan panjang tetap yang unik.

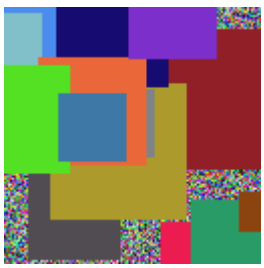
III. Analisis dan Implementasi

Dibangun sebuah antarmuka berbasiskan *command line interface* atau sarana penggunaan program berbasis terminal. Program dibuat dalam bahasa pemrograman Python dengan library-library pendukung seperti numpy untuk proses perhitungan, Pillow untuk pemrosesan citra dasar, dan scikit-image untuk pemrosesan dan analisa citra lebih lanjut.

Program dibentuk dengan 3 fungsi atau fitur utama. Yang pertama adalah pembangkitan pola unik yang akan dijadikan sebagai kunci. Pengguna akan diminta memasukkan sebuah kunci berupa *string* yang akan diproses oleh program dan digunakan untuk membangkitkan sebuah citra unik. Walau sebenarnya strategi pembangkitan citra didasarkan dengan sebuah *seed* yang didapat dari *hash* sehingga secara teknis pola generative dapat dicapai dari 2 *seed* yang berbeda, kita dapat mengabaikannya karena kemungkinannya yang sangat kecil.

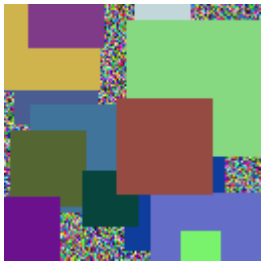
Alur pembuatan citra secara lebih spesifik dimulai dari masukan kunci *string* yang di-*hash* dengan algoritma SHA-256. Hasil dari proses ini adalah string heksadesimal 64 karakter yang merepresentasikan ‘sidik jari’ digital dari citra kunci yang ingin dikembangkan. *String* heksadesimal dikonversi menjadi bilangan bulat atau *integer*. Nilai *integer* ini kemudian disuntikkan sebagai *seed* awal ke dalam fungsi *Pseudo-Random Number Generator* (PRNG). Langkah ini krusial untuk menjamin sifat deterministik sistem. Tanpa *seed* yang konsisten, PRNG akan menghasilkan pola acak yang berbeda setiap kali program dijalankan, yang akan menyebabkan sistem yang tidak *reproducible* dan tidak dapat diandalkan.

Setelah PRNG diinisialisasi, sistem melakukan iterasi sebanyak 15 kali untuk membangkitkan elemen-elemen visual. Pada setiap iterasi, PRNG dipanggil untuk menghasilkan serangkaian parameter acak berupa koordinat spasial, dimensi geometris, dan intensitas warna dalam RGB untuk elemen yang akan diciptakan dan ditambahkan dalam citra kunci. Semua elemen objek geometris tersebut akan dimasukkan secara tumpang tindih.



Gambar 1. Pola unik generatif

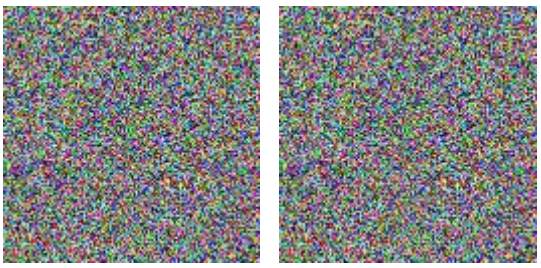
Gambar di atas merupakan contoh pola unik yang dibangkitkan dengan kunci ‘vanson’. Dapat dilihat bahwa citra yang dihasilkan sangatlah berbeda dengan citra yang menggunakan kunci ‘Vanson’ walau hanya berbeda satu huruf pada gambar di bawah ini.



Gambar 2. Pola unik generatif (2)

Pola citra seperti yang ditampilkan akan sangat sulit untuk ditiru dan direplikasi sehingga menambah lapisan keamanan pada kunci, berbeda dengan model citra lain. Salah satu keunggulan lain dari penggunaan pola unik generative sebagai kunci adalah semua aspek berasal dari sistem. Hal ini memudahkan sistem untuk melakukan pemrosesan dan kpmputasi citra secara seragam tanpa perlu memerhatikan format citra masukan pengguna. Karena kunci dibangkitkan di dalam sistem program, sudah dapat dipastikan bahwa kunci satu-satunya hanya dapat ditemukan dan disimpan dari dalam sistem tanpa ada kemungkinan pihak di luar sistem untuk mengetahuinya.

Fitur kedua adalah pembagian *shares* dari citra kunci yang dibangkitkan dari fitur pertama. Program akan meminta masukan pengguna untuk jumlah *shares* yang akan dibangkitkan dan juga jumlah *threshold* atau batas ambang untuk rekonstruksi citra kunci. Minimal jumlah *shares* adalah dua, begitu juga dengan jumlah minimal *threshold*.



Gambar 3. Contoh shares

Kedua gambar di atas merupakan contoh hasil pembagian *shares* dari sebuah citra kunci dengan jumlah *shares* 5 dan *threshold* 2. Dua *shares* tersebut terlihat tidak memiliki perbedaan bagi kita menggunakan mata manusia. Namun secara aktual, kedua *shares* tersebut berbeda pada level data atau piksel pembentuknya.

Mekanisme kriptografis yang diimplementasikan pada fitur ini berfokus pada mengubah citra rahasia tunggal menjadi N buah citra *share*. Proses ini dilakukan pada level piksel dengan menerapkan algoritma *Shamir’s Secret Sharing* yang dimodifikasi menggunakan aritmatika *Finite Field* GF(251).

Sistem menetapkan parameter N (jumlah total *share* yang akan dibuat) dan K (jumlah minimal *share* untuk rekonstruksi atau *threshold*). Sistem juga menetapkan modulus prima P = 251. Sistem membaca citra rahasia dan melakukan penelusuran (*looping*) pada setiap koordinat (x, y) dari piksel pertama hingga piksel terakhir. Pada setiap piksel, nilai warna akan dipisah menjadi tiga komponen warna RGB dan dienkripsi secara paralel. Untuk setiap komponen warna, sistem akan membentuk sebuah polynomial acak dengan derajat K-1. Sistem lalu menghitung nilai dari fungsi polinomial sesuai dengan jumlah *shares* yang akan disimpan ke dalam citra *shares* yang bersesuaian. Hasil akhirnya adalah N buah citra digital baru yang secara visual tampak seperti *noise* (titik-titik acak), namun secara matematis mengandung informasi rahasia yang terdistribusi.

Secara teori jumlah *shares* dapat dihasilkan sebanyak apapun dari citra kunci. Hal yang sama dengan jumlah *threshold* untuk rekonstruksi citra kunci. Walaupun begitu, kita perlu memerhatikan kekuatan komputasi yang dibutuhkan untuk melakukannya. Lebih tepatnya, batasan yang muncul terletak pada representasi angka yang terlibat dalam perhitungan sangatlah besar sehingga *overflow* dan membuat sistem *error*. Maka, untuk mengimplementasi sistem *secret sharing* untuk jumlah *shareholder* yang besar, diperlukan juga sumber daya perangkat keras yang besar pula.

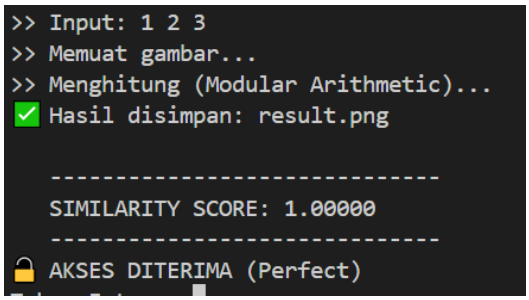
Fitur ketiga adalah rekonstruksi dan validasi citra kunci berdasarkan *shares* masukan. Sistem akan meminta masukan *shares* yang ingin direkonstruksi dengan jumlah bebas dari pengguna. Setelah diterima, sistem akan melakukan rekonstruksi citra kunci dan mengeceknya dengan citra kunci yang disimpan oleh sistem. Jika nilai kesamaan atau *similarity* melewati batas, maka akses akan diterima.

Sistem memproses citra *shares* piksel demi piksel. Pada koordinat yang sama, akan diambil nilai intensitas warna dari tiap *shares*. Untuk mendapatkan nilai intensitas warna yang asli, sistem menghitung penjumlahan berbobot menggunakan rumus lagrange yang disederhanakan untuk x=0. Tantangan teknis utama di sini adalah operasi pembagian. Dalam aritmatika modular GF(251), pembagian a/b diterjemahkan menjadi perkalian a x b^{-1} mod(251). Sistem mencari invers modular dari penyebut menggunakan *Extended Euclidean Algorithm* atau perpangkatan *fermat*. Hasil perhitungan lagrange kemudian dimodulo 251. Nilai akhir ini adalah intensitas warna asli RGB yang dikembalikan pada citra hasil rekonstruksi.

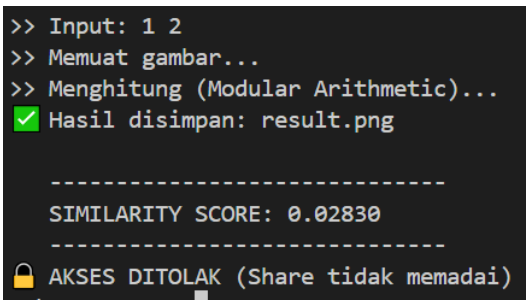
Hasil rekonstruksi citra akan dibandingkan dengan citra kunci asli menggunakan SSIM. Penggunaan SSIM memberikan batas toleransi yang presisi. Jika kumpulan *share* valid, aritmatika modular menjamin hasil rekonstruksi *lossless* sehingga skor SSIM akan mencapai 1.0 atau sangat mendekati 1.0. Sebaliknya, jika terdapat

satu saja *share* yang invalid, sifat *snowball effect* pada operasi polinomial akan menyebabkan nilai piksel teracak secara total, menghasilkan citra *static noise* dengan skor SSIM yang sangat rendah (mendekati 0.0), sehingga sistem dapat menolak akses dengan tingkat kepercayaan tinggi.

Berikut adalah contoh hasil rekonstruksi citra kunci dengan kunci 'Vanson', jumlah *shares* 5, dan *threshold* 3. Ketika jumlah *shares* memadai maka hasil rekonstruksi akan memeroleh nilai SSIM sempurna yaitu 1.0 dan akan sangat rendah ketika jumlah *shares* tidak memadai.



Gambar 4. Hasil rekonstruksi berhasil



Gambar 5. Hasil rekonstruksi gagal

Pada kasus normal, sistem telah berhasil melakukan pembangkitan citra kunci, pembagian *shares* sesuai dengan parameter, dan rekonstruksi kembali citra sebagai skema autentikasi.

Untuk pengujian lebih lanjut, dilakukan rekonstruksi dengan citra *shares* yang telah dirusak. Terdapat 3 jenis kerusakan yang dilakukan pada *shares* untuk mensimulasikan perubahan data. Ada *noise* atau bintik acak untuk mensimulasikan data yang korup. Yang kedua adalah *crop* atau simulasi sebagian data yang hilang. Lalu ada *brightness* atau intensitas keterangan dari citra untuk mensimulasikan manipulasi citra atau edit.

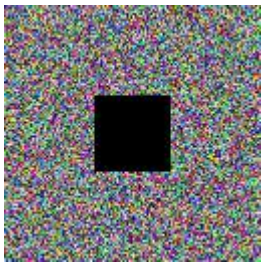
Pengujian dilakukan dengan masukan dan parameter yang sama yaitu kunci pembangkitan citra 'Vanson', jumlah *shares* 5, dan *threshold* berjumlah 3. Hasil dari pengujian dapat dilihat pada tabel di bawah ini.

Jenis Pengujian	Nilai SSIM
noise / bintik acak	0.2673
crop / data hilang	0.8877
brightness / intensitas warna	0.9109

Tabel 1. Hasil pengujian dengan share rusak

Pada pengujian *noise*, *shares* tidak tampak berbeda dari aslinya namun citra telah ditambahkan gangguan berupa bintik-bintik yang tersebar di seluruh citra. Rekonstruksi dengan jumlah *shares* sama dengan *threshold* menghasilkan nilai SSIM sebesar 0.2673 yang artinya rekonstruksi gagal sehingga akses akan ditolak. Hal ini menunjukkan bahwa sistem unggul dalam aspek integritas data. Program dapat menilai bahwa *shares* yang digunakan sebagai masukan telah dirusak karena perubahan nilai piksel sedikit saja dari nilai aslinya akan mengembalikan keluaran yang jauh berbeda. Sekitar 26% kemiripan dari pengujian tersebut hanyalah merupakan kebetulan statistik distribusi warna acak dan bukan kemiripan citra sesungguhnya.

Pengujian kedua terkait *crop* menghasilkan menghasilkan nilai SSIM sebesar 0.88771. Nilai kemiripan ini tergolong cukup besar mengingat data citra *share* telah dirusak. Berikut adalah ilustrasi dari citra *share* yang telah di-*crop* agar kerusakan yang terjadi lebih dapat dipahami.



Gambar 6. Citra share yang di-crop

Secara heuristik, perubahan citra yang cukup signifikan seperti pada ilustrasi seharusnya tidak menghasilkan nilai *similarity* setinggi itu. Namun nyatanya, proses rekonstruksi dan pengecekan citra dilakukan secara *mapping one to one* pada setiap piksel warna. Bagian tengah yang dirusak pada ilustrasi kurang lebih adalah satu dari sembilan bagian citra *share*. Maka dari itu, hanya 11,111…% bagian citra yang menjadi tidak valid. Hal ini sesuai dengan hasil nilai SSIM yang didapatkan.

Pada pengujian *brightness*, nilai SSIM yang didapatkan lebih baik daripada pengujian-pengujian sebelumnya dengan nilai SSIM 0.9109. Hal ini masuk akal karena perubahan yang dilakukan hanyalah pada intensitas warna piksel yang masih mengikuti nilai warna citra awal. Satu *share* rusak dari tiga total *shares* juga membantu meminimalisir perbedaan nilai dari citra hasil rekonstruksi dengan citra kunci.

Berdasarkan hasil pengujian, program berjalan sesuai dengan tujuannya. Beberapa kerusakan memang tetap memberikan nilai kemiripan yang cukup tinggi, namun hal tersebut tidak menjadi halangan dengan mengatur batas validitas sesuai dengan kebutuhan. Untuk menjaga integritas dan ketetatan sistem, disarankan hanya menerima nilai kemiripan di atas 0.99. Batas tersebut dipasang untuk mengantisipasi perubahan data sedikit dari hasil perhitungan, data korup minor, dan keadaan lainnya yang tidak dapat diantisipasi.

IV. KESIMPULAN

Pada makalah ini, sebuah sistem *secret sharing* berbasiskan skema Shamir menggunakan pola unik generatif sebagai kunci dan *similarity* sebagai metode pengecekan, telah berhasil dibuat. Sistem telah berhasil membangkitkan pola unik sebagai kunci, membagi citra kunci menjadi beberapa *shares* dengan *threshold* tertentu, dan melakukan rekonstruksi citra serta pengecekan validitas dengan baik. Penerapan aritmatika *Finite Field* GF(251) terbukti efektif dalam mengatasi permasalahan *floating point error* dan *overflow* data yang umum terjadi pada algoritma Shamir standar. Mekanisme ini menjamin bahwa seluruh operasi kriptografi tetap berjalan dalam domain bilangan bulat 8-bit, sehingga rekonstruksi citra dapat dilakukan secara *lossless* (sempurna) tanpa distorsi warna.

Metode SSIM juga bekerja dengan baik sebagai mekanisme verifikasi integritas. Pengujian menunjukkan bahwa sistem memiliki sensitivitas tinggi terhadap kesalahan. rekonstruksi dengan shares yang valid menghasilkan nilai SSIM mendekati 1.0 (identik), sedangkan rekonstruksi dengan *shares* yang terdistorsi (mengandung *noise*) menghasilkan penurunan nilai SSIM.

Penggunaan pola generatif berbasis *hash* menghilangkan ketergantungan pada penyimpanan basis data citra biometrik atau fisik, meningkatkan aspek privasi dan efisiensi penyimpanan karena kunci dibangkitkan secara langsung di dalam sistem.

REFERENCES

[1] Munir, Rinaldi. 2025. "Kriptografi Visual (Bagian 1)". Bahan Kuliah. Bandung: Program Studi Teknik Informatika, Institut Teknologi Bandung. Diakses dari https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2025-2026/38-Kriptografi-Visual-Bagian1-2025.pdf pada 20 Desember 2025.

[2] Munir, Rinaldi. 2025. "Kriptografi Visual (Bagian 2)". Bahan Kuliah. Bandung: Program Studi Teknik Informatika, Institut Teknologi Bandung. Diakses dari https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2025-2026/39-Kriptografi-Visual-Bagian2-2025.pdf.

[3] Munir, Rinaldi. 2025. "Skema Pembagian Data Rahasia". Bahan Kuliah. Bandung: Program Studi Teknik Informatika, Institut Teknologi Bandung. Diakses dari https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2025-2026/37-Skema-Pembagian-Data-Rahasia-2025.pdf.

[4] Munir, Rinaldi. 2025. "Beberapa Fungsi Hash". Bahan Kuliah. Bandung: Program Studi Teknik Informatika, Institut Teknologi Bandung. Diakses dari https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2025-2026/27-Beberapa-fungsi-hash-2025.pdf.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 20 Desember 2025

Vanson Kurnialim