JURNAL MEDIA INFORMASI TEKNOLOGI
Volume 3, No. 1, Februari 2026
DOI: https://doi.org/10.69616/mit.v3i1.250

Vigenere Cipher Cryptography for Secure Data Transmission in IoT Smart Door Using QR Code

Riski Arasyid [1]*, Syamsul Bahri [1], Kasliono [1]
[1] Universitas Tanjungpura, Indonesia.
* Corresponding Author. E-mail: h1051211025@student.untan.ac.id

Keywords
Vigenere Cipher;
Smart Door;
Internet of Things;
QR Code;
Data Security.

ABSTRACT

The Smart Door system based on the Internet of Things (IoT) using QR Code offers automated access but remains vulnerable to data sniffing attacks. This study implements the Vigenère Cipher algorithm to encrypt transmitted data and analyzes the ciphertext's resistance to attacks as well as its impact on communication delay. The system is built using ESP32-CAM as the QR scanner and NodeMCU ESP32 as the main controller, with encryption applied across three communication paths: ESP32-CAM to server, server to database, and server to ESP32. Testing involved data sniffing, delay analysis, and ciphertext evaluation using tools such as dCode and CryptoTool. From 20 sniffed QR Code results, 10 random samples were tested, and only 3 were recognized as Vigenère Cipher and none were decrypted successfully. In the server-to-ESP32 path, only 1 out of 10 ciphertexts was detected and all remained undeciphered. The average delay was recorded at 2.473 seconds (door unlocking) and 3.491 seconds (buzzer activation), with variations due to network stability. The results indicate that Vigenère Cipher effectively enhances data security in resource-constrained IoT devices, although delay optimization is needed to meet real-time system requirements.

PENDAHULUAN

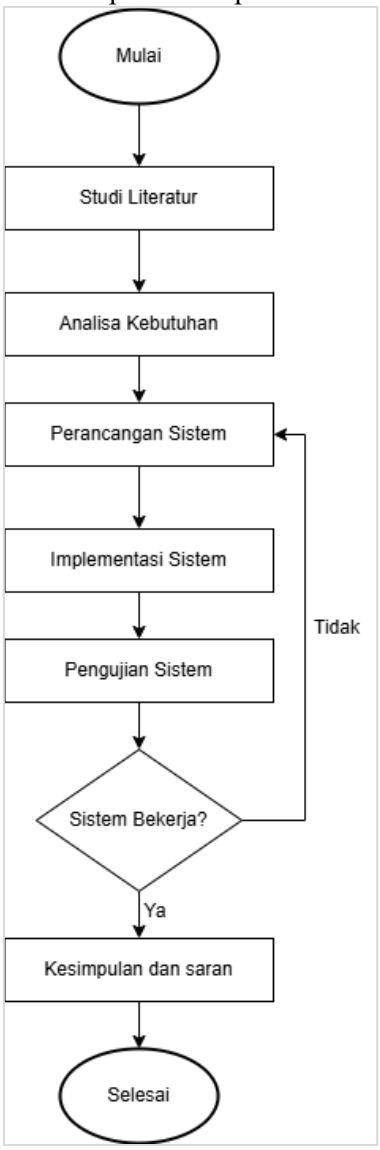
Internet of Things (IoT) menghubungkan objek fisik ke jaringan internet untuk bertukar data secara efisien dan telah banyak diterapkan dalam kehidupan sehari-hari [1]. Salah satu implementasinya adalah Smart Door lock, pengunci pintu elektronik yang dapat diakses melalui perangkat pintar tanpa kunci fisik. Pada sistem ini, QR Code umum digunakan sebagai metode autentikasi karena mampu menyimpan data dalam jumlah besar serta dipindai dengan cepat [2],[3].

Meskipun menawarkan kemudahan, sistem keamanan berbasis IoT menghadapi risiko serius terkait kerahasiaan data. Ancaman yang sering muncul adalah serangan Man-in-the-Middle (MitM) dan sniffing, di mana penyerang dapat menyadap atau memodifikasi data yang ditransmisikan antarperangkat. Kondisi ini semakin mengkhawatirkan mengingat sebagian besar komunikasi IoT dilaporkan masih belum terenkripsi dengan baik [4], [5].

Untuk mengatasi permasalahan tersebut, penelitian ini menerapkan Vigenère Cipher sebagai algoritma kriptografi ringan guna mengamankan jalur komunikasi pada sistem Smart Door berbasis QR Code. Pendekatan ini dipilih karena sesuai dengan keterbatasan perangkat IoT seperti ESP32-CAM yang memiliki daya komputasi terbatas [6]. Penelitian ini juga menganalisis pengaruh enkripsi terhadap delay autentikasi, dengan tujuan menilai keseimbangan antara tingkat keamanan data dan efisiensi sistem pada perangkat IoT.

METODE

Metodologi penelitian ini mencakup sejumlah langkah, dimulai dari studi literatur, analisis kebutuhan, perancangan sistem, implementasi, pengujian sistem, hingga analisis data dan penarikan kesimpulan. Diagram metode penelitian dapat dilihat pada Gambar 1.



Gambar 1. Metode Penelitian

1. Studi Literatur
   Tahapan ini dilakukan dengan mengumpulkan dan mengkaji referensi terkait teknologi IoT, sistem Smart Door, serta algoritma kriptografi Vigenère Cipher. Studi literatur bertujuan untuk memperoleh pemahaman mengenai karakteristik sistem, metode autentikasi, serta mekanisme enkripsi yang sesuai dengan keterbatasan perangkat mikrokontroler, sehingga penelitian dapat dilaksanakan sesuai dengan ruang lingkup yang telah ditetapkan.
2. Analisis Kebutuhan
   Analisis kebutuhan dilakukan untuk memastikan sistem Smart Door sesuai dengan tujuan penelitian dan ruang lingkup yang ditetapkan. Tahapan ini mencakup identifikasi perangkat keras seperti ESP32-CAM, NodeMCU ESP32, dan solenoid door lock, serta perangkat lunak seperti Laravel, Arduino IDE, dan Wireshark.

2.1 Kebutuhan Perangkat Keras

Berikut merupakan komponen perangkat keras yang digunakan dalam sistem Smart Door berbasis IoT:

a) NodeMCU ESP32
   NodeMCU ESP32 adalah mikrokontroler berbasis *WiFi* yang mendukung monitoring dan kontrol pada proyek IoT. Modul ini mirip Arduino, dapat diprogram via Arduino IDE, serta unggul dalam konektivitas internet [7].
b) ESP32-CAM
   ESP32-CAM adalah modul mikrokontroler dengan kamera dan WiFi bawaan, mendukung pemantauan real-time. Pemrogramannya memerlukan modul FTDI USB to TTL sebagai antarmuka ke komputer, dan dilakukan melalui Arduino IDE [8].
c) *Solenoid Door Lock*
   Solenoid Door Lock adalah aktuator linier 12V DC untuk pengunci pintu elektronik, tersedia dalam tipe NC dan NO, serta dikendalikan melalui sinyal digital mikrokontroler [9].
d) Modul Relay
   Modul relay adalah saklar elektromagnetik yang memungkinkan pengendalian arus tinggi menggunakan sinyal bertegangan rendah melalui peralihan kontak NC–NO [10].
e) Buzzer
   Buzzer adalah komponen elektronik yang mengubah energi listrik menjadi suara dan biasa digunakan sebagai indikator atau alarm pada sistem elektronik [11].

2.2 Kebutuhan Perangkat Lunak

Adapun kebutuhan perangkat lunak dalam membangun sistem diantara lain:

a) Laravel
   Laravel adalah framework PHP berbasis MVC yang sederhana dan fleksibel, dirilis di GitHub dengan lisensi MIT [12].
b) Arduino IDE
   Arduino IDE adalah perangkat lunak berbasis Java untuk menulis, mengompilasi, dan mengunggah program C/C++ ke berbagai board mikrokontroler [13].
c) Ettercap
   Ettercap adalah alat analisis jaringan untuk audit keamanan dan pemantauan data. Fitur utamanya termasuk penyadapan aktif, pencurian kata sandi, dan serangan ARP poisoning untuk mengendus lalu lintas, bahkan pada koneksi terenkripsi seperti SSL [14].
d) Wireshark
   Wireshark adalah software open-source untuk menganalisis jaringan, memungkinkan pengguna menangkap dan memeriksa paket data secara rinci. Informasinya mencakup sumber, tujuan, jenis protokol, hingga isi data dalam setiap paket [15].

2.2 Kebutuhan Perangkat Lunak

Vigenère Cipher adalah metode kriptografi klasik berbasis substitusi abjad majemuk, di mana setiap karakter plainteks dienkripsi menggunakan karakter kunci yang berulang. Pergeseran huruf yang bervariasi membuat pola ciphertext tidak seragam, sehingga lebih sulit dipecahkan dibandingkan substitusi tunggal [16].

Enkripsi dilakukan menggunakan bujursangkar Vigenère (Tabula Recta), di mana setiap baris merupakan hasil pergeseran alfabet ala Caesar Cipher. Selain menggunakan tabel, proses enkripsi dan dekripsi juga dapat dilakukan melalui persamaan matematika [17].

Dalam penelitian ini, metode Vigenère Cipher dimodifikasi untuk mengenali 62 karakter yang terdiri atas huruf besar (A–Z), huruf kecil (a–z), dan angka (0–9). Proses enkripsi dilihat pada Rumus 1.

$$Ci = ((Pi + Ki) \bmod 62) \quad (1)$$

Dari Rumus 2, ciphertext (C) diperoleh dengan melakukan operasi modulo pada hasil penjumlahan plaintext (P) dan kunci (K). Untuk mendapatkan kembali pesan asli, proses dekripsi dilakukan dengan Rumus 2.

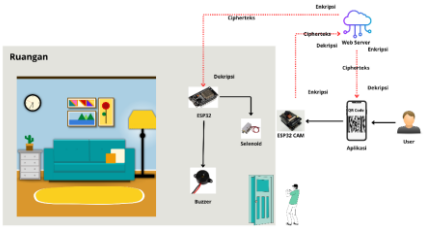
$$Pi = ((Ci - Ki + 62) \bmod 62) \quad (2)$$

Dengan keterangan:
Ci = karakter hasil enkripsi (ciphertext),
Pi = karakter asli (plaintext),
Ki = karakter kunci.

3. Perancangan Sistem

Perancangan sistem pada penelitian ini mencakup integrasi antara perangkat keras dan perangkat lunak dalam arsitektur Smart Door berbasis Internet of Things (IoT). Pada sisi perangkat keras, sistem menggunakan ESP32-CAM untuk memindai QR Code yang berisi data verifikasi pengguna. Hasil pemindaian kemudian dikirimkan ke cloud server untuk diproses lebih lanjut. Selanjutnya, NodeMCU ESP32 berfungsi sebagai pengendali aktuator berupa modul relay, solenoid door lock, dan buzzer, yang akan dioperasikan berdasarkan hasil autentikasi dari server.

Pada sisi perangkat lunak, sistem dibangun menggunakan Arduino IDE untuk pemrograman mikrokontroler serta framework Laravel sebagai backend server. Komunikasi antara perangkat dengan server dilakukan melalui API yang dilengkapi pengamanan menggunakan algoritma Vigenère Cipher. Algoritma ini digunakan untuk mengenkripsi data autentikasi, termasuk UUID, hasil pemindaian QR Code, dan instruksi sistem, sehingga tidak mudah diakses atau dimanipulasi oleh pihak ketiga. Ilustrasi menyeluruh mengenai arsitektur sistem ditampilkan pada Gambar 2.



Gambar 2. Arsitektur Sistem

4. Implementasi Sistem

Tahap implementasi dilakukan berdasarkan rancangan sistem yang telah dibuat, dimulai dari perakitan perangkat keras hingga pemrograman perangkat lunak. ESP32-CAM digunakan untuk membaca QR Code, sedangkan NodeMCU ESP32 berfungsi sebagai pengendali utama yang mengatur komunikasi dengan server serta pengoperasian aktuator. Setelah seluruh perangkat keras selesai dirakit, perangkat lunak diimplementasikan dengan menanamkan program pada masing-masing mikrokontroler. Program ini mencakup proses pembacaan QR Code, pengiriman data yang telah dienkripsi menggunakan algoritma Vigenère Cipher, serta verifikasi data pada sisi server.

5. Pengujian Sistem

Setelah sistem Smart Door berbasis IoT selesai dikembangkan, dilakukan pengujian untuk memastikan kinerja sesuai dengan rancangan. Pengujian difokuskan pada dua aspek utama, yaitu keamanan transmisi data dan pengukuran waktu delay. Uji keamanan dilakukan dengan memantau lalu lintas jaringan untuk memastikan bahwa data sensitif seperti UUID dan instruksi sistem tidak dapat diakses oleh pihak yang tidak berwenang. Selain itu, dilakukan uji ketahanan ciphertext dengan mencoba mengenali metode enkripsi yang digunakan serta melakukan analisis brute force menggunakan tools kriptografi umum untuk menilai daya tahan terhadap serangan. Selanjutnya, pengukuran waktu delay dilakukan pada setiap jalur komunikasi guna mengetahui pengaruh algoritma Vigenère Cipher terhadap performa sistem Smart Door berbasis IoT.

6. Pengumpulan Data

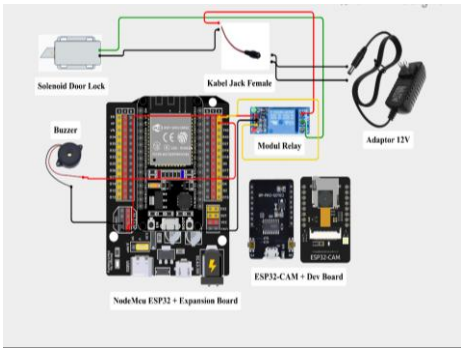
Pada tahap ini, data dikumpulkan dari hasil pengujian sistem setelah implementasi algoritma Vigenère Cipher pada Smart Door berbasis IoT. Data yang diperoleh mencakup hasil pemantauan lalu lintas komunikasi melalui simulasi sniffing, hasil uji ketahanan ciphertext menggunakan perangkat analisis kriptografi, serta hasil pengukuran waktu delay pada setiap jalur komunikasi antar perangkat. Proses pengumpulan data ini bertujuan untuk mengevaluasi sejauh mana algoritma Vigenère Cipher dapat menjaga kerahasiaan informasi sekaligus menilai efisiensi kinerja sistem pada perangkat dengan sumber daya terbatas seperti ESP32 dan ESP32-CAM..

7. Analisa dan Kesimpulan
   Tahap akhir dari penelitian ini adalah melakukan analisis serta menarik kesimpulan berdasarkan hasil pengujian sistem dan data yang telah diperoleh. Kesimpulan difokuskan untuk menjawab rumusan masalah yang telah ditetapkan sebelumnya, terutama mengenai penerapan algoritma Vigenère Cipher dalam menjaga keamanan transmisi data serta pengaruhnya terhadap waktu delay pada sistem Smart Door berbasis IoT. Selain itu, penelitian ini juga memberikan saran sebagai rekomendasi untuk pengembangan dan penyempurnaan sistem di masa mendatang agar lebih optimal dan adaptif terhadap kebutuhan pengguna.

HASIL DAN PEMBAHASAN

Implementasi Perangkat Keras

Pada tahap implementasi perangkat keras, sistem Smart Door berbasis IoT dibangun dengan memanfaatkan beberapa komponen inti. ESP32-CAM berfungsi sebagai pemindai QR Code yang ditempatkan di dekat pintu untuk membaca kode dari pengguna, sedangkan NodeMCU ESP32 berperan sebagai pengendali utama yang menerima hasil verifikasi dari server dan mengoperasikan aktuator. Aktuator tersebut terdiri dari relay, solenoid door lock, serta buzzer yang bekerja sesuai instruksi. Rangkaian hubungan antar komponen secara keseluruhan divisualisasikan melalui skema perangkat keras yang ditampilkan pada Gambar 3.



Gambar 3. Wiring Diagram Perangkat Keras

Dalam skema rangkaian, NodeMCU ESP32 berfungsi mengontrol relay yang menentukan kondisi buka atau tutup pada kunci pintu, sekaligus mengaktifkan buzzer sebagai penanda apabila proses verifikasi tidak berhasil. Selain itu, digunakan adaptor 12V untuk menyediakan suplai daya bagi solenoid door lock, karena komponen ini membutuhkan arus yang lebih besar dibandingkan perangkat keras lainnya.

Konfigurasi hubungan pin pada NodeMCU ESP32 dengan relay dan buzzer dijelaskan dalam Tabel 1.

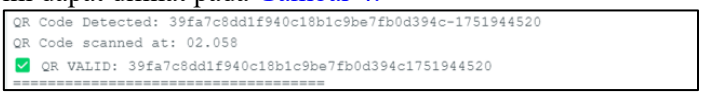
Tabel 1. Hubungan Pin Perangkat Keras

Komponen	Pin	Pin
ESP32	3v	Relay VCC
ESP32	GND	Relay GND

Komponen	Pin	Pin
ESP32	D23	Relay IN
ESP32	GND	Buzzer
ESP32	D22	Buzzer
Relay	COM	Kabel merah solenoid
Relay	NO	Kabel merah jack female
Solenoid	Kabel Hitam	Kabel Hitam jack female

Implementasi Perangkat Lunak

Implementasi perangkat lunak dilakukan pada tiga komponen utama, yaitu ESP32-CAM, server, dan NodeMCU ESP32. Sistem dirancang untuk menangani proses verifikasi QR Code sekaligus mengontrol perangkat keras seperti solenoid door lock dan buzzer, berdasarkan hasil autentikasi yang telah dienkripsi menggunakan algoritma Vigenère Cipher. Pada perangkat ESP32-CAM, dilakukan proses pemindaian QR Code untuk mengekstraksi UUID pengguna. Setelah data berhasil dipindai, UUID dikirimkan ke server melalui koneksi Wi-Fi untuk selanjutnya diproses. Ilustrasi alur proses ini dapat dilihat pada Gambar 4.



Gambar 4. Pembacaan QR Code pada ESP32-CAM

Setelah menerima UUID dari ESP32-CAM, server mencocokkannya dengan database. Jika valid, server mengirimkan perintah "unlock" yang dienkripsi menggunakan Vigenère Cipher; jika tidak valid, perintah "buzzer" dikirimkan. Server juga memverifikasi timestamp, di mana QR Code hanya dianggap sah dalam waktu 30 detik, selebihnya dinyatakan expired. Untuk respon proses ini dapat dilihat pada Gambar 5.

Gambar 5. Respon Verifikasi dari Server

NodeMCU ESP32 berfungsi mengambil instruksi dari server secara berkala melalui metode client-pull menggunakan REST API. Setiap instruksi yang diterima kemudian diproses dengan melakukan dekripsi menggunakan Vigenère Cipher. Apabila hasil dekripsi berupa perintah "unlock", maka sistem akan mengaktifkan relay untuk membuka kunci pintu. Sebaliknya, apabila hasil dekripsi menghasilkan perintah "buzzer", maka buzzer akan diaktifkan sebagai notifikasi bahwa akses telah ditolak. Hasil instruksi yang diterima dapat dilihat pada Gambar 6.

Gambar 6. Perintah yang diterima dari Server

Implementasi Algoritma Vigenere Cipher

Implementasi algoritma Vigenere Cipher pada sistem Smart Door berbasis IoT bertujuan untuk mengamankan komunikasi data pada tiga jalur utama: pengiriman UUID dari server ke website pengguna, pengiriman hasil pemindaian QR Code dari ESP32-CAM ke server, dan pengiriman instruksi perintah dari server ke ESP32.

Contoh Proses Enkripsi dan Dekripsi

Data Plaintext:
e94cab3d482548babdca7480e5eee904
Kunci :
oGl7Mho7d9

A. Ulangi kunci agar panjangnya sama dengan plainteks, sehingga kunci menjadi : oGl7Mho7d9 oGl7Mho7d9 oGl7Mho7d9oG

B. Konversi karakter plainteks dan kunci ke indeks
Tiap karakter dari plainteks dan kunci dikonversi ke indeks berdasarkan 62 karakter alfanumerik (A–Z = 0–25, a–z = 26–51, 0–9 = 52–61). Hasil konversi dapat dilihat pada Tabel 2.

Tabel 2. Konversi Karakter ke Indeks

No	PT	PT Index	Key	Key Index
1	e	30	o	40
2	9	61	G	6
...	...	...	...	...
31	0	52	o	40
32	4	56	G	6

C. Penjumlahan indeks dan modulo
Selanjutnya lakukan perhitungan sesuai Rumus 3. Hasil perhitungan datap dilihat pada Tabel 3.

Tabel 3. Perhitungan Enkripsi

No	(P + K) mod 62
1	(30+40)mod 62 = 8
2	(61+6) mod 62 = 5
...	...
31	(52+40) mod 62 = 30
32	(56+6) mod 62 = 0

D. Konversi kembali indeks ke karakter
Setelah didapat indeks hasil perhitungan enkripsi, maka konversi indeks tersebut ke karakter. Hasil konversi dapat dilihat pada Tabel 4.

Tabel 4. Konversi Indeks Cipherteks ke Karakter

No	Indeks	Karakter
1	8	I
2	5	F
...	...	...
31	30	e
32	6	A

Setelah di konversi kembali ke karakter, maka didapat ciphertext:
IFfZm8haX7gBf5n7Fa5ZlAjxqcIb78eA

E. Proses Dekripsi
Sebelum melakukan dekripsi, seperti halnya proses enkripsi kunci harus diulangi hingga panjangnya sama dengan cipherteks, kemudian karakter dikonversi ke indeks. Dekripsi dilakukan dengan menerapkan Rumus 2 pada ciphertext dan key:

Ciphertext:
IFfZm8haX7gBf5n7Fa5ZlAjxqcIb78eA
Key:
oGl7Mho7d9

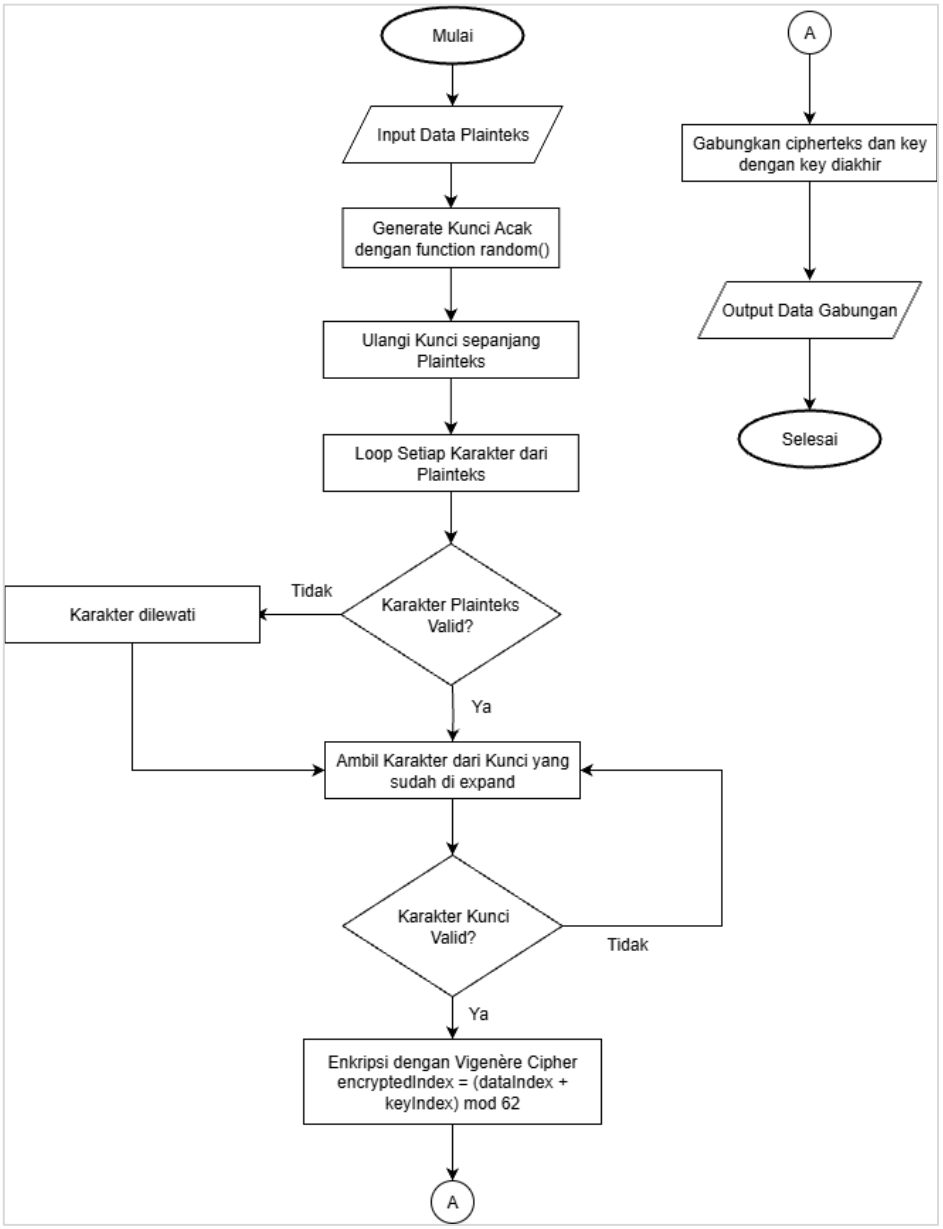
Untuk proses dekripsi dapat dilihat pada Tabel 5.

Tabel 5. Perhitungan Dekripsi

No	(C – K + 62) mod 62
1	(8-40+62)%62 = 30
2	(5-6+62)%62 = 61
...	...
31	(30-40+62)%62 = 52
32	(0-6+62)%62 = 56

Hasil Dekripsi Final:
e94cab3d482548babdca7480e5eee904

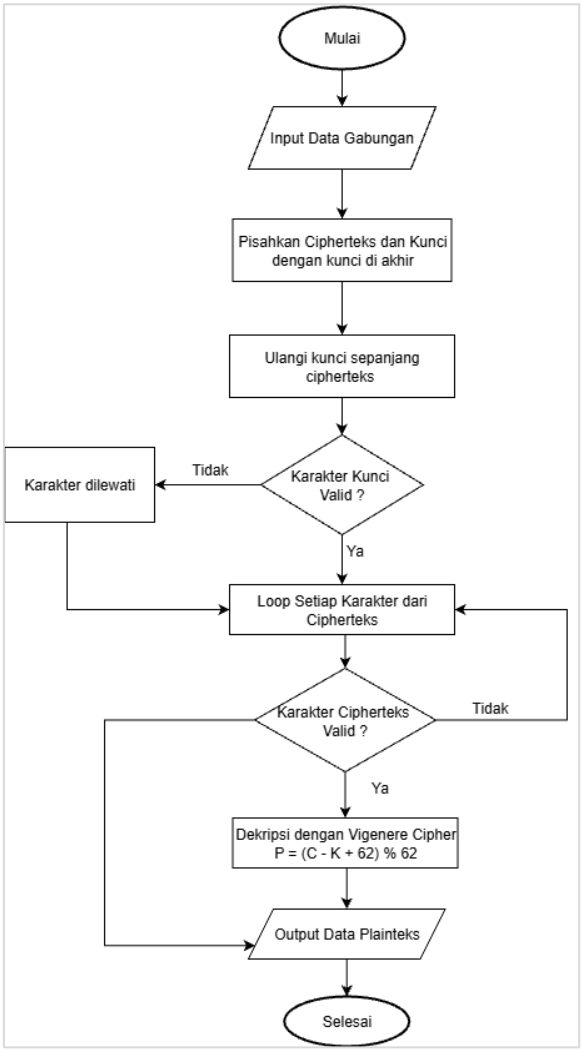
Komunikasi data dienkripsi menggunakan Vigenère Cipher, di mana setiap karakter plaintext disubstitusi dengan kunci berulang sesuai panjang teks. Proses enkripsi ini digambarkan pada Gambar 7.



Gambar 7. Proses Enkripsi

Proses enkripsi dimulai dari plaintext sebagai data asli. Sistem kemudian menyesuaikan panjang kunci dengan panjang plaintext melalui pengulangan karakter kunci. Setelah itu dilakukan enkripsi dengan Vigenère Cipher, di mana setiap karakter plaintext digeser sesuai nilai karakter kunci. Hasil pergeseran ini disusun menjadi ciphertext yang menjadi data terenkripsi. Diagram alir proses dekripsi menggunakan Vigenère Cipher dapat dilihat pada Gambar 8.

Proses dekripsi dimulai dari input berupa ciphertext dan kunci. Panjang kunci disesuaikan dengan panjang cipherteks, kemudian dilakukan proses dekripsi dengan Vigenère Cipher, yaitu menggeser setiap karakter ciphertext berdasarkan nilai kunci secara berulang. Hasil pergeseran tersebut akan mengembalikan data ke bentuk semula, yaitu plaintext sebagai data asli.

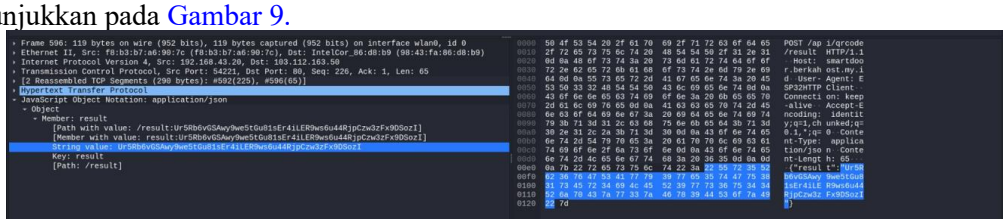


Gambar 8. Proses Dekripsi

Pengujian Algoritma Vigenere Cipher

Pengujian ini melihat kemampuan Vigenere Cipher dalam mengamankan transmisi data.

1. Pengujian *Sniffing*
   Hasil simulasi serangan sniffing dengan Wireshark menunjukkan bahwa meskipun data berhasil disadap, informasi yang tertangkap berupa ciphertext yang tidak dapat dipahami, sebagaimana ditunjukkan pada Gambar 9.



Gambar 9. Hasil Tangkapan Wireshark

   Terlihat pada Gambar 9 bahwa meskipun lalu lintas data berhasil disadap melalui sniffing, yang terekam hanyalah ciphertext sehingga penyerang tidak dapat membaca atau mengetahui isi informasi asli.
2. Pengujian Ketahanan Chipertext
   Untuk menilai ketahanan ciphertext terhadap serangan kriptanalisis, dilakukan pengujian menggunakan tools dCode dan CrypTool. Pengujian ini ditujukan untuk melihat apakah

ciphertext yang diperoleh dari hasil sniffing dapat dikenali atau dipecahkan melalui metode identifikasi publik maupun brute force. Data yang diuji terdiri atas dua jenis, yaitu ciphertext dari hasil pemindaian QR Code serta ciphertext dari instruksi perintah sistem.

a) Pengujian dengan Chiper Identifier

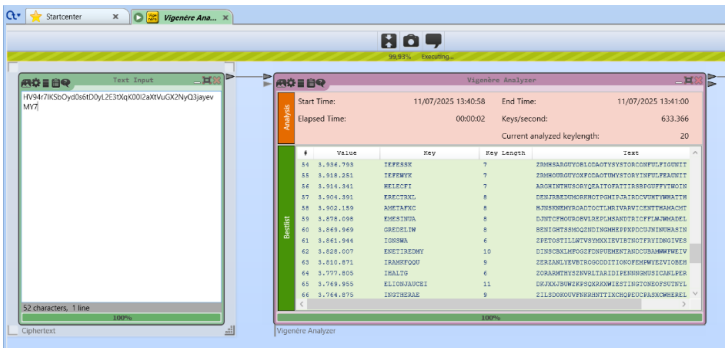
Dalam pengujian menggunakan Cipher Identifier, ciphertext hasil pemindaian QR Code dimasukkan ke menu Cipher Identifier untuk dianalisis. Selanjutnya, dCode menampilkan beberapa metode yang dianggap memiliki kecocokan dengan ciphertext tersebut. Proses pengujian ketahanan ini ditunjukkan pada Gambar 10.

Gambar 10. Pengujian Chiper Identifier

Setelah dilakukan percobaan sebanyak 10 kali, hasil menunjukkan bahwa hanya 30% ciphertext (3/10 data) berhasil dikenali sebagai hasil enkripsi menggunakan algoritma Vigenère Cipher dengan tingkat deteksi masing-masing 11,6%, 23,1%, dan 42,9%. Sementara itu, 70% ciphertext (7/10 data) tidak teridentifikasi sebagai Vigenère, yang kemungkinan disebabkan oleh keterbatasan panjang ciphertext, minimnya pola berulang, serta tingginya entropi karakter.

b) Pengujian dengan Brute Force

Langkah selanjutnya adalah menguji apakah cipherteks dapat dipecahkan menggunakan teknik brute force. Proses ini dilakukan dengan bantuan CrypTool. Proses pengujian dapat dilihat pada Gambar 11.



Gambar 11. Pengujian Brute Force

Setelah dilakukan percobaan sebanyak 10 kali. Hasil pengujian menunjukkan bahwa dari 10 data cipherteks *QR Code* yang diuji tidak ada yang berhasil di brute force

3. Pengujian Waktu Delay

Pengujian ini bertujuan untuk mengukur durasi delay pada keseluruhan komunikasi sistem, yaitu dari saat QR Code berhasil dipindai olehESP32-CAM hingga perintah dijalankan. Pengambilan data dilakukan dengan mencatat waktu pindai QR Code pada ESP32-CAM dan waktu eksekusi pada ESP32. Waktu delay didapat dengan mengurangi waktu eksekusi perintah dengan waktu pindai QR Code. Terdapat dua skenario pengujian, yaitu waktu aktivasi solenoid dan waktu penyalaan buzzer setelah pemindaian QR Code. Sebelum pengambilan data delay, dilakukan uji

kecepatan internet menggunakan Speedtest by Ookla dengan hasil unduh 11,39 Mbps, unggah 6,78 Mbps, dan ping 36 ms. Untuk hasilnya dapat dilihat pada Tabel 6.

Tabel 6. Pengujian Waktu Delay Selenoid Terbuka

No	Delay (s)
1	1,175
2	1,326
3	4,422
4	4,927
5	3,399
...	...
20	1,747
Rata-Rata	2,473

Dari Tabel 6, rata-rata delay pembukaan solenoid adalah 2,473 detik dengan rentang 0,193–5,840 detik, dipengaruhi kualitas jaringan dan beban server. Hasil delay penyalaan buzzer tercantum pada Tabel 7.

Tabel 7. Pengujian Waktu Delay Selenoid Terbuka

No	Delay (s)
1	7,907
2	3,353
3	4,847
4	3,22
5	3,383
...	...
20	3,62
Rata-Rata	3,491

Pada skenario buzzer, rata-rata waktu dari pemindaian hingga buzzer menyala adalah 3,491 detik, dengan delay tercepat 1,540 detik dan terlama 7,907 detik. Variasi ini kembali menunjukkan pengaruh kondisi jaringan dan latensi server.

SIMPULAN

Berdasarkan penelitian ini, penerapan metode Vigenère Cipher pada sistem Smart Door berbasis IoT terbukti mampu meningkatkan keamanan data. Hasil uji menunjukkan ciphertext selalu berbeda meskipun plaintext sama, dengan hanya 30% data hasil sniffing QR Code yang terdeteksi sebagai Vigenère Cipher dan semuanya gagal dipecahkan, termasuk satu ciphertext pada jalur instruksi server ke ESP32. Dari sisi kinerja, rata-rata delay tercatat 2,473 detik untuk membuka solenoid dan 3,491 detik untuk aktivasi buzzer, dengan delay tertinggi 7,907 detik, yang belum memenuhi standar real-time Tiphon (<1 detik) dan dipengaruhi terutama oleh kecepatan internet. Untuk pengembangan lebih lanjut, disarankan penggunaan teknik penyisipan kunci yang lebih kompleks agar distribusi kunci lebih aman, serta optimalisasi jaringan, server, dan mikrokontroler agar sistem lebih cepat, stabil, dan sesuai kebutuhan real-time.

DAFTAR PUSTAKA

[1] M. Syahdi Nasution, Muhammad Amin, and Wirda Fitriani, “Smart Sistem Iot Pemberi Pakan Ikan Dengan Menggunakan Metode Time Schedulling Berbasis Mikrokontroller,” *J. Zetroem*, vol. 5, no. 2, pp. 161–164, 2023, doi: 10.36526/ztr.v5i2.3082.

[2] M. A. Juniawan and A. H. Rismayana, “PROTOTIPE SMART DOOR LOCK BERBASIS INTERNET OF THINGS (STUDI KASUS LAB KOMPUTER POLITEKNIK TEDC BANDUNG),” vol. 8, no. 5, pp. 10856–10861, 2024.

[3] M. R. Abdahu, U. Ristian, and H. Hasfani, “Implementasi Smart Helmet Cabinet pada

Penyimpanan Helm Berbasis Mobile QR Code," vol. 9, no. 1, pp. 78–85, 2024.

[4] E. Lantz and T. Pierrou, "Security evaluation of a smart door lock system," 2022.

[5] H. Fereidouni, O. Fadeitcheva, and M. Zalai, "IoT and Man-in-the-Middle Attacks," 2023, [Online]. Available: http://arxiv.org/abs/2308.02479

[6] C. Silva, V. A. Cunha, J. P. Barraca, and R. L. Aguiar, "Analysis of the Cryptographic Algorithms in IoT Communications," *Inf. Syst. Front.*, vol. 26, no. 4, pp. 1243–1260, 2024, doi: 10.1007/s10796-023-10383-9.

[7] H. A. Wahid, J. Maulindar, and A. I. Pradana, "Rancang Bangun Sistem Penyiraman Tanaman Otomatis Aglonema Berbasis IoT Menggunakan Blynk dan NodeMCU 32," *Innov. J. Soc. Sci. Res.*, vol. 3, no. 2, pp. 6265–6276, 2023.

[8] D. Setiawan, H. Jaya, S. Nurarif, T. Syahputra, and M. Syahril, "Implementasi Esp32-Cam Dan Blynk Pada Wifi Door Lock System Menggunakanteknik Duplex," *J. Sci. Soc. Res.*, vol. 5, no. 1, p. 159, 2022, doi: 10.54314/jssr.v5i1.807.

[9] Z. Avista and O. Fahlovi, "Rancang Bangun Smart Door Access Berbasis Fingerprint untuk Keamanan Ruang Laboratorium," *Venus J. Publ. Rumpun Ilmu Tek.*, vol. 2, no. 1, pp. 1–13, 2024, [Online]. Available: https://doi.org/10.61132/venus.v2i1.73

[10] R. Juliansyah, E. Fitriani, N. Paramita, and ..., "Rancang Bangun Sistem Kontrol Motor Feeder dan Monitoring Pakan Ikan Nila Berbasis Smart Relay Zelio," *J. Pendidik. ...*, vol. 8, pp. 11157–11167, 2024, [Online]. Available: https://www.jptam.org/index.php/jptam/article/view/14054%0Ahttps://www.jptam.org/index.php/jptam/article/download/14054/10820

[11] I. A. Deswiyan, S. Solikhun, S. Sumarno, P. Poningsih, and S. R. Andani, "Rancang Bangun Alat Pendeteksi Ketinggian Air dan Alarm Pemberitahuan Antisipasi Datangnya Banjir Berbasis Arduino Uno," *J. Penelit. Inov.*, vol. 1, no. 2, pp. 155–164, 2021, doi: 10.54082/jupin.23.

[12] W. Muthia Kansha, Saherih, and Muchlis, "Analisis Perbandingan Struktur dan Performa Framework Codeigniter dan Laravel dalam Pengembangan Web Application," *J. Tek. Inform. STMIK Antar Bangsa*, vol. 9, no. 1, pp. 25–31, 2023.

[13] A. Herlan, I. Fitri, and R. Nuraini, "Rancang Bangun Sistem Monitoring Data Sebaran Covid-19 Secara Real-Time menggunakan Arduino Berbasis Internet of Things (IoT)," *J. JTIK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 5, no. 2, p. 206, 2021, doi: 10.35870/jtik.v5i2.212.

[14] N. Santi and Y. Mulyanto, "Analisis Kelayakan Jaringan Komputer Menggunakan Alat Sniffing Dan Intrusion Detection System (Ids)," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 4, pp. 6141–6147, 2024, doi: 10.36040/jati.v8i4.10079.

[15] M. A. Rizkiawan, E. Kurniawan, H. Ramza, P. Takumi, T. Elektro, and P. Loss, "ANALISIS QUALITY OF SERVICE JARINGAN NIRKABEL MENGGUNAKAN," vol. 8, no. 5, pp. 9876–9882, 2024.

[16] I. Murni, A. S. Br pa, B. R. Lubis, and A. Ikhwan, "Pengamanan Pesan Rahasia dengan Algoritma Vigenere Cipher Menggunakan PHP," *J. Educ.*, vol. 5, no. 2, pp. 3466–3476, 2023, doi: 10.31004/joe.v5i2.1027.

[17] D. Arfandy, M. Simanjuntak, and T. Pasaribu, "Penerapan Metode Vigenere Chiper untuk Mengamankan Data Text," *JUKI J. Komput. dan Inform.*, vol. 4, 2022.