Implementasi One-Time Password (OTP) dengan untuk Enkripsi QR Code pada Distribusi Subsidi Bahan Bakar Pertamina

Rayhan Nugraha Putra (18221149)
Program Studi Sistem dan Teknologi Informasi
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
18221149@std.stei.itb.ac.id

***Abstract*— Makalah ini mengusulkan pendekatan baru untuk meningkatkan keamanan dan aksesibilitas sistem distribusi subsidi solar Pertamina. Pendekatan tersebut mengimplementasikan One-Time Password (OTP) yang terintegrasi dengan enkripsi kode QR. Sistem ini memanfaatkan kemajuan dalam bidang kriptografi dan teknologi seluler untuk mengatasi keterbatasan metode distribusi subsidi sebelumnya. Sistem yang diusulkan dapat memberikan keamanan yang ditingkatkan dengan OTP menyediakan lapisan autentikasi sekali pakai, mengurangi risiko akses tidak sah ke manfaat subsidi. Makalah membahas detail teknis penerapan sistem OTP dan enkripsi kode QR, serta manfaat sistem untuk meningkatkan proses distribusi yang lebih aman, mudah diakses, dan efisien sesuai dengan Perpres No. 191 Tahun 2014.**

Keywords—One-Time Password (OTP); QR Code; Enkripsi; AES; SHA-3; Subsidi Bahan Bakar; Kriptografi

I. Introduction

Pemerintah Indonesia, melalui Pertamina, menerapkan program subsidi solar untuk memastikan keterjangkauan bahan bakar bagi penerima manfaat yang ditargetkan. Program ini berperan penting dalam mendukung kehidupan dan aktivitas ekonomi, terutama di daerah terpencil. Namun, memastikan distribusi subsidi yang aman dan efisien tetap menjadi tantangan penting [1, 2].

Metode distribusi subsidi tradisional rentan terhadap penipuan dan manipulasi. Masalah seperti penggunaan kartu bahan bakar palsu atau akses tidak sah ke manfaat subsidi dapat menyebabkan kerugian finansial dan menghambat efektivitas program [1]. Selain itu, ketergantungan pada metode distribusi fisik bisa jadi rumit dan membatasi aksesibilitas, terutama di daerah dengan infrastruktur terbatas [2].

Makalah ini mengusulkan pendekatan baru untuk meningkatkan keamanan dan aksesibilitas sistem distribusi subsidi solar Pertamina. Makalah ini membahas implementasi One-Time Password (OTP) yang terintegrasi dengan enkripsi kode QR. Pendekatan ini memanfaatkan kemajuan dalam bidang kriptografi dan teknologi seluler untuk mengatasi keterbatasan metode yang ada.

Sistem yang diusulkan dapat memberikan keamanan yang ditingkatkan dengan OTP menyediakan lapisan autentikasi sekali pakai, mengurangi risiko akses tidak sah ke manfaat subsidi.

Makalah ini membahas detail teknis tentang penerapan sistem OTP dan enkripsi kode QR yang diusulkan. Selain itu, makalah ini juga akan membahas prinsip-prinsip kriptografi di balik pendekatan tersebut, alur kerja operasionalnya, serta potensi pertimbangan keamanan.

Selain itu, kami menganalisis manfaat sistem untuk program subsidi solar Pertamina, yang berkontribusi pada proses distribusi yang lebih aman, mudah diakses, dan efisien. Ini sejalan dengan peraturan pemerintah Indonesia, seperti Perpres No. 191 Tahun 2014, yang menekankan pentingnya transparansi dan akuntabilitas dalam program subsidi [3].

II. Landasan Teori

A. One Time Password (OTP)

One-Time Password (OTP) adalah kode rahasia yang hanya berlaku untuk satu kali penggunaan. OTP digunakan sebagai lapisan keamanan tambahan dalam sistem autentikasi untuk mencegah akses tidak sah. OTP biasanya dihasilkan secara acak dan dikirimkan kepada pengguna melalui berbagai cara, seperti SMS, email, atau aplikasi autentikasi. Keuntungan OTP: (1) Meningkatkan keamanan: OTP mengurangi risiko akses tidak sah karena kode hanya valid untuk satu kali penggunaan. (2) Mudah digunakan: OTP mudah digunakan dan tidak memerlukan perangkat lunak atau perangkat keras khusus. (3) Cocok untuk berbagai aplikasi: OTP dapat digunakan untuk berbagai aplikasi, seperti autentikasi online, login perangkat, dan transaksi keuangan.

Jenis OTP:

- OTP berbasis waktu: OTP dihasilkan berdasarkan waktu saat ini dan hanya valid untuk jangka waktu tertentu.
- OTP berbasis HMAC: OTP dihasilkan menggunakan algoritma HMAC (Hash Message Authentication Code) dan kunci rahasia yang dibagikan antara pengguna dan server.

- OTP berbasis QR code: OTP dapat ditampilkan dalam QR code dan dipindai oleh pengguna untuk autentikasi.

B. QR Code

QR Code (Quick Response Code) adalah kode dua dimensi yang dapat dipindai dengan perangkat pintar untuk mengakses informasi yang tersimpan di dalamnya. QR code dapat digunakan untuk berbagai keperluan, seperti: (1) Berbagi informasi: QR code dapat digunakan untuk berbagi informasi seperti URL, alamat email, nomor telepon, atau teks biasa. (2) Pembayaran: QR code dapat digunakan untuk melakukan pembayaran di toko atau online. (3) Autentikasi: QR code dapat digunakan untuk autentikasi, seperti memindai kode untuk login ke aplikasi atau situs web.

C. Perpres No. 191 Tahun 2014

Peraturan Presiden Republik Indonesia Nomor 191 Tahun 2014 tentang Penyediaan, Pendistribusian, dan Harga Jual Eceran Bahan Bakar Minyak mengatur tentang penyediaan, pendistribusian, dan harga jual eceran bahan bakar minyak (BBM). Peraturan ini bertujuan untuk memastikan ketersediaan BBM yang merata dan terjangkau bagi masyarakat, serta untuk melindungi konsumen dari fluktuasi harga BBM yang tidak terkendali.

Beberapa poin penting dalam Perpres No. 191 Tahun 2014:

- Penyediaan BBM: Perpres ini mengatur tentang siapa yang berhak menyediakan BBM, yaitu badan usaha yang ditunjuk oleh pemerintah.
- Pendistribusian BBM: Perpres ini mengatur tentang cara pendistribusian BBM, yaitu melalui jaringan SPBU yang tersebar di seluruh Indonesia.
- Harga Jual Eceran BBM: Perpres ini mengatur tentang harga jual eceran BBM yang ditetapkan oleh pemerintah.

D. Fungsi HASH SHA-3 (Keccak)

SHA-3 (Secure Hash Algorithm 3) adalah fungsi hash kriptografi yang dikembangkan oleh National Institute of Standards and Technology (NIST) melalui sebuah kompetisi terbuka. Fungsi hash ini dirancang sebagai komplementer untuk SHA-1 dan SHA-2, serta menjadi standar fungsi hash baru yang lebih aman dan lebih efisien. Pemenang kompetisi SHA-3 adalah Keccak, yang dirancang oleh tim Guido Breton, Joan Daemen, Michaël Peeters, dan Gilles Van Assche.

Cara Kerja SHA-3 (Keccak) sebagaimana Fig.1 adalah sebagai berikut:

1. Konstruksi Spons (Sponge Construction):

   SHA-3 menggunakan konstruksi spons, yang berbeda dari pendekatan fungsi kompresi yang digunakan oleh fungsi hash sebelumnya. Konstruksi spons terdiri dari dua fase: penyerapan (absorbing) dan pemerasan (squeezing).

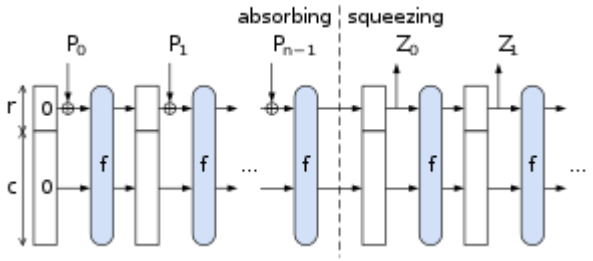

Fig. 1. Konstruksi spons dari fungsi hash dimana P_i adalah input dan Z_i adalah output hash.

2. Inisialisasi:

   Tentukan panjang digest yang diinginkan (d bit) dan panjang blok (r bit). Inisialisasi state (S) dengan nilai 0 sepanjang b = r + c bit, di mana c adalah kapasitas (capacity) yang bertujuan untuk mencegah kolisi.

3. Fase Penyerapan (Absorbing):

   Pesan masukan (M) dipadding (ditambah bit-bit pengganjal) sehingga panjangnya habis dibagi dengan r. Pesan yang sudah dipadding (P) dipotong menjadi blok-blok P_i berukuran r bit. Setiap blok P_i di-XOR dengan r bit pertama dari state S, lalu dimasukkan ke dalam fungsi permutasi f untuk menghasilkan state baru S.

4. Fase Pemerasan (Squeezing):

   Inisialisasi message digest (Z) dengan string kosong. Selama panjang Z belum sama dengan d, r bit pertama dari state S disambungkan (append) ke Z. Jika panjang Z masih belum sama dengan d, masukkan state S ke dalam fungsi permutasi f untuk menghasilkan state baru S, lalu ulangi langkah sebelumnya.

5. Keluaran:

   Setelah fase pemerasan selesai, message digest (Z) dengan panjang d bit diperoleh sebagai keluaran fungsi hash SHA-3 (Keccak).

Kelebihan utama dari SHA-3 (Keccak) adalah penggunaan konstruksi spons yang berbeda dari pendekatan tradisional, sehingga memberikan keamanan yang lebih baik dan memungkinkan panjang digest yang fleksibel. Selain itu, SHA-3 juga dirancang untuk memiliki kinerja yang baik dan dapat diimplementasikan secara efisien pada berbagai platform. [5]

E. Fungsi AES

Advanced Encryption Standard (AES) adalah algoritma kriptografi simetri berbasis cipher blok yang digunakan untuk mengamankan data dengan melakukan proses enkripsi dan dekripsi. AES ditetapkan sebagai standar enkripsi oleh National Institute of Standards and Technology (NIST) pada tahun 2001 setelah melalui proses kompetisi yang dimenangkan oleh algoritma Rijndael yang dirancang oleh Joan Daemen dan Vincent Rijmen dari Belgia

AES menggunakan operasi substitusi dan permutasi untuk mengenkripsi atau mendekripsi data. Proses enkripsi dan

dekripsi dilakukan dalam sejumlah putaran (rounds) tertentu, bergantung

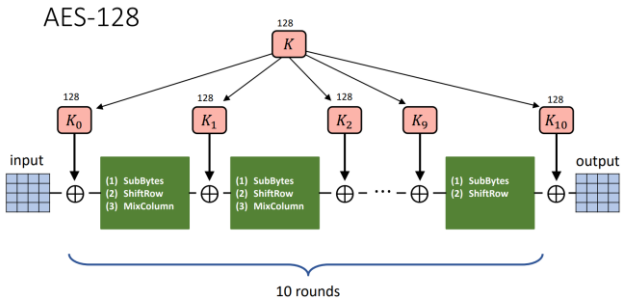

Fig. 2. Ilustrasi proses enkripsi AES.

pada panjang kunci. Panjang kunci dapat dipilih sebesar 128 bit, 192 bit, atau 256 bit sedangkan ukuran blok data yang dienkripsi/dekripsi adalah 128 bit. Setiap putaran menggunakan kunci putaran (round key) yang berbeda, yang diturunkan dari kunci utama melalui proses ekspansi kunci. Berikut adalah tahapan enkripsi AES:

1. AddRoundKey: Plainteks di-XOR dengan kunci putaran awal.
2. Untuk setiap putaran (Nr - 1 putaran):
    a. SubBytes: Substitusi byte-byte dalam state dengan menggunakan tabel substitusi (S-box).
    b. ShiftRows: Pergeseran baris-baris dalam state secara wrapping (siklik).
    c. MixColumns: Operasi pengacakan data di setiap kolom state.
    d. AddRoundKey: State di-XOR dengan kunci putaran berikutnya.
3. Putaran Terakhir:
    a. SubBytes
    b. ShiftRows
    c. AddRoundKey

Tahapan dekripsi AES mirip dengan enkripsi, tetapi dengan operasi yang dibalik dan menggunakan kunci putaran dalam urutan terbalik.

Kunci putaran dibangkitkan dari kunci eksternal melalui proses ekspansi kunci yang kompleks, melibatkan operasi substitusi, pergeseran, dan XOR dengan konstanta putaran (Rcon). Beberapa operasi dalam AES, seperti substitusi byte dan pembangkitan kunci putaran, melibatkan operasi aritmatika dalam Galois Field GF(2^8), yang merupakan medan berhingga biner dengan operasi penjumlahan dan perkalian yang didefinisikan secara khusus.

AES dikenal sebagai algoritma kriptografi yang kuat dan aman, serta banyak digunakan dalam berbagai aplikasi yang membutuhkan perlindungan data. Keamanan AES bergantung pada panjang kunci yang digunakan, dengan kunci yang lebih panjang memberikan keamanan yang lebih tinggi. [4]

III. Analisa Kondisi As-Is Sistem

A. *Deskripsi Sistem As-Is*

Sistem Subsidi Tepat Pertamina adalah program yang dirancang untuk memberikan subsidi bahan bakar bersubsidi kepada konsumen yang memenuhi syarat. Sistem ini menggunakan kombinasi pendaftaran online dan verifikasi fisik untuk memastikan bahwa hanya konsumen yang berhak yang menerima subsidi.

B. *Alur Penggunaan Sistem*

Secara garis besar terdapat dua alur proses yang harus dilalui oleh pengguna untuk melakukan klaim subsidi bahan bakar. Alur proses pendaftaran hanya perlu dilakukan untuk setiap kendaraan yang ingin menerima subsidi dan proses klaim dilakukan setiap kali kendaraan ingin melakukan klaim subsidi di Stasiun Pengisian Bahan bakar Umum (SPBU).

Alur proses pendaftaran calon penerima subsidi:

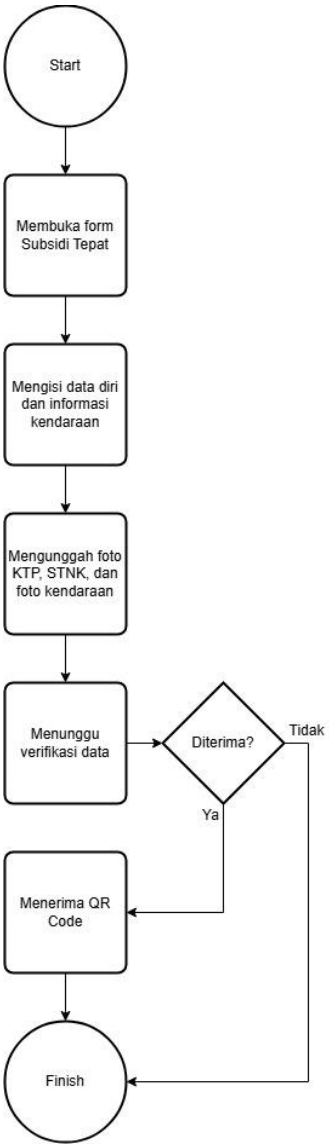

Fig. 3. Diagram Alir Pendaftaran Calon Penerima Subsidi Bahan Bakar.

1. Konsumen mengisi form Subsidi Tepat.
2. Konsumen mengisi data diri dan informasi kendaraan.
3. Konsumen mengunggah foto KTP (kartu tanda penduduk), STNK (surat tanda nomor kendaraan), dan foto kendaraan.

4. Konsumen menunggu verifikasi data oleh Pertamina.
5. Setelah verifikasi disetujui, konsumen menerima QR code.

Setelah kendaraan terverifikasi berhak mendapatkan subsidi bahan bakar, maka proses klaim subsidi tersebut adalah sebagai berikut:

1. Konsumen mengunjungi SPBU (stasiun pengisian bahan bakar umum) yang berpartisipasi dalam program subsidi tepat.
2. Konsumen menunjukkan QR code kepada petugas SPBU.

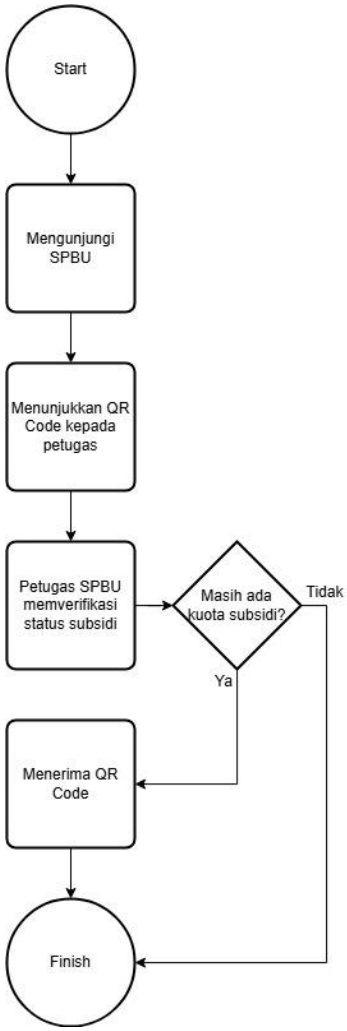

Fig. 4. Diagram Alir Proses Klaim Subsidi Bahan Bakar.

3. Petugas SPBU memindai QR code dan memasukkan jumlah bahan bakar yang ingin dibeli.
4. Sistem melakukan pemotongan kuota subsidi harian bahan bakar.
5. Konsumen menerima subsidi bahan bakar sesuai dengan haknya.

C. Evaluasi Kondisi As-Is

Mempertimbangkan kesesuaian sistem distribusi subsidi dengan berbagai protokol keamanan dan aplikasinya di kondisi nyata masyarakat, Terdapat beberapa evaluasi mengenai sistem yang ada saat ini.

Kelebihan:

1. Sistem ini lebih transparan dan akuntabel dibandingkan dengan sistem subsidi sebelumnya.
2. Sistem ini membantu menargetkan subsidi kepada konsumen yang benar-benar membutuhkan.
3. Sistem ini membantu mengurangi penyalahgunaan subsidi.

Kekurangan:

1. Sistem ini masih dalam tahap awal implementasi dan mungkin ada beberapa kendala teknis.
2. Proses pendaftaran dan verifikasi data bisa memakan waktu lama.
3. Tidak semua SPBU berpartisipasi dalam program ini.
4. Ada potensi penyalahgunaan QR code.

D. Analisis Kondisi Ideal Sistem

Peningkatan desain sistem dapat dilakukan dengan merancang sistem subsidi tepat Pertamina yang ideal dengan fokus pada keamanan kode QR yang dinamis untuk melindungi setiap transaksi, menjamin identitas penerima subsidi, dan mencegah pencurian kode QR dengan rincian sebagai berikut.

1) Pendaftaran dan Verifikasi Pengguna:

- Pendaftaran akun MyPertamina diperkuat dengan verifikasi biometrik (misalnya, sidik jari, pengenalan wajah) untuk memastikan identitas pengguna.
- Verifikasi data kendaraan dilakukan dengan inspeksi fisik oleh petugas terpercaya di lokasi yang ditentukan.

2) Mekanisme Kode QR Dinamis:

- Implementasi kode QR dinamis yang unik untuk setiap transaksi subsidi. Kode QR dinamis terhubung dengan akun pengguna, data kendaraan, dan kuota subsidi yang tersisa. Kode QR dinamis memiliki masa berlaku singkat (misalnya, 5 menit) setelah diaktifkan. Kode QR dinamis terintegrasi dengan sistem OTP (One Time Password) untuk menambah lapisan keamanan.

3) Keamanan Transaksi:

- Implementasi enkripsi multi-level untuk melindungi data pengguna dan transaksi subsidi.
- Penerapan sistem pemantauan real-time untuk mendeteksi aktivitas mencurigakan dan mencegah penipuan.
- Integrasi dengan sistem pelaporan dan investigasi untuk menindaklanjuti transaksi yang bermasalah.
- Edukasi dan sosialisasi kepada pengguna tentang keamanan kode QR dan modus penipuan yang marak terjadi.

4) Peningkatan Infrastruktur:

- Peningkatan kapasitas server dan jaringan untuk memastikan kelancaran sistem, terutama saat peak season.
- Penggunaan perangkat keras dan perangkat lunak yang terjamin keamanannya.
- Penerapan audit dan pengujian keamanan secara berkala untuk mendeteksi kerentanan sistem.
- Kerjasama dengan pakar keamanan siber untuk meningkatkan ketahanan sistem terhadap serangan.

5) Tata Kelola dan Pengawasan:

- Pembentukan tim khusus yang bertanggung jawab atas pengelolaan dan pengawasan sistem subsidi tepat.
- Penerapan kebijakan dan prosedur yang jelas untuk mengelola akses data, perubahan sistem, dan penanganan insiden keamanan.
- Pelaksanaan audit internal dan eksternal secara berkala untuk memastikan kepatuhan terhadap regulasi dan standar keamanan.
- Kerjasama dengan lembaga penegak hukum untuk menindak pelanggaran dan penyalahgunaan sistem.

E. Gap Analysis

Analisa kondisi real di dunia nyata dan evaluasi keberjalanan sistem saat ini menghasilkan beberapa usulan keamanan, efisiensi, dan aksesibilitas distribusi subsidi bahan bakar. Usulan-usulan ini merupakan hasil perbandingan antara kondisi sistem saat ini (As-is) dengan kondisi sistem idel (To-be). Untuk menjaga relevansi dengan topik makalah, maka perbandingan dibatasi hanya pada sistem QR code subsidi.

TABLE 1. GAP ANALYSIS SISTEM AS-IS DENGAN TO-BE

Aspek	**Sistem As-Is**	**Rancangan Kondisi Ideal (To-Be)**	**Gap**
Jenis Kode QR	Statis	Dinamis	- Kode QR mudah ditiru dan disalahgunakan. - Kode QR dapat digunakan berulang kali jika dicuri.
Masa Berlaku Kode QR	Tidak ada	Pendek (misalnya, 5 menit)	- Kurangnya kontrol terhadap penggunaan kode QR. - Peningkatan risiko penyalahgunaan kode QR yang dicuri.
Verifikasi Tambahan	Tidak ada	OTP (One Time Password)	- Keamanan transaksi hanya mengandalkan kode QR yang rentan. - Kemungkinan transaksi curang oleh pihak yang mengetahui kode QR pengguna.

IV. RANCANGAN TO-BE SISTEM

A. Deskripsi Sistem To-Be

Sistem subsidi tepat Pertamina yang baru akan mengimplementasikan kode QR dinamis yang dilindungi dengan One-Time Password (OTP) dan enkripsi AES untuk meningkatkan keamanan dan mencegah penyalahgunaan subsidi bahan bakar. Kode QR akan terintegrasi dengan akun pengguna, data kendaraan, dan kuota subsidi yang tersisa, serta memiliki masa berlaku singkat setelah diaktifkan.

Sistem ini akan menggunakan pendekatan keamanan multi-layer, di mana kode QR yang terenkripsi akan dikirimkan ke server Pertamina bersama dengan OTP. Server akan melakukan dekripsi kode QR menggunakan OTP yang diterima, lalu memverifikasi data penerima subsidi dengan melakukan pencocokkan nilai hash dari identitas penerima dengan rekaman data di server.

B. Mekanisme Sistem To-Be

Secara garis besar, tidak ada perubahan proses pendaftaran antara sistem As-Is dengan sistem To-Be.

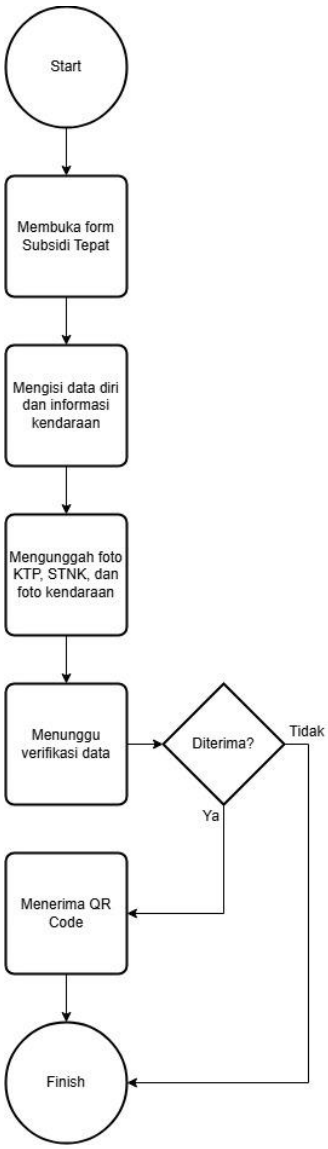

Fig. 5. Diagram Alir Pendaftaran Calon Penerima Subsidi Bahan Bakar pada sistem To-Be.

1. Konsumen mengisi form Subsidi Tepat.
2. Konsumen mengisi data diri dan informasi kendaraan.
3. Konsumen mengunggah foto KTP (kartu tanda penduduk), STNK (surat tanda nomor kendaraan), dan foto kendaraan.
4. Konsumen menunggu verifikasi data oleh Pertamina.
5. Setelah verifikasi disetujui, konsumen menerima QR code.

Proses klaim subsidi:

1. Pengguna meminta kode QR dinamis dari server Pertamina.
2. Server Pertamina menghasilkan kode QR dinamis yang terenkripsi dengan AES dan OTP dengan masa berlaku singkat.
3. Pengguna memindai kode QR di SPBU untuk melakukan klaim subsidi.
4. SPBU mengirimkan kode QR dari pengguna ke server Pertamina.
5. Server Pertamina melakukan dekripsi kode QR menggunakan OTP, lalu memverifikasi identitas penerima subsidi dengan mencocokkan nilai hash dari identitas penerima dengan rekaman data di server.

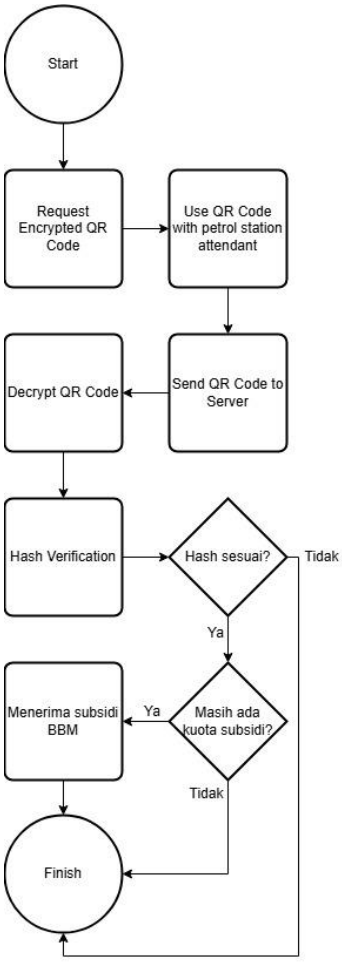

Fig. 6. Diagram Alir Klaim Subsidi Bahan Bakar pada sistem To-Be.

6. Jika verifikasi berhasil, subsidi disetujui. Jika tidak, subsidi ditolak.

Skenario alternatif:

1. Jika kuota subsidi untuk pengguna tersebut telah habis, meskipun verifikasi identitas berhasil, transaksi akan ditolak dan pengguna tidak dapat melakukan klaim subsidi.
2. Jika durasi OTP telah habis saat server Pertamina menerima kode QR dan OTP, dekripsi tidak dapat dilakukan dan permohonan subsidi akan ditolak.
3. Jika terdapat masalah lain seperti data pengguna tidak valid atau terjadi kesalahan sistem, permohonan subsidi juga dapat ditolak oleh server Pertamina.

Adapun alur pergerakan data pengguna di perangkat pengguna dan server pertamina adalah sebagai berikut:

1. Identitas penerima subsidi dienkripsi dengan algoritma AES, lalu ditambahkan OTP untuk membentuk kode QR terenkripsi.
2. Kode QR terenkripsi dikirimkan ke server Pertamina.
3. Server Pertamina melakukan dekripsi kode QR menggunakan OTP yang diterima.
4. Setelah didekripsi, server menghitung nilai hash dari identitas penerima subsidi.
5. Nilai hash tersebut diverifikasi dengan rekaman data di server Pertamina untuk memastikan keabsahan identitas penerima subsidi.
6. Jika verifikasi berhasil, subsidi disetujui. Jika tidak, subsidi ditolak.

V. Mockup Implementasi Sistem

Untuk menguji kinerja sistem dan membuktikan bahwa solusi ini dapat diimplementasi, maka dibuat sebuah program sederhana yang berfungsi untuk membuat kode QR terenkripsi dengan Time Based One-Time Password (TOTP)

1) Pembuatan kode OTP

Pada simulasi ini, kode OTP akan dibuat setiap menit oleh sistem (server pertamina).

TABLE 2. Algoritma Pembangkitan Time Based OTP

```
import hashlib
import hmac
import datetime
import time

def generate_totp(secret_key):
    time_interval = 60
    current_time = time.time()
    time_counter = int(current_time / time_interval)
    time_counter_bytes = time_counter.to_bytes(8, byteorder='big')

    hmac_result = hmac.new(secret_key.encode(), time_counter_bytes, hashlib.sha1).digest()

    offset = hmac_result[-1] & 0x0F

    truncated_hash = hmac_result[offset:offset+4]

    truncated_hash_int = int.from_bytes(truncated_hash, byteorder='big')

    totp = truncated_hash_int % (10 ** 6)

    return '{:06d}'.format(totp)

SECRET_KEY = "ASOFTANDSWEETSENSATIONSOFLUFFY"

otps = []
for i in range(5):
    otp = generate_totp(SECRET_KEY)
    print(f"Generated OTP {i+1}: {otp}")
    otps.append(otp)
    if i < 4:
        time.sleep(60)

print("Generated OTPs:", otps)
```

Hasil output sistem dari program ini adalah sebagaimana Fig.7.

```
Generated OTP 1: 122453
Generated OTP 2: 476537
Generated OTP 3: 908231
Generated OTP 4: 530955
Generated OTP 5: 126232
Generated OTPs: ['122453', '476537', '908231', '530955', '126232']
```

Fig.7. Hasil luaran sistem Step 1.

2) Enkripsi Data Pengguna dengan Kode OTP

Asumsi data pengguna terdiri atas «user_id» dan «vehicle_id»

TABLE 3. Algoritma Enkripsi Data Pengguna

```
import json
import os
from cryptography.hazmat.primitives.ciphers import Cipher, algorithms, modes
from cryptography.hazmat.primitives import padding

# User data
user_data = {
    "user_id": 12345,
    "vehicle_id": 67890
}
user_data_json = json.dumps(user_data).encode()

print("User Data (Before Encryption):", user_data)

# Encrypt Data
```

```
def encrypt_data(data, key):
    iv = os.urandom(16)
    cipher = Cipher(algorithms.AES(key),
modes.CBC(iv))
    encryptor = cipher.encryptor()

    padder = padding.PKCS7(128).padder()
    padded_data = padder.update(data) +
padder.finalize()

    encrypted_data = encryptor.update(padded_data)
+ encryptor.finalize()
    return iv + encrypted_data

# Generate a 32-byte key from the fifth OTP
key = otps[4].ljust(32)[:32].encode()

encrypted_data = encrypt_data(user_data_json, key)
print("Encrypted Data (hex):",
encrypted_data.hex())
```

Hasil output sistem dari program ini adalah sebagaimana Fig.8 .

```
User Data (Before Encryption):
{'user_id': 12345, 'vehicle_id': 67890}

Encrypted Data (hex):
4c81f2b35c45d7896ca516f9241ec37882ad341aabc872ddc
0a2d6dfdee4067493c1608d414a77bf2eb80f70d3639b3be3
0ee9379021fd321aed6fc603be4bc5
```

Fig.8. Hasil luaran sistem Step 2.

3) Pembuatan Kode QR dengan data terenkripsi

TABLE 4. Algoritma Pembuatan Kode QR

```
import qrcode

def create_qr_code(data, filename):
    qr = qrcode.QRCode(
        version=1,

error_correction=qrcode.constants.ERROR_CORRECT_L,
        box_size=10,
        border=4,
    )
    qr.add_data(data)
    qr.make(fit=True)

    img = qr.make_image(fill='black',
back_color='white')
    img.save(filename)

create_qr_code(encrypted_data.hex(),
'encrypted_data_qr.png')
```

QR Code yang telah dibuat adalah Fig.9.

Fig. 9. Kode QR yang berisi data pengguna terenkripsi

4) Dekripsi data dari QR code

TABLE 5. Algoritma Dekripsi Kode QR

```
def decrypt_data(encrypted_data, key):
    iv = encrypted_data[:16]  # Extract IV from
the beginning
    encrypted_data = encrypted_data[16:]  # The
rest is the encrypted data

    cipher = Cipher(algorithms.AES(key),
modes.CBC(iv))
    decryptor = cipher.decryptor()

    decrypted_padded_data =
decryptor.update(encrypted_data) +
decryptor.finalize()

    # Unpadding data
    unpadder = padding.PKCS7(128).unpadder()
    decrypted_data =
unpadder.update(decrypted_padded_data) +
```

```
unpadder.finalize()

    return decrypted_data.decode()

decrypted_data =
decrypt_data(bytes.fromhex(encrypted_data.hex()),
key)
print("Decrypted Data:", decrypted_data)
```

Hasil dekripsi.

```
Decrypted Data: {"user_id": 12345, "vehicle_id":
67890}
```

Fig. 10. Hasil Dekripsi kode QR

5) Verifikasi data dekripsi dengan hash database

Untuk meningkatkan efisiensi verifikasi data, maka pengecekan di server dilakukan dengan mencocokkan hasil hash.

TABLE 6. Algoritma Verifikasi dan Hashing

```
import hashlib
import json
import pickle

def compute_sha3_hash(data):
    data_str = str(data)
    data_str = data_str.replace('"', "'")
    return
hashlib.sha3_256(data_str.encode()).hexdigest()

# Compute SHA-3 hashes for original and decrypted
data
original_hash = compute_sha3_hash(user_data)
decrypted_hash = compute_sha3_hash(decrypted_data)

print('Original Hash:', original_hash)
print('Decrypted Hash:', decrypted_hash)

if original_hash == decrypted_hash:
    print('Data Integrity Verified.')
else:
    print('Data Integrity Verification Failed.')
```

Berikut adalah hasil langkah ini.

```
Original Hash:
5cfaf0c1200c7453c9e29abc
7ee4cc4b9a98049c207cba6bb18dbe5cf08b3865
Decrypted Hash:
5cfaf0c1200c7453c9e29abc
7ee4cc4b9a98049c207cba6bb18dbe5cf08b3865
Data Integrity Verified.
```

Fig.11. Hasil verifikasi menggunakan hash

VI. KESIMPULAN

Makalah ini mengusulkan implementasi One-Time Password (OTP) yang terintegrasi dengan enkripsi kode QR untuk meningkatkan keamanan dan aksesibilitas sistem distribusi subsidi solar Pertamina. Sistem ini menggunakan pendekatan keamanan multi-lapis, di mana kode QR yang terenkripsi dikirimkan ke server Pertamina bersama dengan OTP. Server melakukan dekripsi kode QR menggunakan OTP yang diterima, lalu memverifikasi data penerima subsidi dengan melakukan pencocokkan nilai hash dari identitas penerima dengan rekaman data di server. Sistem yang diusulkan dapat memberikan solusi yang lebih aman dan efisien dalam proses distribusi subsidi, sejalan dengan peraturan pemerintah terkait transparansi dan akuntabilitas program subsidi.

REFERENCES

[1] Pertamina, "FAQ Subsidi Tepat," [Online]. Available: https://mypertamina.id/faq-subsidi-tepat [Accessed: 12-Jun-2024].

[2] "Panduan BBM," [Online]. Available: https://storage.googleapis.com/assets-mpv/web-subsidi/Panduan%20BBM.pdf [Accessed: 12-Jun-2024].

[3] Republik Indonesia, "Peraturan Presiden No. 191 Tahun 2014," [Online]. Available: https://peraturan.bpk.go.id/Details/41710/perpres-no-191-tahun-2014 [Accessed: 12-Jun-2024].

[4] D. R. Munir, "Informatika STEI ITB," 2024. [Online]. Available: https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi-dan-Koding/2023-2024/08-AES-2024 [Accessed 12-Jun-2024].

[5] D. R. Munir, "Informatika STEI ITB," 2024. [Online]. Available: https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi-dan-Koding/2023-2024/16-Fungsi-hash-SHA-3-2024.pdf [Accessed 12-Jun-2024].

LAMPIRAN

Link Gihub: SirRay03/makalah-kripto (github.com)

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, 12 Juni 2024

Rayhan Nugraha Putra
18221149