BAB III
METODE PENELITIAN

Pada penelitian ini dilakukan studi literatur untuk mengidentifikasi masalah, pengembangan model kriptosistem, dan pengujian model ke dalam program *python*. Berikut adalah langkah-langkah yang dilakukan dalam penelitian ini.

3.1. Identifikasi Masalah

Program yang akan dikontruksi memiliki dua fungsi utama, yaitu enkripsi dan dekripsi. *Input* yang dimasukkan pada menu enkripsi berupa *file Comma-separated values* (CSV) dan kunci yang dipilih oleh *user* sebagai kunci untuk proses enkripsi. *Output* dari menu enkripsi berupa *file* AES yang diperoleh dengan mengenkripsi *file* csv menggunakan SHA-256 dan AES-256 yang telah dimodifikasi. Sebaliknya, untuk menu dekripsi *input* berupa *file* AES dan kunci pilihan *user* untuk proses dekripsi. *Output* dari menu dekripsi berupa *file* CSV yang merupakan *plaintext* asli yang dienkripsi oleh pengirim.

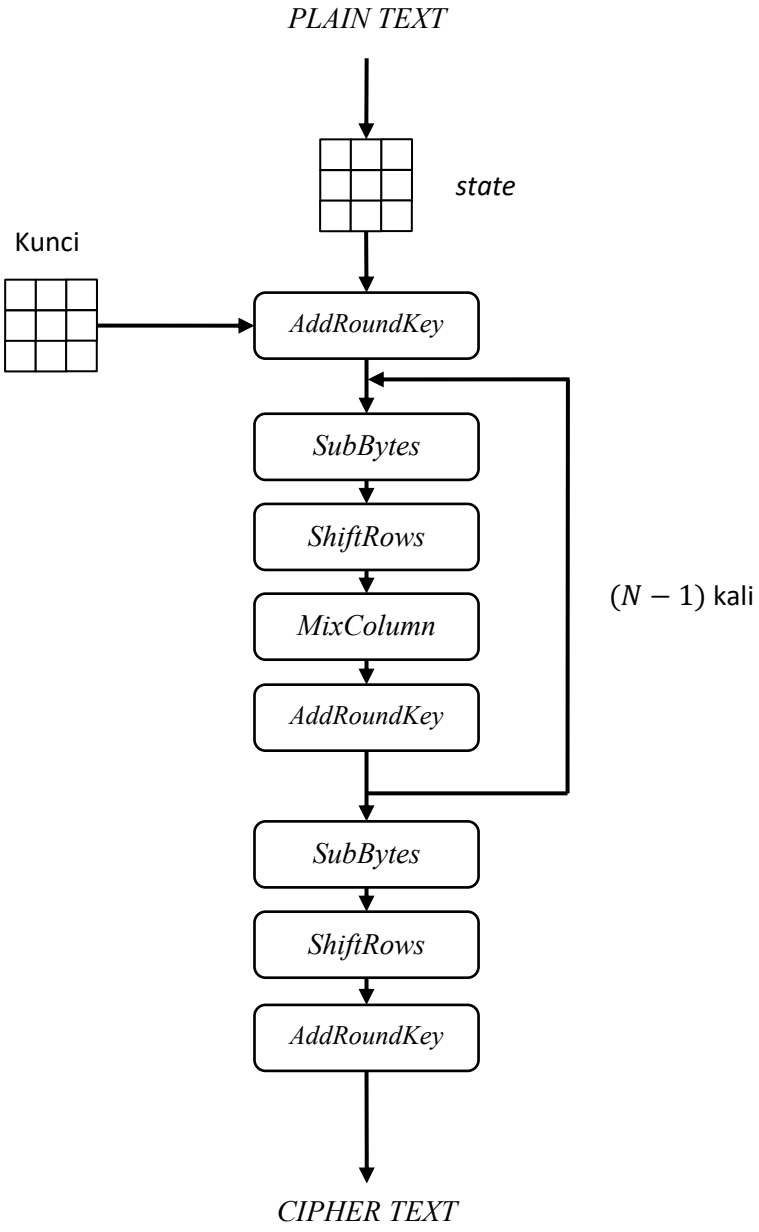
AES-256 merupakan salah satu algoritma kriptografi yang memiliki kemampuan yang baik untuk mengamankan pesan. Namun dengan ditambahkannya SHA-256 akan meningkatkan keamanan *file* yang dikirim. SHA-256 biasa digunakan sebagai tanda tangan digital yang digunakan untuk menjamin keaslian pesan. Pada penelitian kali ini SHA-256 digunakan untuk mengenkripsi kunci yang di*input* oleh *user* yang kemudian akan digunakan dalam proses AES-256.

3.2. Model Dasar

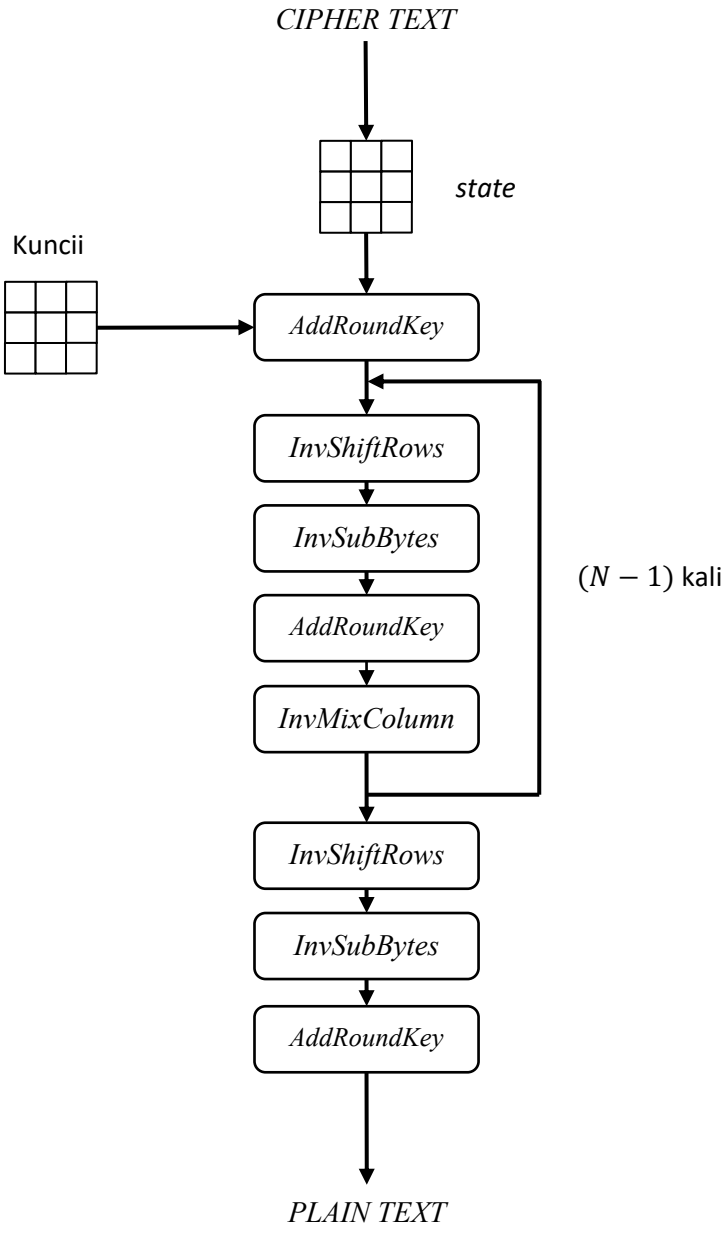
3.2.1. AES-256

Saat mengenkripsi suatu teks dengan menggunakan AES-256, *cipher key* yang digunakan biasanya merupakan bentuk heksadesimal dari *plain key* yang di*input* oleh *user* dengan memiliki panjang 256-bit. Apabila kunci yang diinput kurang dari 256-bit, akan dilakukan *padding*. Enkripsi AES-256 menggunakan

transposisi *SubBytes*, *MixColumns*, dan *AddRoundKey* dengan perputaran sebanyak 14 kali. Pada proses dekripsi digunakan *invers* dari transposisi-transposisi tersebut.



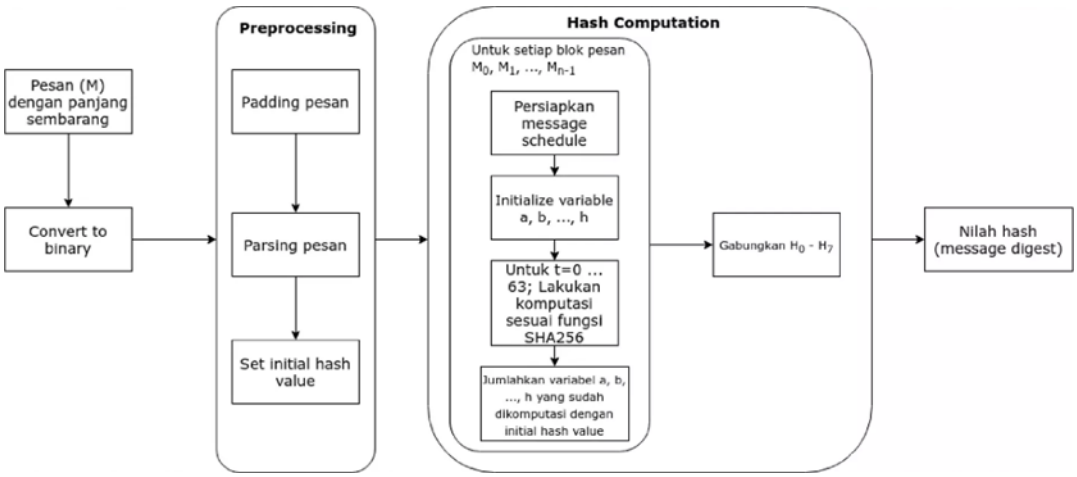
Gambar 3.1 Skema Enkripsi *Advanced Encryption Standard*



Gambar 3.2 Skema Dekripsi *Advanced Encryption Standard*

3.2.2. SHA-256

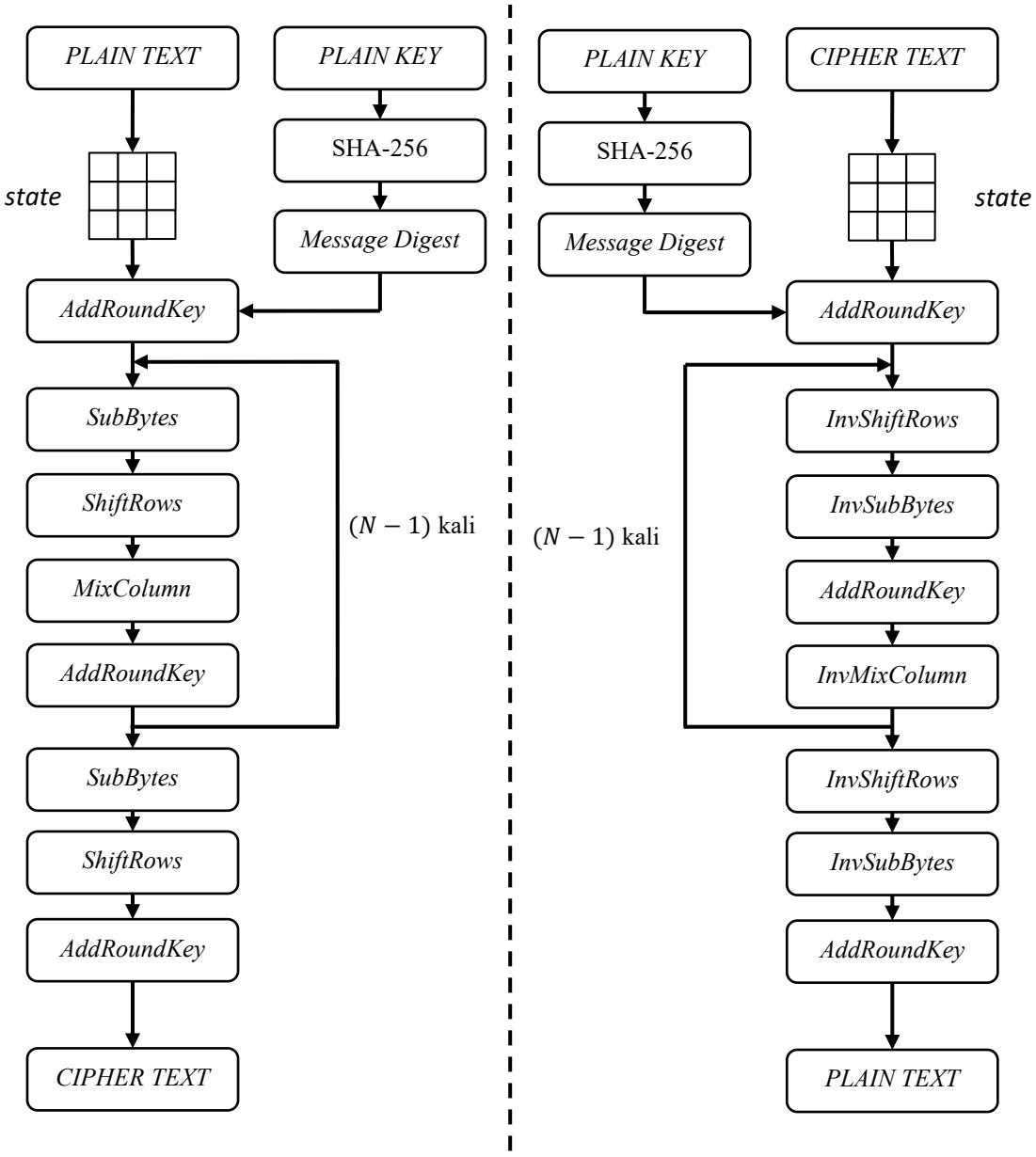
Algoritma SHA-256 digunakan sebagai autentikasi ketika *user* melakukan proses dekripsi, di mana *server* akan memastikan bahwa nilai *hash* yang dihasilkan dari *password* yang *user input* sesuai dengan nilai *hash* yang dihasilkan dari *password* yang *user input* ketika mengenkripsi.



Gambar 3.3 Skema Enkripsi *Secure Hash Algorithm-256*

3.3. Pengembangan Model

Umumnya *cipher key* yang digunakan dalam algoritma AES-256 adalah bentuk heksadesimal dari *plain key* yang di*input* oleh *user*. Namun pada penelitian ini *cipher key* yang akan digunakan adalah *message digest* dari *plain key* yang di*input* oleh *user*. Berikut skema hasil pengembangan model algoritma:



Gambar 3.4 Skema Hasil Pengembangan Model

3.4. Konstruksi Program

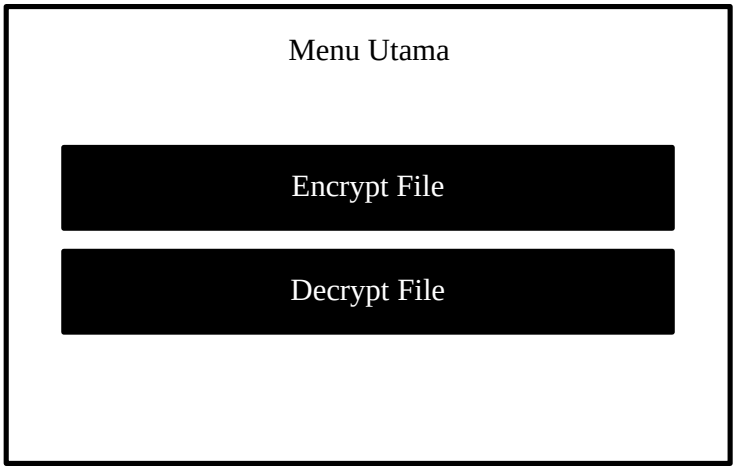
3.4.1. Input dan Output

Pada program pengamanan *big data* ini, diperlukan *input* yang terdiri dari *file* dengan format *.csv dan kunci yang akan digunakan untuk mengenkripsi *file* tersebut. *Plaintext* serta kunci yang diinputkan pada program berupa karakter yang nantinya akan diubah menjadi heksadesimal sesuai dengan tabel ASCII. Kunci yang di*input* akan terlebih dahulu melalui proses *hashing* yang kemudian akan

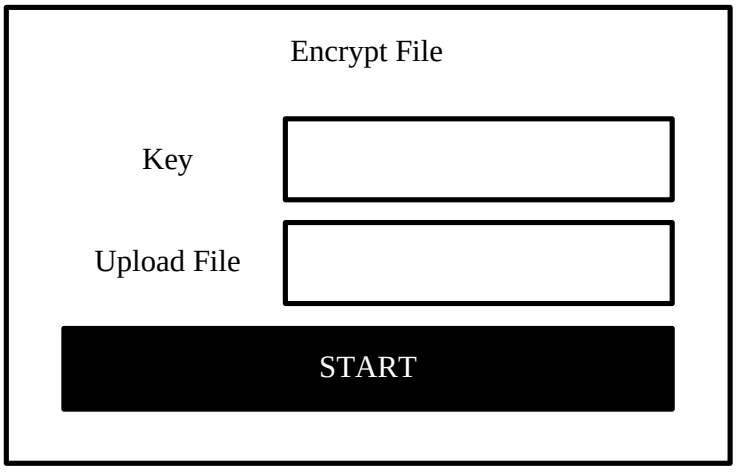
menghasilkan sebuah *message digest*. *Message digest* tersebut akan digunakan sebagai kunci pada proses enkripsi *file* menggunakan AES-256. *Output* yang dihasilkan oleh program akan berbentuk karakter yang nantinya dapat disimpan dalam bentul *file* dengan format *.aes.

3.4.2. Rancangan Tampilan

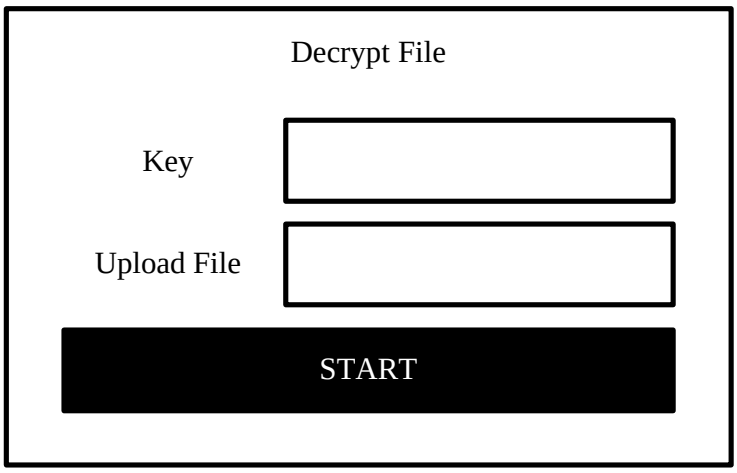
Program yang dibuat akan memiliki tampilan sebagai berikut



Gambar 3.5 Rancangan Tampilan Menu Utama



Gambar 3.6 Rancangan Tampilan Menu Enkripsi



Gambar 3.7 Rancangan Tampilan Menu Dekripsi

3.5. Algoritma Deskriptif

Algoritma untuk melakukan proses enkripsi dan dekripsi *big data* diuraikan sebagai berikut:

a. Enkripsi

Proses enkripsi pesan menggunakan algoritma AES-256 dan kunci *hashing* SHA-256 adalah sebagai berikut:

1. Pengguna menentukan kunci yang akan digunakan dan dimasukkan ke dalam program pada tempat yang telah disediakan.
2. Pengguna memasukkan *file* dengan format *.csv yang akan dienkripsi.
3. Pengguna menekan tombol pada program untuk memulai proses enkripsi.
4. Setelah *file* berhasil dienkripsi, pengguna dapat menyimpan *file ciphertext* dengan format *.aes.

b. Dekripsi

Proses enkripsi pesan menggunakan algoritma AES-256 dan kunci *hashing* SHA-256 adalah sebagai berikut:

1. Pengguna memasukkan kunci yang sebelumnya telah digunakan untuk mengenkripsi *file*.
2. Pengguna memasukkan *file* dengan format *.aes yang akan didekripsi.
3. Pengguna menekan tombol pada program untuk memulai proses dekripsi.

4. Setelah *file* berhasil didekripsi, pengguna dapat menyimpan *file plaintext* dengan format *.aes.

3.6. Coding *Python*

Terdapat beberapa *library python* yang dapat digunakan pada proses pemrograman, yaitu:

1. *Tkinter*

   Tkinter merupakan sebuah *library* yang dapat digunakan untuk mengkontruksi antarmuka system operasi yang berbasis grafis. *Library* ini mempermudah dalam pembuatan aplikasi dengan berbagai komponen seperti tombol, *scroll bar*, dan lainnya.
2. *Hashlib*

   Hashlib merupakan modul bawaan di *python* yang memiliki fungsi untuk melakukan *hashing* dengan berbagai algoritma seperti SHA-1, SHA-256, SHA-512, dan MD5.
3. *Cryptography*

   Cyrptography merupakan *library python* yang menyediakan berbagai fungsi kriptografi. *Library* ini menyediakan berbagai fitur yang mendukung segala jenis algoritma kriptografi. Salah satu bagian dari *library cryptography* adalah *Fernet*. *Fernet* dirancang khusus untuk algoritma kriptografi simetris yang menjamin komunikasi aman dengan menggunakan kunci yang sama untuk enkripsi dan dekripsi karena *Fernet* memastikan bahwa *ciphertext* tidak dapat diubah atau dibaca tanpa kunci yang tepat.

3.7. Proses Validasi

Pada tahap ini akan dilakukan uji coba program dengan cara membandingkan hasil *plaintext* yang diperoleh melalui proses dekripsi dengan *plaintext* asli. Hasil yang diperoleh dari program dikatakan valid apabila *plaintext* yang dihasilkan oleh proses dekripsi sama dengan *plaintext* asli.

3.8. Pengambilan Kesimpulan

Tahap terakhir yang akan dilakukan adalah pengambilan kesimpulan berdasarkan hasil yang diperoleh selama penelitian, serta rekomendasi untuk penelitian selanjutnya agar memperoleh hasil yang lebih maksimal.