

# **PERANCANGAN APLIKASI ENKRIPSI DAN DEKRIPSI FILE MENGGUNAKAN ALGORITMA KRIPTOGRAFI AES-256 BERBASIS STREAMLIT**



**LATAR BELAKANG**

**METODE PENELITIAN**

**HASIL DAN PEMBAHASAN**

**KESIMPULAN**

## LATAR BELAKANG

- ▶ Pengguna kini menyimpan banyak dokumen penting dalam bentuk digital di perangkat pribadi, sehingga data rentan dicuri, diubah, atau diakses pihak tidak berwenang.
- ▶ Diperlukan mekanisme pengamanan file yang kuat namun tetap mudah digunakan oleh pengguna umum.
- ▶ AES-256 adalah algoritma kriptografi simetris standar NIST yang banyak dipakai dan memberikan tingkat keamanan tinggi dengan performa yang masih memadai untuk enkripsi file.
- ▶ Namun, sebagian implementasi enkripsi file masih berbasis baris perintah dan kurang ramah bagi pengguna non-teknis.
- ▶ Streamlit menyediakan cara sederhana untuk membangun aplikasi web interaktif yang mendukung unggah dan unduh file melalui browser tanpa instalasi kompleks.
- ▶ Penelitian ini bertujuan merancang aplikasi web sederhana untuk enkripsi dan dekripsi file menggunakan AES-256 berbasis password dengan antarmuka Streamlit dan menguji keberhasilan proses serta waktu eksekusinya pada beberapa ukuran file.



LATAR BELAKANG

METODE PENELITIAN

HASIL DAN PEMBAHASAN

KESIMPULAN

## METODE PENELITIAN

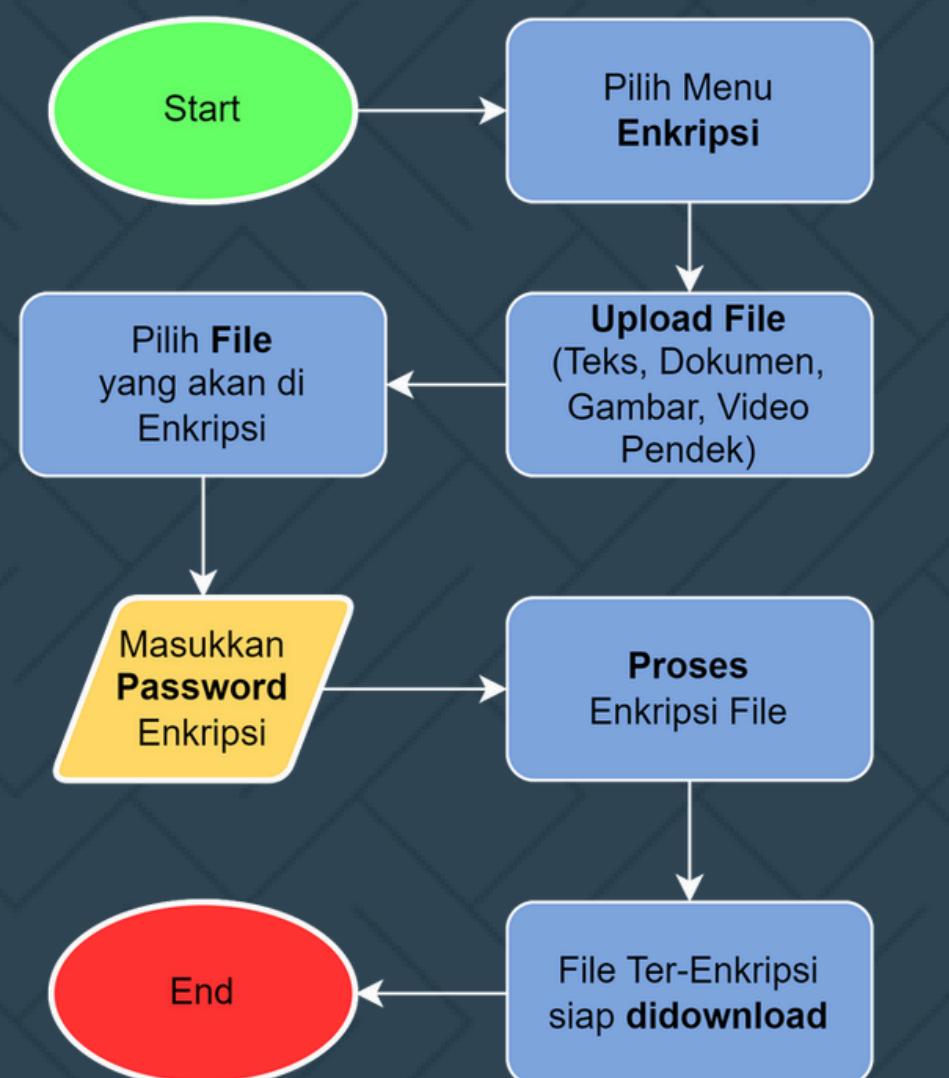
Penelitian ini menggunakan pendekatan rekayasa perangkat lunak dengan membangun prototipe aplikasi web untuk enkripsi dan dekripsi file. Algoritma yang digunakan adalah AES-256 dalam mode GCM dengan kunci yang diturunkan dari password melalui PBKDF2-HMAC-SHA256 yang diberi salt dan iterasi tinggi.

Aplikasi dikembangkan menggunakan Python dan Streamlit sehingga fungsi enkripsi dan dekripsi dapat diakses melalui browser lokal tanpa interaksi langsung dengan command line. Sistem menyediakan dua mode, Encrypt dan Decrypt, yang masing-masing menangani proses unggah file, pembangkitan salt dan nonce, enkripsi/dekripsi AES-GCM, serta pengunduhan file hasil dengan ekstensi .enc maupun plaintext.

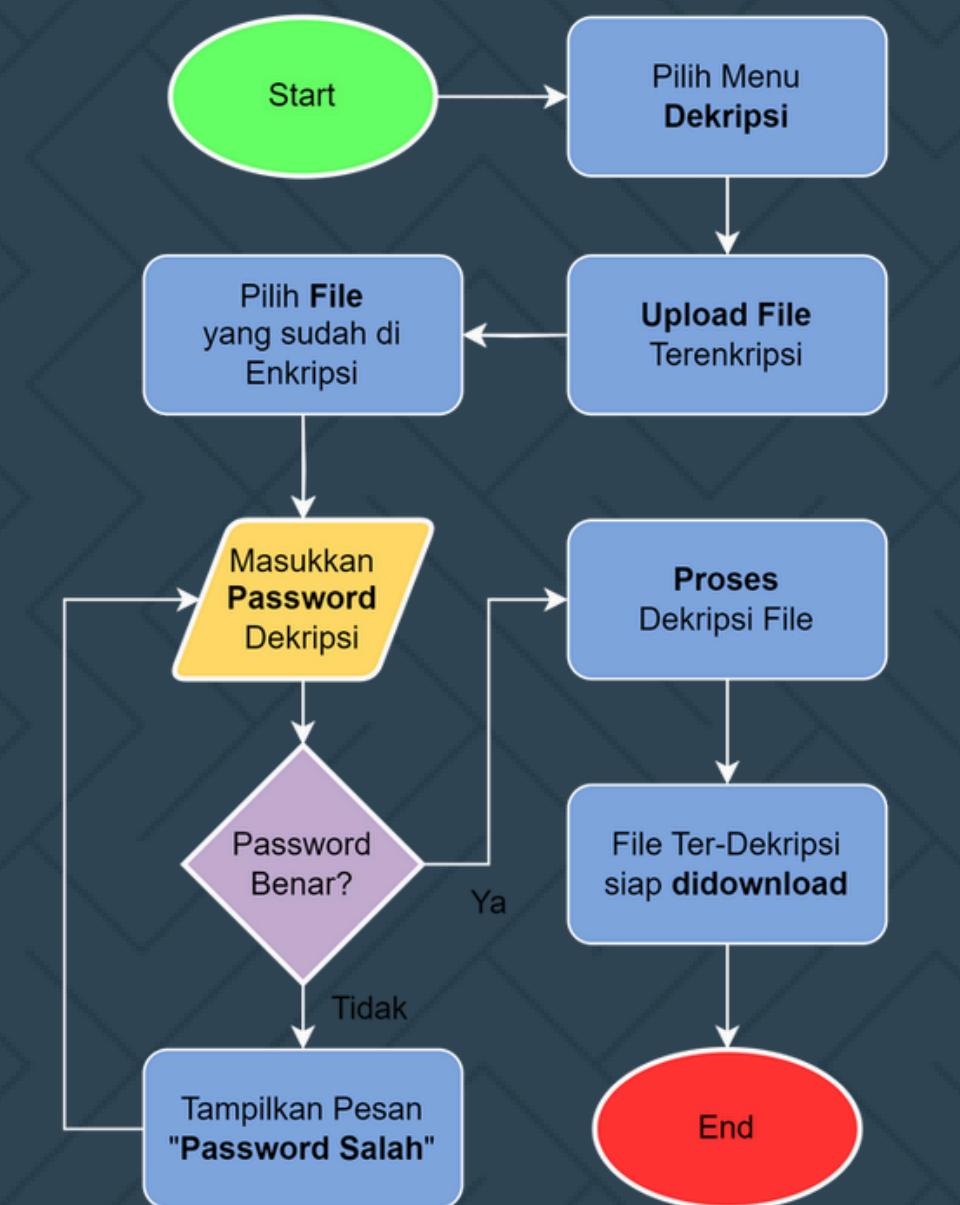
Pengujian dilakukan secara eksperimental pada satu komputer pribadi menggunakan beberapa file uji berukuran kecil, sedang, dan besar. Untuk setiap file dicatat ukuran, waktu enkripsi, waktu dekripsi, serta keberhasilan pemulihan file, dan dilakukan uji negatif dengan password salah guna memastikan data tidak dapat dikembalikan tanpa kunci yang benar.

# DIAGRAM / CHART

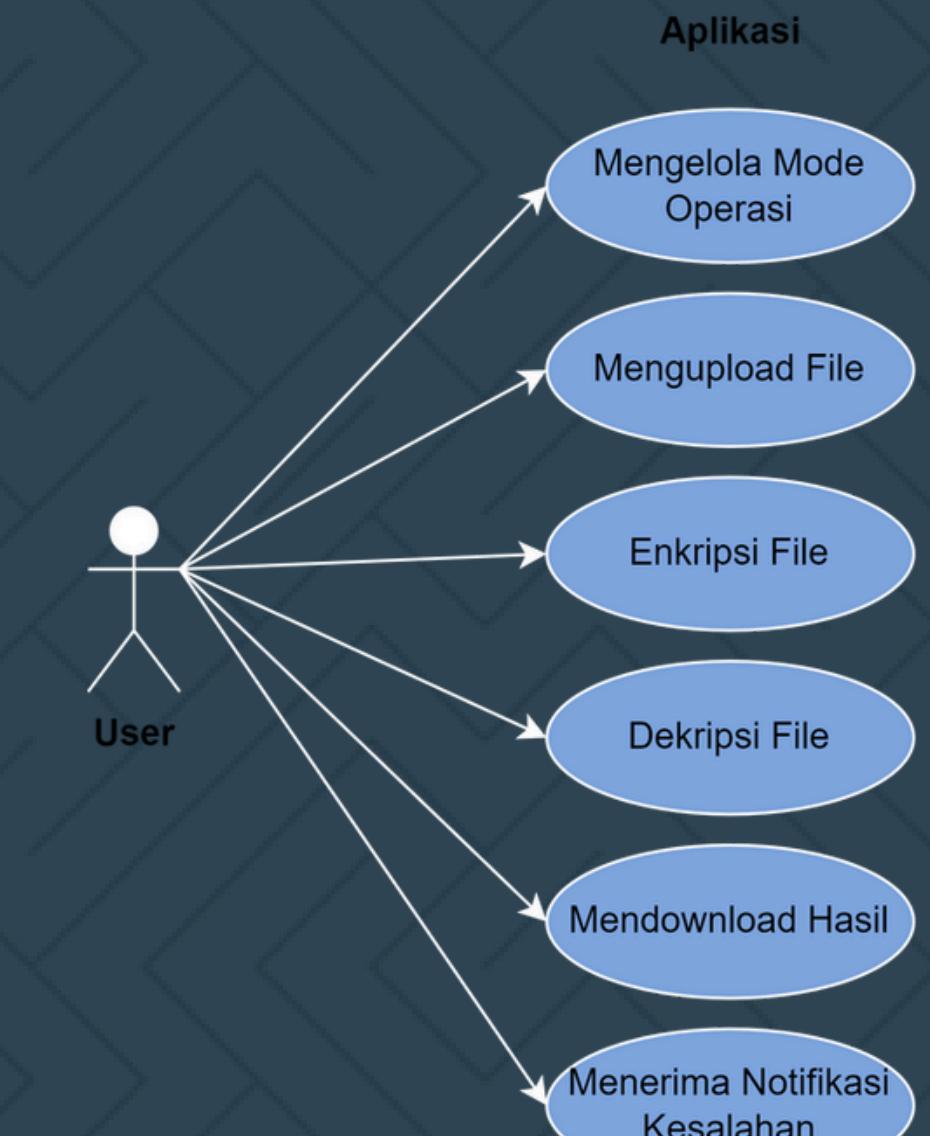
## FLOWCHART ENKRIPSI



## FLOWCHART DEKRIPSI



## USECASE DIAGRAM



# Aplikasi Enkripsi & Dekripsi File AES-256

Aplikasi sederhana untuk mengenkripsi dan mendekripsi file menggunakan algoritma AES-256 (mode GCM).

Pilih mode operasi:

- Encrypt
- Decrypt

Pilih file



Drag and drop file here  
Limit 200MB per file

Browse files

Masukkan password



Enkripsi



LATAR BELAKANG

METODE PENELITIAN

HASIL DAN PEMBAHASAN

KESIMPULAN

## HASIL DAN PEMBAHASAN

ID file	Jenis File	Ukuran (MB)	Waktu enkripsi (detik)	Waktu dekripsi (detik)
F1	Teks (.txt)	0,1	0,3	0,2
F2	Dokumen PDF	1,25	0,8	0,7
F3	Gambar (.jpg)	5,4	1,9	1,7
F4	Video pendek (.mp4)	18,7	5,4	5

Hasil pengujian menunjukkan bahwa seluruh file uji (teks, PDF, gambar, dan video pendek) dapat dienkripsi dan didekripsi kembali dengan benar tanpa kerusakan isi. File terenkripsi tidak dapat dibaca secara langsung oleh aplikasi umum sehingga kerahasiaan konten tetap terjaga.

## HASIL DAN PEMBAHASAN

ID file	Jenis File	Ukuran (MB)	Waktu enkripsi (detik)	Waktu dekripsi (detik)
F1	Teks (.txt)	0,1	0,3	0,2
F2	Dokumen PDF	1,25	0,8	0,7
F3	Gambar (.jpg)	5,4	1,9	1,7
F4	Video pendek (.mp4)	18,7	5,4	5

Waktu proses meningkat seiring pertambahan ukuran file, dari sekitar 0,3 detik untuk file teks 0,10 MB hingga sekitar 5 detik untuk video 18,70 MB, baik pada tahap enkripsi maupun dekripsi. Rentang waktu ini masih dapat diterima untuk penggunaan harian pada file kecil hingga menengah, sehingga prototipe dinilai cukup responsif bagi pengguna individu.

## HASIL DAN PEMBAHASAN

ID file	Jenis File	Ukuran (MB)	Waktu enkripsi (detik)	Waktu dekripsi (detik)
F1	Teks (.txt)	0,1	0,3	0,2
F2	Dokumen PDF	1,25	0,8	0,7
F3	Gambar (.jpg)	5,4	1,9	1,7
F4	Video pendek (.mp4)	18,7	5,4	5

Pada pengujian negatif, percobaan dekripsi dengan password yang salah menyebabkan kegagalan autentikasi AES-GCM dan aplikasi tidak pernah menghasilkan plaintext. Hal ini menunjukkan bahwa kombinasi PBKDF2 dan AES-GCM telah diimplementasikan dengan benar dan mampu mencegah pemulihan data jika password tidak sesuai.



**LATAR BELAKANG**

**METODE PENELITIAN**

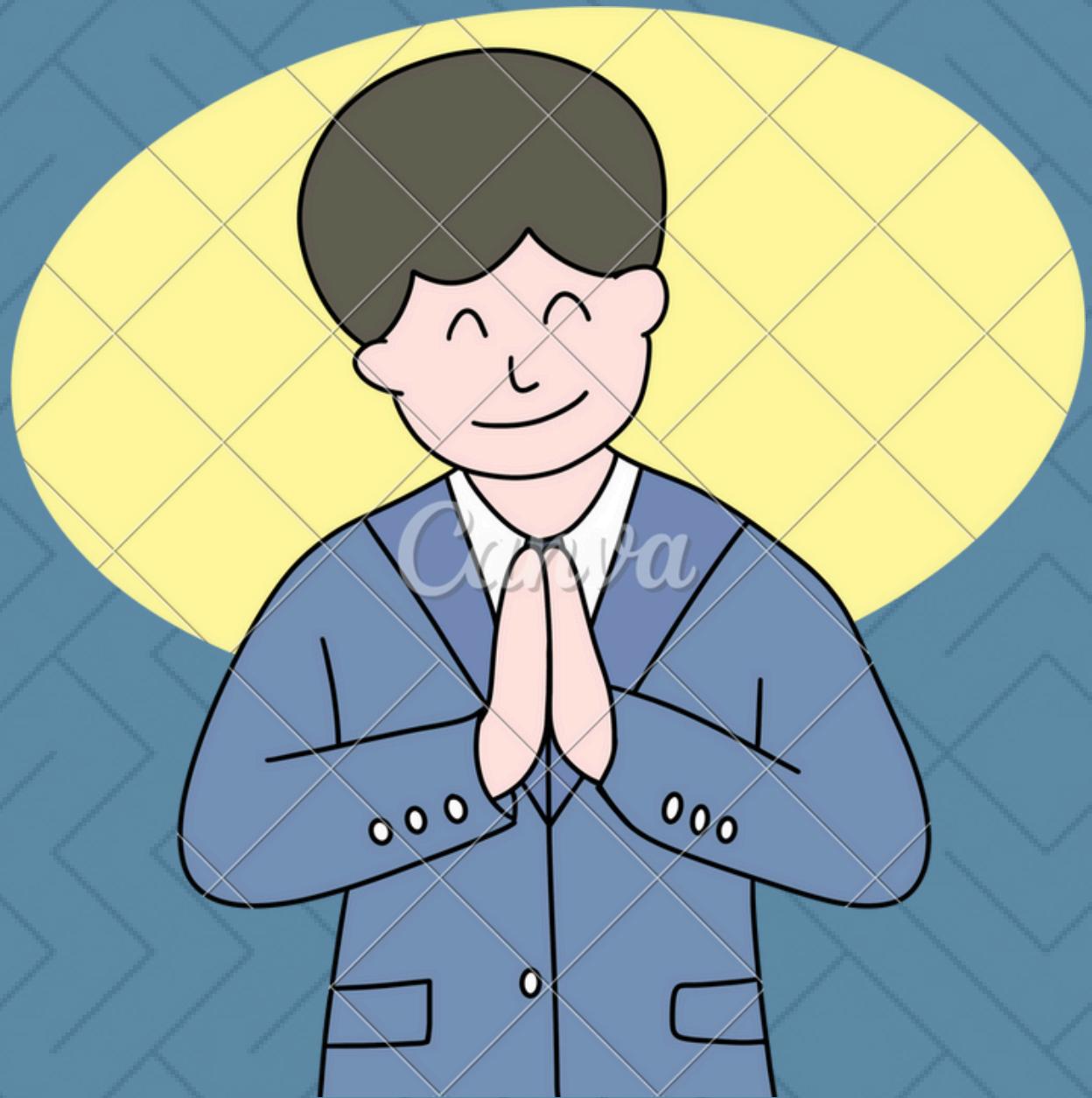
**HASIL DAN PEMBAHASAN**

**KESIMPULAN**

## KESIMPULAN

Penelitian ini berhasil mengembangkan prototipe aplikasi enkripsi dan dekripsi file berbasis web menggunakan AES-256 dan Streamlit dengan skema derivasi kunci PBKDF2-HMAC-SHA256. Aplikasi memungkinkan pengguna umum untuk mengamankan file lokal melalui antarmuka web sederhana hanya dengan mengunggah file dan memasukkan password.

Hasil pengujian memperlihatkan bahwa proses enkripsi-dekripsi berjalan dengan benar pada berbagai ukuran file, dengan overhead waktu yang masih wajar untuk penggunaan individu, serta tidak menghasilkan plaintext ketika password salah. Oleh karena itu, prototipe ini layak dipertimbangkan sebagai solusi sederhana untuk meningkatkan keamanan penyimpanan file lokal dan dapat dikembangkan lebih lanjut dengan manajemen password dan integrasi layanan awan.



**TERIMA KASIH**