BUSINESS REQUIREMENTS DOCUMENT

1. Latar belakang dan masalah bisnis

Penyimpanan dokumen penting (tugas akademik, laporan kerja, data pribadi) di PC/laptop pribadi tanpa enkripsi menimbulkan risiko pencurian, modifikasi, dan kebocoran data bila perangkat hilang, dicuri, atau terkena malware. Sebagian solusi enkripsi file yang ada masih berbasis command line dan kurang ramah bagi pengguna non-teknis, sehingga adopsinya rendah meskipun risiko keamanan tinggi.

SLR menunjukkan bahwa AES adalah standar emas kriptografi simetris dengan tiga varian utama (AES-128, AES-192, AES-256) yang memiliki trade-off kinerja vs keamanan; AES-128 unggul kecepatan, sedangkan AES-256 unggul keamanan jangka panjang dan ketahanan terhadap ancaman kuantum. Dengan mempertimbangkan kebutuhan pengguna umum dan rekomendasi literatur, dipilih desain solusi enkripsi file berbasis AES-256 yang tetap praktis untuk penggunaan sehari-hari.

2. Tujuan bisnis dan indikator keberhasilan

Tujuan bisnis utama:

- Menyediakan platform enkripsi–dekripsi file lokal yang mudah digunakan pengguna umum tanpa perlu pengetahuan command line.
- Meningkatkan tingkat keamanan penyimpanan file lokal untuk dokumen sensitif dengan memanfaatkan kekuatan keamanan AES-256.
- Menunjukkan bahwa overhead waktu proses AES-256 masih dapat diterima untuk skenario penggunaan harian (file kecil hingga menengah) pada PC personal.

Key Performance Indicator (KPI):

- $\geq$ 99% file uji (teks, PDF, gambar, video) berhasil dienkripsi dan didekripsi dengan benar menggunakan password yang tepat.
- 0% keberhasilan dekripsi ketika password salah atau data rusak (tidak ada plaintext yang dihasilkan).
- Waktu enkripsi–dekripsi untuk file sampai ±20 MB berada dalam rentang beberapa detik (misalnya < 6 detik untuk video pendek ±18–20 MB).
- Tingkat kepuasan pengguna uji internal terhadap kemudahan penggunaan antarmuka minimal “baik” (misalnya $\geq$ 4/5 dalam kuisioner usability, jika diuji).

3. Ruang lingkup dan batasan

Dalam ruang lingkup

- Aplikasi web lokal berbasis Streamlit yang berjalan pada satu PC (local host) di mana PC yang sama berperan sebagai server dan client.
- Fitur enkripsi dan dekripsi file dengan algoritma AES-256 di mode Galois/Counter Mode (GCM) menggunakan kunci yang diturunkan dari password melalui PBKDF2-HMAC-SHA256.
- Dukungan tipe file umum:
    - Teks kecil (misalnya 0,1 MB)
    - Dokumen PDF sekitar 1 MB
    - File gambar beberapa MB
    - File video pendek hingga puluhan MB (±18–20 MB)
- Antarmuka web dengan komponen: pemilihan mode Encrypt/Decrypt, upload file, input password, tombol proses, dan tombol download hasil.

Di luar ruang lingkup (versi awal)

- Manajemen akun dan autentikasi multi-user (login, role, dsb.).
- Penyimpanan file di server terpisah atau integrasi langsung dengan cloud storage (hanya diusulkan untuk pengembangan ke depan).
- Pengelolaan password tingkat lanjut seperti password manager, recovery password, atau reset password terpusat.
- Benchmark komparatif AES-128 vs AES-256 di berbagai arsitektur hardware (hanya dibahas di SLR sebagai research gap, belum diimplementasikan di aplikasi).

4. Profil pengguna dan pemangku kepentingan

Pengguna utama

- Mahasiswa dan staf akademik yang menyimpan tugas, laporan, dan data pribadi di laptop/PC pribadi.
- Pekerja kantoran yang perlu mengamankan dokumen kerja (laporan, kontrak, dokumen internal) di perangkat pribadi tanpa infrastruktur enterprise yang kompleks.

Kebutuhan utama pengguna:

- Aplikasi mudah dijalankan (cukup buka browser lokal dan akses antarmuka).

- Proses enkripsi–dekripsi jelas langkahnya dan memberikan feedback sukses/gagal yang eksplisit.
- Tidak perlu memahami detail teknis AES, PBKDF2, atau mode GCM.

Pemangku kepentingan lain

- Pengembang/peneliti yang ingin mengimplementasikan rekomendasi SLR terkait penggunaan AES-256 untuk keamanan jangka panjang.
- Dosen atau institusi yang menggunakan aplikasi ini sebagai artefak penelitian atau media pembelajaran kriptografi terapan.

5. Kebutuhan fungsional (Functional Requirements)

5.1. Manajemen mode proses

- Sistem harus menyediakan pilihan mode Encrypt dan Decrypt pada halaman utama yang dapat dipilih pengguna (misalnya radio button).
- Sistem harus mengubah komponen antarmuka (teks, label tombol) sesuai dengan mode yang dipilih pengguna.

5.2. Enkripsi file (Encrypt mode)

- Pengguna dapat mengunggah satu file yang akan dienkripsi melalui komponen file uploader pada antarmuka.
- Pengguna wajib mengisi field password sebagai dasar pembangkitan kunci; proses tidak boleh berjalan jika password kosong.
- Sistem harus:
    - Menghasilkan salt acak.
    - Menerapkan PBKDF2-HMAC-SHA256 dengan salt dan iterasi cukup tinggi untuk menurunkan kunci AES-256 dari password.
    - Menghasilkan nonce untuk AES-GCM.
    - Mengenkripsi isi file menggunakan AES-256 GCM dan menghasilkan ciphertext beserta tag autentikasi.
- Sistem harus menggabungkan saltnonceciphertext (dan tag, jika diatur eksplisit) menjadi satu struktur data yang kemudian dikemas sebagai file keluaran ber-ekstensi .enc.
- Sistem harus menampilkan pesan bahwa enkripsi berhasil dan menyediakan tombol Download untuk mengunduh file .enc.

5.3. Dekripsi file (Decrypt mode)

- Pengguna dapat mengunggah file .enc melalui file uploader ketika memilih mode Decrypt.

- Pengguna wajib memasukkan password yang diyakini sama dengan password pada proses enkripsi.
- Sistem harus:
    - Mengekstrak salt dan nonce dari struktur data file .enc.
    - Menurunkan kembali kunci AES-256 dengan PBKDF2-HMAC-SHA256 menggunakan password dan salt yang sama.
    - Menjalankan proses dekripsi AES-GCM.
- Jika autentikasi GCM gagal (password salah atau data rusak), sistem hanya menampilkan pesan error dan tidak menghasilkan/mengunduh file plaintext.
- Jika dekripsi berhasil, sistem menghasilkan kembali file asli dan menyediakan tombol Download agar pengguna dapat mengunduh file plaintext.

5.4. Umpan balik proses dan pesan kesalahan

- Sistem harus menampilkan indikator atau pesan proses (misalnya "sedang mengenkripsi…" atau "sedang mendekripsi…") selama operasi berlangsung.
- Sistem harus memberikan pesan sukses yang jelas setelah enkripsi–dekripsi berhasil, beserta informasi bahwa file siap untuk diunduh.
- Sistem harus memberikan pesan error yang jelas jika:
    - Password salah.
    - File .enc tidak valid (struktur tidak sesuai).
    - Terjadi kegagalan internal enkripsi–dekripsi lain (misalnya exception kriptografi).

6. Kebutuhan non-fungsional (Non-Functional Requirements)

6.1. Keamanan

- Algoritma enkripsi harus menggunakan AES-256 untuk memberikan margin keamanan tinggi dan ketahanan yang lebih baik terhadap ancaman komputasi kuantum dibanding AES-128.
- Skema pembangkitan kunci harus menggunakan PBKDF2-HMAC-SHA256 dengan salt acak dan jumlah iterasi cukup tinggi untuk mengurangi risiko brute-force terhadap password lemah.
- Mode operasi harus AES-GCM, yang sekaligus menjamin kerahasiaan dan autentikasi integritas data dalam satu proses.
- Sistem tidak boleh menyimpan password pengguna dalam bentuk plaintext, dan sebaiknya tidak menyimpan password sama sekali di sisi server lokal setelah proses selesai.

6.2. Performa

- Sistem harus mampu memproses file teks kecil (±0,1 MB) dengan waktu enkripsi sekitar 0,3 detik dan dekripsi sekitar 0,2 detik, atau dalam kisaran serupa.
- Sistem harus mampu memproses file PDF ±1,25 MB dengan waktu enkripsi sekitar 0,8 detik dan dekripsi sekitar 0,7 detik.
- Sistem harus mampu memproses file gambar ±5,4 MB dengan waktu enkripsi sekitar 1,9 detik dan dekripsi sekitar 1,7 detik.
- Sistem harus mampu memproses file video pendek ±18,7 MB dengan waktu enkripsi sekitar 5,4 detik dan dekripsi sekitar 5,0 detik, atau kisaran yang dianggap masih nyaman untuk penggunaan individu.

6.3. Usability

- Antarmuka berbasis Streamlit harus menampilkan pilihan mode, upload file, input password, tombol proses, dan tombol download dengan layout yang sederhana dan mudah dipahami.
- Pengguna harus dapat menyelesaikan tugas enkripsi atau dekripsi dalam beberapa langkah linear (pilih mode → upload file → isi password → proses → download).[

6.4. Lingkungan operasi dan teknologi

- Aplikasi dikembangkan menggunakan bahasa pemrograman Python dan framework Streamlit sebagai antarmuka web.
- Prototipe dijalankan pada satu PC yang bertindak sebagai server dan client, dengan file tersimpan di sistem file lokal pengguna.

7. Asumsi, ketergantungan, dan risiko

Asumsi

- Pengguna memiliki PC/laptop dengan kemampuan komputasi yang memadai untuk menjalankan Python, Streamlit, dan library kriptografi tanpa lag berlebihan.
- Pengguna bersedia mengingat dan menjaga kerahasiaan password karena tidak ada mekanisme recovery password terpusat.

Ketergantungan

- Ketersediaan library kriptografi Python yang mendukung AES-GCM dan PBKDF2-HMAC-SHA256 sesuai rekomendasi NIST.
- Ketersediaan lingkungan Python dan Streamlit yang stabil pada perangkat pengguna.

Risiko utama

- Password lemah masih dapat menjadi titik lemah meskipun AES-256 dan PBKDF2 digunakan, terutama bila diserang secara offline dengan resource besar.
- User error, seperti kehilangan password, menyebabkan file .enc tidak dapat didekripsi kembali karena tidak ada backdoor atau recovery.

8. Roadmap pengembangan lanjutan (optional untuk bab “Saran” jurnal)

Mengacu pada SLR yang menyoroti trade-off performa AES-128/192/256 pada platform modern, BRD ini membuka peluang pengembangan berikut:

- Menambahkan opsi pemilihan varian AES (128/256) dan melakukan benchmarking performa pada CPU berbeda (x86 vs ARM) untuk mengkaji trade-off yang lebih praktis.
- Integrasi dengan layanan penyimpanan cloud untuk mendukung skenario backup dan berbagi file terenkripsi secara aman.
- Penambahan modul manajemen password yang lebih aman (misalnya integrasi password manager eksternal atau kebijakan kekuatan password).