BAB V
PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pengujian dapat disimpulkan bahwa kombinasi algoritma RSA dan AES telah berhasil diimplementasikan untuk enkripsi dan dekripsi file serta transaksi data di database. Algoritma RSA digunakan untuk mengenkripsi kunci AES, sedangkan AES digunakan dalam proses enkripsi dan dekripsi file. Aplikasi ini mampu menangani berbagai jenis file, seperti PDF, DOCX, PNG, JPG, dan TXT, serta berjalan di lingkungan lokal tanpa memerlukan koneksi internet. Seluruh proses mulai dari unggah file, enkripsi, penyimpanan, hingga dekripsi, dapat dijalankan dengan baik tanpa hambatan yang berarti, meskipun sistem belum dilengkapi dengan autentikasi pengguna dan fitur manajemen akses.

Evaluasi terhadap performa sistem dilakukan terhadap data terenkripsi, terdekripsi, dan tanpa enkripsi. Parameter yang digunakan meliputi waktu pemrosesan, pemakaian CPU, penggunaan memori (RAM), dan throughput. Penilaian efektivitas melalui metode AHP dan SAW menunjukkan bahwa proses enkripsi memperoleh nilai 53,19% yang masuk dalam kategori "Cukup Baik", sementara proses dekripsi dan proses tanpa enkripsi masing-masing memperoleh nilai 24,07% dan 32,20%, yang tergolong "Tidak Baik". Rata-rata nilai efektivitas keseluruhan adalah 36,49%, yang mengindikasikan bahwa sistem masih belum optimal atau "Tidak Baik" untuk digunakan dalam skala produksi secara luas, khususnya dalam pengelolaan file berukuran besar yang memerlukan sumber daya komputasi tinggi.

5.2 Saran

Sebagai tindak lanjut dari hasil yang telah diperoleh selama penelitian ini, beberapa rekomendasi yang dapat dipertimbangkan untuk pengembangan di masa mendatang sebagai berikut:

1. Sistem diadaptasi ke dalam lingkungan server atau layanan cloud, sehingga dapat diakses secara daring dan mendukung pengujian dalam skenario multi-user.
2. Penambahan fitur autentikasi serta manajemen pengguna sangat diperlukan untuk meningkatkan keamanan sistem serta mengatur hak akses terhadap file yang dienkripsi.
3. Optimalisasi pemanfaatan sumber daya perangkat, khususnya dalam menangani file berukuran besar, dapat dilakukan melalui penerapan teknik seperti *data compression* atau *chunking* untuk membagi file menjadi beberapa bagian kecil sebelum dienkripsi.
4. Penggabungan algoritma tambahan seperti SHA untuk pengecekan integritas data dan digital signature untuk menjamin keaslian file, dapat meningkatkan tingkat keamanan sistem agar lebih sesuai dengan standar kriptografi modern.
5. Uji coba lebih lanjut disarankan dilakukan terhadap berbagai jenis format file lain serta pada sistem operasi dan perangkat yang berbeda untuk mengukur fleksibilitas dan kompatibilitas sistem yang dikembangkan.
6. Rendahnya nilai efektivitas pada proses dekripsi dan data tanpa enkripsi menunjukkan perlunya peningkatan efisiensi sistem. Hal ini dapat dicapai melalui perbaikan algoritma yang digunakan atau peningkatan kapasitas perangkat keras yang mendukung proses enkripsi dan dekripsi.
7. Melakukan uji coba pada kondisi nyata yang membutuhkan efisiensi tinggi, seperti pemrosesan file dengan ukuran sangat besar atau transaksi data secara waktu nyata (real-time), agar sistem benar-benar siap diterapkan dalam konteks penggunaan yang lebih luas dan kompleks.
8. Penelitian ini hanya mencakup file di bawah 50 MB, sehingga pengujian lanjutan pada file berukuran lebih besar perlu dilakukan untuk menilai kemampuan dan kestabilan sistem dalam menangani proses enkripsi dan dekripsi berskala besar.