e-ISSN 2962-6129
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 20 Agustus 2022

IMPLEMENTASI KEAMANAN *FILE* PADA APLIKASI PENYIMPANAN BERBASIS *CLOUD COMPUTING* DENGAN ALGORITMA *ADVANCED ENCRYPTION STANDARD* (AES) DAN KOMPRESI *LEMPEL ZIV WELCH* (LZW)

Dwi Setyo Wiratomo[1] , Bayu Hananto[2] , I Wayan Widi Pradnyana[3]
Program Studi Informatika, Fakultas Ilmu Komputer
Universitas Pembangunan Nasional Veteran Jakarta
Jl. Rs. Fatmawati No. 1, Pondok Labu, Jakarta Selatan, DKI Jakarta, 12450
dwiratomo12@gmail.com[1], bayuhananto.2020@gmail.com[2], wayan.widi@upnvj.ac.id[3]

Abstrak. Data *file* merupakan data yang digunakan tiap perusahaan, organisasi, maupun perseorangan. Apalagi data penting yang tersimpan dalam database. Supaya data itu bisa diakses fleksibel, maka bisa disimpan dalam *cloud* storage. Walaupun sangat mudah diakses, maka harus disiapkan juga untuk keamanan data dari pengguna *cloud*. Setiap layanan *cloud* harus menyediakan keamanan data yang aman untuk penggunanya supaya menghindari penyusup untuk mengambil dan merusak data pengguna *cloud*. Cloud storage yang digunakan dalam peneliatian ini adalah AWS S3. Metode kriptografi yang digunakan dalam pengamanan *file* adalah algoritma *Advanced Encryption* Standard (AES) yang akan menghasilkan enkripsi dan dekripsi data *file* serta melakukan kompresi *file* menggunakan algoritma *Lempel Ziv Welch* (LZW) untuk menurunkan ukuran *file* dan mempercepat pengiriman *file* ke *cloud storage*. Hasil dari penelitian ini adalah pengamanan *file* menggunakan algoritma AES dan algoritma LZW dapat mempengaruhi ukuran *file* yang mengalami kenaikan rata-rata 200% dari ukuran *file* aslinya.

Kata Kunci: kriptografi, *cloud computing*, *cloud storage*, Algoritma *Advanced Encryption* Standard (AES), *Lempel Ziv Welch* (LZW).

1 Pendahuluan

Perkembangan teknologi saat ini semakin pesat membuat proses penggunaan teknologi menjadi lebih mudah di semua bidang. Salah satunya adalah media penyimpanan yang bisa di akses dalam keadaan apapun dan dimanapun. Ada beberapa jenis media penyimpanan yang berkembang salah satunya adalah media penyimpanan berbasis *cloud*. *Cloud computing* merupakan teknologi yang dapat menyimpan sebuah informasi dalam *server* secara *virtual* dan dapat diakses kapan saja.

Cloud storage merupakan bagian dari sistem cloud computing yang menyediakan media penyimpanan yang dapat diakses hanya dengan memerlukan jaringan internet. Walaupun sangat mudah diakses, maka harus disiapkan juga untuk keamanan data dari pengguna cloud. Setiap layanan cloud harus menyediakan keamanan data yang aman untuk penggunanya supaya mengatasi penyusup untuk mengambil dan merusak data pengguna cloud.

Keamanan data merupakan hal yang penting dalam layanan *cloud*. Teknik kriptografi adalah salah satu cara untuk mengamankan data yang tersimpan dalam layanan *cloud*. Salah satu kriptografi yang digunakan adalah algoritma AES (*Advanced Encryption Standard*). Dengan menggunakan AES, keamanan data yang tersimpan dalam *cloud storage* akan menjadi berlapis dan tidak mudah untuk diambil maupun dirusak oleh penyusup. Setelah data dienkripsi, akan dikombinasikan dengan algoritma kompresi *Lempel Ziv* Welch (LZW) supaya penyimpanan dalam *cloud storage* tidak terlalu besar dan efisien dalam pengiriman data ke *cloud storage.*

Penelitian ini bertujuan untuk membuat aplikasi menyimpan *file* yang penting dan dapat diakses dimana saja dan kapan saja hanya menggunakan internet. Aplikasi ini diharapkan dapat membantu siapa saja dalam menyimpan suatu file tanpa takut di bajak/di curi oleh *hacker* dan mempercepat penyimpanan file secara fleksibel.

2 Metodologi Penelitian

2.1 Cloud Computing

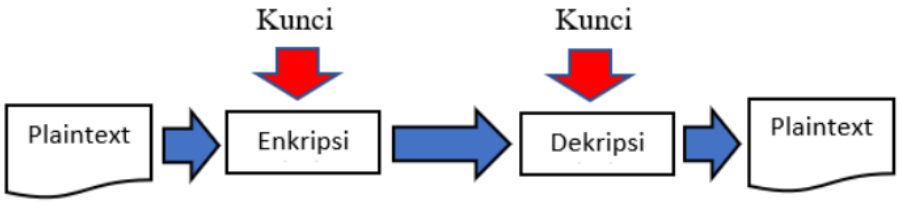
Cloud Computing merupakan penyedia layanan dalam bidang teknologi yang berhubungan dengan infrastruktur komputasi, pembuatan aplikasi, manajemen bisnis hingga kolaborasi sebagai layanan yang dapat digunakan pada saat dibutuhkan kapan saja dan dimana saja.

2.2 Cloud Storage

Cloud storage merupakan penyimpanan media yang dapat diakses kapan saja dan dimana saja serta hanya memerlukan jaringan internet. Pengguna dapat menyimpan dan melindungi data dalam jumlah berapapun di dalam cloud storage ini karena layanan ini menawarkan skalabilitas, ketersediaan data, keamanan, dan performa penyimpanannya.
Keuntungan dari menggunakan cloud storage ini adalah bisa menyimpan data penting dan dalam pengaksesannya mudah, back-up data tanpa takut kehilangan datanya, bisa berkolaborasi dengan mudah sebagai wadah untuk menyimpan *file-file* dengan teman, serta biaya yang digunakan sangat murah dibandingkan dengan media penyimpanan fisik.

2.3 Kriptografi

Kata crypto berasal dari dua kata Yunani, wocrypto dan graphene. Kriptografi adalah rahasia dalam arti, tetapi graphene, yang kami maksud di sini, adalah goresan atau tulisan, jadi cipher itu sendiri secara harfiah adalah tulisan rahasia. Dari segi istilah, enkripsi adalah ilmu yang mempelajari teknik, dan teknik ini dikaitkan dengan berbagai aspek keamanan informasi seperti kerahasiaan, integritas data, dan otentikasi [2].
Kriptografi bertujuan untuk menjaga kerahasiaan informasi yang terkandung dalam data sehingga informasi tersebut tidak dapat diketahui oleh pihak yang tidak bertanggung jawab. Orang yang membuat algoritma kriptografi disebut Kriptografer.

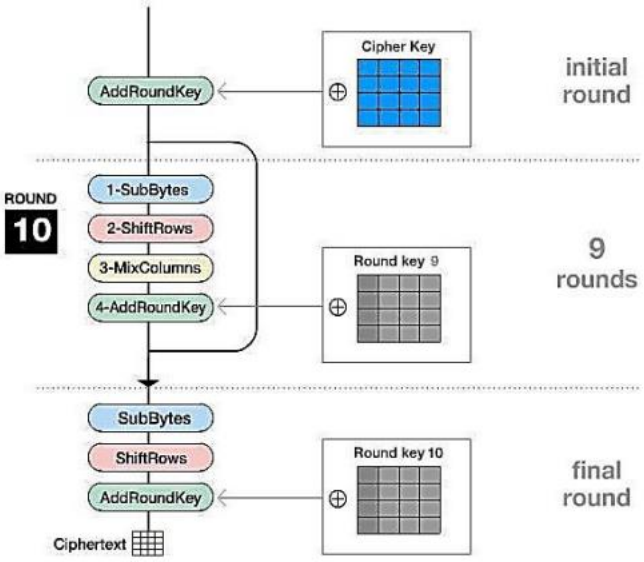


Gambar 1. Sistem Kriptografi

Diatas merupakan proses dari sistem kriptografi yang akan melakukan pengamanan suatu text dari awal sebelum pengamanan sampai melakukan pembukaan pengamanannya. Dilihat dari proses diatas text asli akan dilakukan enkripsi untuk mengamankan text aslinya dengan menggunakan kunci yang diinput. Lalu jika ingin membuka pengamanannya, maka akan dilakukan dekripsi yang dibutuhkan kunci sebelumnya pada saat melakukan pengamanan text. Setelah melalui dekripsi maka akan menghasilkan text seperti semula.

2.4 *Advanced Encryption Standard* (AES)

AES atau Advanced Encryption Standard adalah standar enkripsi kunci simetris yang awalnya diterbitkan oleh algoritma Rijndael. Algoritma ini dibuat dan dikembangkan oleh dua kriptografer Belgia, Joan Daemen dan Vincent Rijmen. Algoritma AES adalah algoritma simetris yang menggunakan kunci yang sama untuk enkripsi dan dekripsi. Algoritma AES memiliki tiga pilihan kunci yaitu tipe AES-128, AES 193, dan AES-256. Setiap tipe menggunakan kunci internal yang berbeda dan menggunakan round key untuk setiap putarannya.



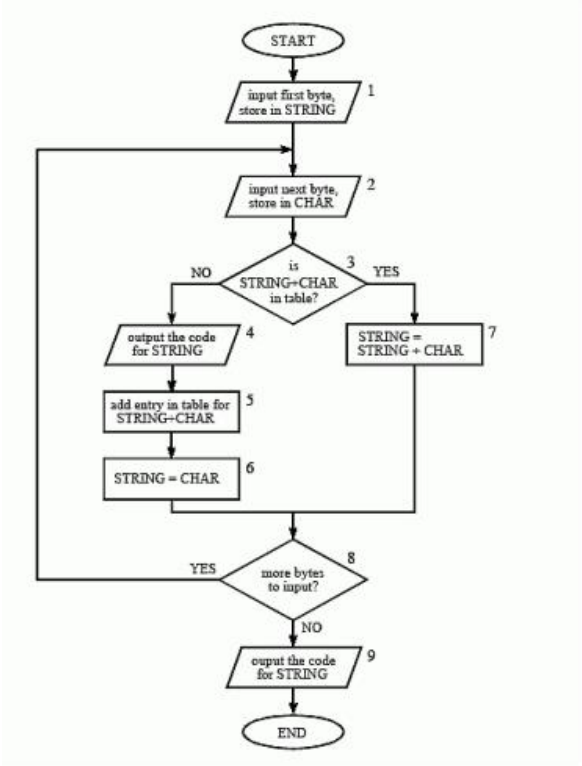
Gambar 2. Alur Kerja AES-128

Diatas merupakan alur kerja enkripsi AES-127 yang dilakukan sebanyak 10 kali (a=10), yaitu sebagai berikut:

a. Addroundkey : di tahap ini melakukan XOR antara *state* awal dengan *cipher key*. Tahap ini disebut juga *initial round.*
b. Putaran A-1, proses yang dijalankan pada setiap putaran adalah SubBytes yang melakukan substitusi byte menggunakan tabel pengganti (S-Box), ShiftRows yang menggeser baris-baris status array dengan cara membungkus, dan mengacak status array kolom. pindahkan setiap data ke Gunakan, dan AddRoundKey untuk melakukan XOR antar status menggunakan kunci bulat.
c. Final Round, adalah proses putaran untuk putaran terakhir yang meliputi SubBytes, ShiftRows, MixColumn, dan AddRoundKey.

2.5 *Lempel Ziv* Welch (LZW)

Algoritma lempel-ziv-welch merupakan algoritma kompresi yang dibuat oleh tiga sekawan yang bernama Abraham Lempel, Jacob Ziv, dan Terry Welch. Algoritma ini menggunakan teknik kompresi lossless yang artinya teknik kompresi yang tidak kehilangan data apapun dalam proses kompresi. Kompresi lossless "mengemas" data ke dalam ukuran *file* yang lebih kecil dengan menggunakan semacam singkatan internal untuk menandakan data yang berlebihan. Algoritma kompresi LZW tertentu mengambil setiap urutan input bit dengan panjang tertentu (misalnya, 12 bit) dan membuat entri dalam tabel (biasa disebut "kamus" atau "buku kode") untuk pola bit tertentu, yang terdiri dari pola itu sendiri dan kode yang lebih pendek.



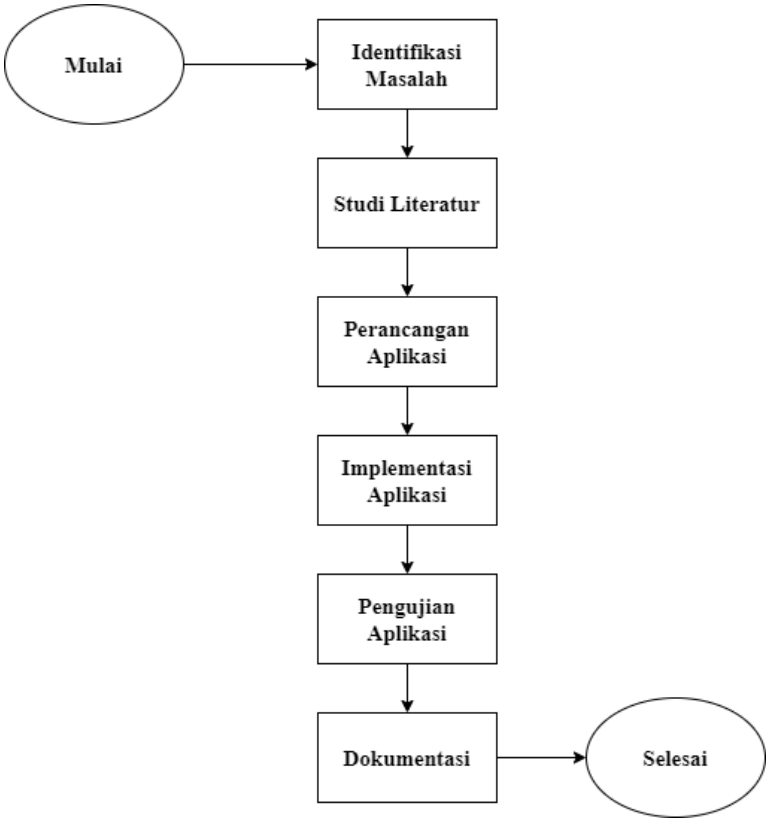
Gambar 3. Proses Kompresi LZW

Berikut adalah step dari kompresi LZW:

1. Menginput byte pertama lalu simpan ke dalam STRING atau kalimat
2. Menginput byte selanjutnya dan simpan ke dalam character
3. Cek apakah String + character ada di tabel yang tersimpan ?
4. keluarkan kode untuk string
5. tambahkan string+character ke dalam tabel
6. menghasilkan string = character
7. menambahkan string = string + character
8. ada lagi bytes yang ingin diinput ?
9. keluarkan kode untuk string

2.6 Metode Penelitian

Pada metodologi penelitian ini, tahapan dalam penelitian disajikan dalam bentuk *flowchart* di bawah ini:



Gambar 4. Flowchart Metodologi Penelitian

Penjelasan proses tahapan penelitian pada gambar 5 adalah sebagai berikut:

1. **Identifikasi Masalah**
   Pada tahap ini penulis memperlihatkan permasalahan yang ada pada bidang keamanan data, khususnya pada *file* yang bersifat privasi dan sensitif. Tingkat kerentanan pencurian data pun masih tinggi tanpa adanya pengamanan tambahan. Beserta juga keamanan terhadap file yang disimpan dalam database cloud yang dapat sangat mudah diakses dengan menggunakan internet sehingga terjadinya kebocoran data masih tinggi.

2. **Studi Literatur**
   Pada tahap ini, penulis melakukan pengumpulan data informasi berupa konsep dan teori yang berhubungan dengan kriptografi, cloud computing, cloud storage, pengamanan *file*, serta penelitian yang berkaitan dengan algoritma AES dan kompresi LZW.

3. **Perancangan Aplikasi**
   Pada tahap ini penulis akan merancang sistem yang akan mengkombinasikan dua algoritma yaitu algoritma kriptografi AES dan algoritma kompresi LZW yaitu dengan proses kunci dan buka *file* yang sudah diproses.

4. **Implementasi Aplikasi**
   Pada penelitian ini penulis akan menggunakan kombinasi antara algoritma kriptografi AES dengan algoritma kompresi LZW untuk melakukan pengamanan *file* dokumen dan juga memperkecil ukuran *file* setelah melalui proses enkripsi. Implementasi dari program tersebut sesuai dengan flowchart serta digunakan dalam bahasa pemrograman dalam bentuk program aplikasi.

5. **Pengujian Aplikasi**
   Pada tahap ini akan dilakukan pengujian menggunakan *file* yang dihasilkan setelah melalui proses enkripsi lalu kompresi dan dekompresi serta dekripsi mulai dari ukuran *file*, waktu yang diperlukan, serta hasil rasio dari kompresi, kemudian untuk perbandingan *file* asli dengan *file* hasil akan dilakukan uji coba menggunakan Checksum.

6. **Dokumentasi**
   Aplikasi yang telah siap digunakan setelah tahap pengujian akan didokumentasikan dengan tujuan bisa dimanfaatkan oleh masyarakat dan peneliti lain yang berhubungan dengan topik saat ini.

3 Hasil dan Pembahasan

3.1 Pengumpulan data

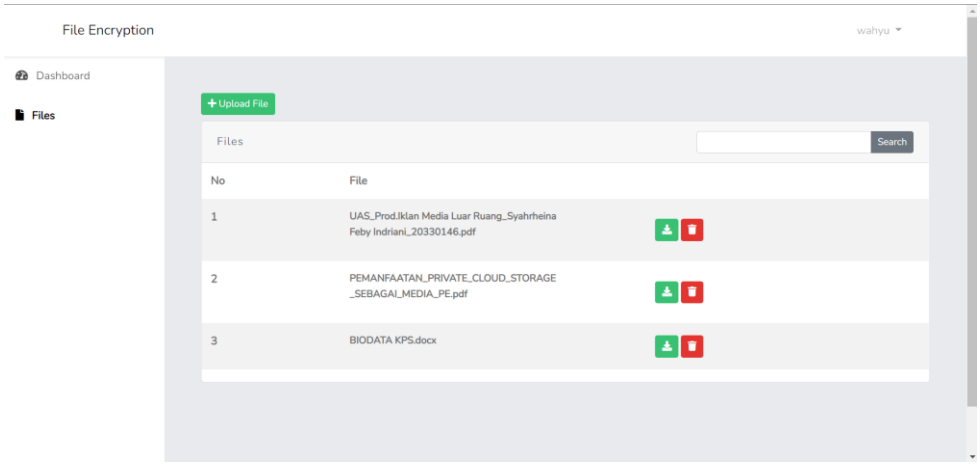
Data yang digunakan dalam penelitian ini bersumber dari 2 tipe data dokumen acak yang diperoleh dari internet dan juga instansi yang memiliki *file* dokumen yang sesuai formatnya. *File* yang akan diganakan memiliki informasi mengenai kebijakan dan juga ketentuan dalam bentuk teks yang tersimpan kedalam format sebagai berikut:

Table 1. Sumber Data

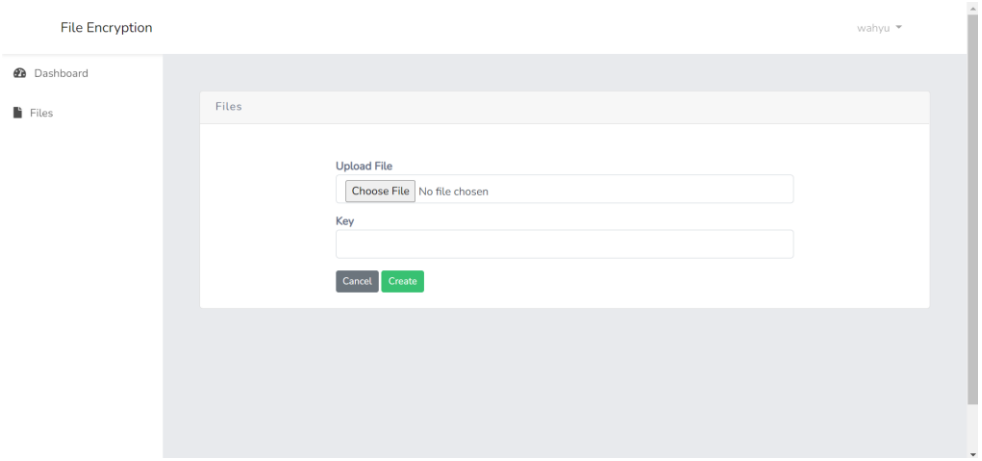
No	Type *File* (format)	Keterangan
1	.doc	Tipe format *file* (.doc) didapatkan dari aplikasi pengolahan kalimat seperti Microsoft office word, dalam *file* ini juga memiliki informasi berupa gambar, teks, tabel, dan grafik.
2	.pdf	Tipe format *file* (.pdf) didapatkan dari pengolahan data dan gambar seperti microsft office dan adobe acrobat, dalam *file* ini juga memiliki informasi berupa teks, gambar, tabel, dan juga grafik.

3.2 Perancangan Aplikasi

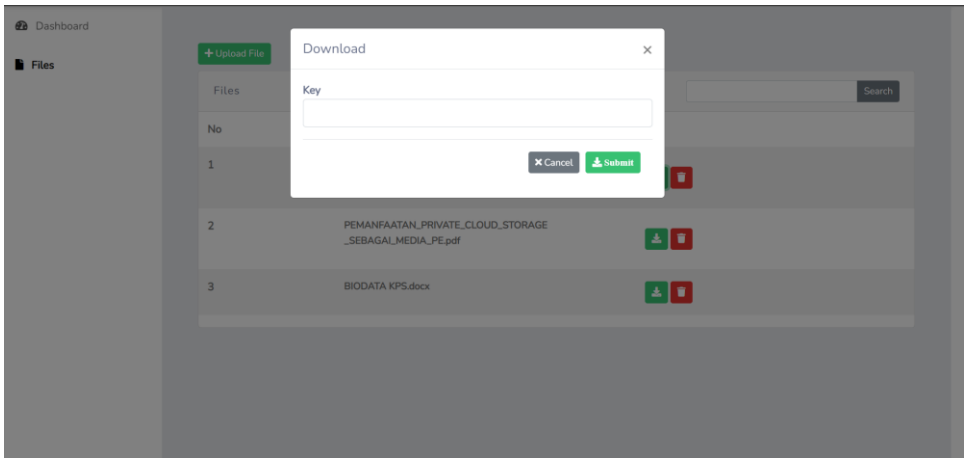
Aplikasi yang dibuat dalam penelitian ini berdasarkan dari sebuah rancangan yang telah dibuat sebelumnya seperti rancangan UML, rancangan basisdata sehingga aplikasi ini berjalan sesuai dengan tujuan penelitian ini dibuat dan bisa digunakan oleh user lainnya. Berikut adalah hasil dari rancangan tersebut.



Gambar 5. Menu utama dalam melakukan upload atau download *file*.



Gambar 6. Form untuk mengupload *file* dan menginput key untuk penguncian berkas yang digunakan untuk proses enkripsi dengan AES dan kompresi dengan LZW serta tersimpan ke dalam cloud storage.



Gambar 7. proses melakukan download yang diharuskan menginput key untuk membuka berkas yang sudah dilakukan proses enkripsi sebelumnya.

3.3 Pengujian Aplikasi

Pengujian aplikasi yang dilakukan dalam penelitian ini bertujuan untuk melihat sejauh mana penguncian dan pembukaan berkas ini berhasil. Apakah terdapat perubahan dari saat berkas melakukan penguncian dan kompresi, kemudian apakah jika berkas dapat dilihat jika belum melalui proses pembukaan.

Table 2. Pengujian ukuran *file* format .doc

No	Nama *file*	*File* asli	*File* setelah enkripsi	*File* setelah enkripsi dan kompresi	Rasio
1	test 1.doc	12kb	21kb	48kb	300%
2	test 2.doc	19kb	34kb	75kb	295%
3	test 3 .doc	16kb	28kb	62kb	288%
4	test 4.doc	41kb	73kb	157kb	283%
5	test 5.doc	13kb	23kb	51kb	292%

Table 3. Pengujian Ukuran *file* forma .pdf

No	Nama *file*	*File* asli	*File* setelah enkripsi	*File* setelah enkripsi dan kompresi	Rasio
1	test 1.pdf	37kb	66kb	143kb	286%
2	test 2.pdf	92kb	163kb	341kb	271%
3	test 3.pdf	36kb	64kb	138kb	283%
4	test 4.pdf	54kb	96kb	205kb	280%
5	test 5.pdf	13kb	23kb	53kb	308%

Tabel 2 dan 3 merupakan hasil dari pengujian ukuran dari *file* asli yang telah melalui proses penguncian dengan panjang kunci yang sama sehingga mendapatkan hasil dari pengaruh penggunaan kriptografi dengan AES dan kompresi dengan LZW adalah sebagai berikut.

Table 4. Pengujian Waktu *file* format .doc

No	Nama *file*	*File* Awal	*File* Tersimpan	Waktu
1	test 1.doc	12kb	48kb	3.29 s
2	test 2.doc	19kb	75kb	6.10 s
3	test 3 .doc	16kb	62kb	4.26 s
4	test 4.doc	41kb	157kb	24.57 s
5	test 5.doc	13kb	51kb	3.29 s

Table 5. Pengujian Waktu *file* format .pdf

No	Nama *file*	*File* Awal	*File* Tersimpan	Waktu
1	test 1.pdf	37kb	143kb	1.10 m
2	test 2.pdf	92kb	341kb	6.58 m
3	test 3.pdf	36kb	138kb	1.05 m
4	test 4.pdf	54kb	205kb	2.24 m
5	test 5.pdf	13kb	53kb	0.11 s

Tabel 4 dan 4 adalah hasil pengujian pada waktu proses eksekusi enkripsi menggunakan AES dan proses kompresi menggunakan LZW serta penyimpanan ke dalam cloud storage. Dalam percobaan diatas juga dilihat waktu penyimpanan dan proses penguncian memiliki waktu yang sedikit berdekatan yang artinya waktu tersebut berbanding lurus berdasarkan besar dari *file* yang akan diproses. Jika *file* semakin besar maka diperlukan juga waktu yang lebih lama.

Table 6. Pengujian checksum *file* format .doc

No	Nama *file*	Keterangan *file*	Nilai Checksum	Keterangan
1	test 1.doc	*File* asli	566DDBA71E0C32A3162253855EDA1138	-
		Penguncian	D4ACD2137EC07AD79BEEA500FEB5A43F	Berbeda

		pembukaan	566DDBA71E0C32A3162253855EDA1138	Sama
2	test 2.doc	*File* asli	B753CAC760D20DFAAFCD06B67BF672F9	-
		Penguncian	8CBD7F0B54191174ED4B34B9DB80BF14	Berbeda
		pembukaan	B753CAC760D20DFAAFCD06B67BF672F9	Sama
3	test 3.doc	*File* asli	2F9685F43C9D55D4B9CD7F65CCF17A0C	-
		Penguncian	271C8F7717305691933BD05B2A620774	Berbeda
		pembukaan	2F9685F43C9D55D4B9CD7F65CCF17A0C	Sama
4	test 4.doc	*File* asli	9519515FA5725CBC708B1A6186462C00	-
		Penguncian	0CF9701764D3FE10A4202CD990518549	Berbeda
		pembukaan	9519515FA5725CBC708B1A6186462C00	Sama
5	test 5.doc	*File* asli	39264AE1F852C8B2321B0678081A537C	-
		Penguncian	9519515FA5725CBC708B1A6186462C00	Berbeda
		pembukaan	39264AE1F852C8B2321B0678081A537C	Sama

Table 7. Pengujian checksum *file* format .pdf

No	Nama *file*	Keterangan *file*	Nilai Checksum	Keterangan
1	test 1.pdf	*File* asli	FC73EDA416435DCF5BAF928C3C890ED1	-
		Penguncian	30D24C25ECD9F58F9508903AC7C9D3D4	Berbeda
		pembukaan	FC73EDA416435DCF5BAF928C3C890ED1	Sama
2	test 2.pdf	*File* asli	CDF175A2BC067514CBC2A9A36B2A4863	-
		Penguncian	B3777DE81084B4B8322734176BF03FD1	Berbeda
		pembukaan	CDF175A2BC067514CBC2A9A36B2A4863	Sama
3	test 3.pdf	*File* asli	1841F06CE08097487D72D60475D17C15	-
		Penguncian	3B2223B711B981CB61408A14F0417481	Berbeda
		pembukaan	1841F06CE08097487D72D60475D17C15	Sama
4	test 4.pdf	*File* asli	1657C2F2E38814DF49034E87AB14AD71	-
		Penguncian	1113A3E8FC348ECAE89AB5D73C504670	Berbeda
		pembukaan	1657C2F2E38814DF49034E87AB14AD71	Sama
5	test 5.pdf	*File* asli	9C984BEE314622DEBEEE9E7A582C0E1D	-
		Penguncian	794AC095D34CB26EC451260A109E1B2B	Berbeda
		pembukaan	9C984BEE314622DEBEEE9E7A582C0E1D	Sama

Tabel 6 dan 7 merupakan hasil dari pengujian dengan menggunakan checksum yang dimana nilai dari *file* asli dibandingkan dengan *file* hasil kunci dan *file* asli pembukaan apakah terdapat perubahan nilai atau tidak. Dilihat pada tabel diatas memiliki nilai checksum yang berbeda dan juga sama dengan file aslinya. Nilai checksum akan berbeda dengan aslinya karena telah melalui proses penguncian yaitu enkripsi yang akan mengubah nilai di

dalamnya secara acak dan melakukan proses kompresi untuk memperkecil ukuran file aslinya. Lalu selanjutnya nilai checksum akan sama dengan aslinya karena telah melalui proses pembukaan yaitu melalui proses dekompresi dan dekripsi untuk mengembalikan nilai yang sebelumnya acak menjadi seperti semula kembali.

4 Kesimpulan

Berdasarkan hasil pengujian yang dilakukan pada bab sebelumnya dengan menggunakan algoritma Kriptografi AES (Advanced Encryption Standard) dan dikombinasikan dengan algoritma kompresi LZW (Lempel Ziv Welch). Maka dapat disimpulkan dari hasil pengujian ini sebagai berikut:

1. Kombinasi dari algoritma AES (Advanced Encryption Standard) dan algoritma kompresi LZW (Lempel Ziv Welch) dapat digunakan dalam pengamanan *file* dokumen. Dari kombinasi algoritma sebelumnya menghasilkan *file* yang tidak dapat dilihat setelah proses enkripsi.
2. Berdasarkan pengujian yang dilakukan dalam penelitian ini, metode algoritma AES dalam melakukan enkripsi *file* dan metode algoritma LZW dalam melakukan kompresi *file* mempengaruhi ukuran *file* aslinya. Dengan melakukan proses enkripsi dan kompresi, *file* tersebut akan mengalami kenaikan dari ukuran *file* aslinya.
3. Dalam penelitian ini algoritma kompresi LZW memberikan pengaruh terhadap ukuran *file* setelah dilakukan proses pengamanan *file* menggunakan algoritma AES yang berakibat ukuran *file* setelah dilakukan kompresi mengalami kenaikan daripada ukuran *file* asli yang disebabkan oleh enkripsi file yang mengakibatkan isi file tersebut memiliki nilai secara acak dan proses kompresi LZW ini menggunakan singkatan untuk internal untuk mengganti nilai yang sama serta *file* setelah enkripsi yang memiliki presentase rata-rata diatas 200 persen lebih besar dari ukuran *file* aslinya.

Referensi

[1] Herwanto, Riko, Aziz, R., & Purbo, O. W. Cloud Computing : Manajemen dan Perencanaan Kapasitas. 2021.

[2] Munir,Rinaldi, KRIPTOGRAFI. Bandung, Informatika, 2019.

[3] Jamaludin, & Romindo. Kriptografi: Teknik Hybrid Cryptosystem Menggunakan Kombinasi Vigenere Cipher dan RSA. 2020.

[4] Mirsyah, M. A.-A., Aksara, L. F., & Sajiah, A. M. Penerapan Algoritma Advanced Encryption Standard (AES) Untuk Mengamankan *File* Pada Layanan Infrastructure as a Service. 2019.

[5] Dheemanth, H N. LZW Data Compression, American Journal of Engineering Research (AJER), pp-22-26 e-ISSN : 2320-0847. 2014.

[6] Giap, Y., Kurnaedi, D., Nursanty, E., Nugroho, M., Simarmata, J., & Ardilla, Y. *Cloud Computing: Teori dan Implementasi.* Medan: Yayasan Kita Menulis. ISBN: 978-623-6512-26-5. 2020.

[7] K.M., A., Kumar M, P., & B.R., P. Enhanced Cloud Data Security Using AES Algorithm. *International Conference on Intelligent Computing and Control*. DOI: 10.1109/I2C2.2017.8321820. 2017.

[8] Bahri, G. Perancangan dan Implementasi Sistem Manajemen Peminjaman Mobil Dengan Metode Scrum di Universitas Internasional Batam. *UIB Repository*. 2019.

[9] Herlambang, Panjianom B. Implementasi Algoritma Kriptografi AES (Advanced Encryption Standard) dan Algoritma Kompresi LZW (Lempel Ziv Welch) Pada Citra Digital. *UPNVJ Repository.* 2022.

[10] Wibowo, Rizky S. Implementasi Keamanan File Dengan Kompresi Huffman dan Kriptografi Advanced Encryption Standard (AES) Pada Pengamanan File Data Antemortem. *UPNVJ Repository.* 2021.