BAB II
LANDASAN TEORI

2.1 Studi Literatur

Dalam penyusunan untuk riset yang dilakukan, penulis menghimpun studi yang terkait dengan konteks dari beberapa penelitian sebelumnya sebagai pedoman dan acuan untuk merancang studi tentang penerapan sistem *file encryption* pada sistem operasi Linux untuk memastikan kerahasiaan data pengguna. Perkembangan perangkat lunak yang menangani keamanan komputer semakin canggih, terutama dalam keamanan sistem operasi. Sistem operasi berusaha menjamin keamanan data pengguna seperti mengantisipasi tidak terjadinya pencurian data atau perbuatan ilegal sejenis lainnya dengan menambahkan fitur khusus untuk menangani masalah keamanan data dengan menerapkan algoritma Kriptografi tertentu.

2.2 Sistem Operasi

Sistem operasi secara umum ialah pengelola seluruh sumber daya yang terdapat pada sistem komputer dan menyediakan sekumpulan layanan *(System Call)* ke pemakai sehingga memudahkan dan menyamankan penggunaan serta pemanfaatan sumber daya sistem komputer tersebut.

Linux adalah salah satu sistem operasi yang sering digunakan di server, perangkat seluler, sistem terbenam, dan bahkan desktop. Namun, sebagai sistem operasi yang terbuka dan dapat diakses, Linux memiliki risiko keamanan yang perlu diatasi. Salah satu cara untuk memitigasi risiko ini adalah melalui penerapan sistem *file encryption*.

2.3 Linux

Linux adalah sebuah sistem operasi komputer. Pembangunan Linux dimulai tahun 1990 oleh Linus Torvald, seorang mahasiswa dari *University of Helsinki*, Finlandia. Saat ini, Linux adalah salah satu pilihan sistem operasi selain Microsoft Windows, UNIX, Solaris, QNX dan beberapa sistem operasi lainnya.

Sebagai sistem operasi, Linux bertugas mengambil kendali infrastruktur perangkat keras *(hardware)* ketika pertama kali sistem tersebut dijalankan. Nama Linux adalah nama inti dari sistem operasi itu saja *(kernel)*, bukan seluruh aplikasi yang terpaket dalam CD atau aplikasi-aplikasi yang berjalan di atasnya, namun sudah umum diketahui bahwa sistem operasi dengan kernel Linux disertai aplikasi-aplikasi pendukungnya disebut secara keseluruhan sebagai Linux.

Lisensi kernel Linux yang free (dalam arti merdeka dan tidak selalu gratis) maka setiap komunitas dapat memaket CD instalasi Linux dengan aplikasi lain sesuai dengan kebutuhan masing-masing. Ada yang memaket *kernel* Linux dengan aplikasi server untuk membangun server yang handal dan terpercaya, tetapi ada juga yang memaketnya dengan aplikasi office untuk membangun workstation yang user-friendly. Paket-paket aplikasi yang melengkapi kernel Linux secara keseluruhan disebut sebagai distribusi Linux atau distro Linux.(Burlian et al., 2022).

2.4 *Pretty Good Privacy* (PGP)

PGP *(Pretty Good Privacy)* adalah suatu metode program enkripsi informasi yang memiliki tingkat keamanan cukup tinggi bersifat rahasia dengan menggunakan *"PrivatePublic Key"* sebagai dasar autentifikasinya sehingga jangan sampai dengan mudah diketahui oleh orang lain yang tidak berhak. PGP membuat sebuah *session key*, dimana sebuah kunci rahasia pada saat itu. *Session Key* ini berkerja dengan sangat aman, algoritma enkripsi konvesional yang cepat untuk meng-enkrip *plaintext*. Hasilnya adalah berupah chiper text. Sekali data dienkripsi, lalu *session key* ini dienkripsi lagi menggunakan kunci publik penerima. *session key* yang terenkripsi kunci *publickey* penerima dikirim dengan *chipertext* ke penerima.

Proses deskripsi bekerja sebaliknya, Penerima menerima pesan lalu membuka pesan tersebut dengan kunci privatnya, namun pesan tersebut masih terenkripsi dengan *session key*. Dengan Menggunakan PGP, penerima mendekrip *chipertext* yang terenkripsi secara konvensional. Kombinasi dari 2 metode enkripsi menggabungkan kehandalan dari enkripsi kunci publik

dengan kecepatan pada enkripsi konvensional. Enkripsi Konvensional kuarang lebih 1000x lebih cepat dari enkripsi kunci publik. Jadi enkripsi kunci publik memberikan sebuah solusi pada distribusi kunci dan masalah transmisi data. Dengan menggunakan keduanya, perfoma dan distribusi kunci dapat ditingkatkan tanpa mengorbankan sesuatu dalam keamanan.*(Pramana Hostiadi & Suradarma, 2017).*

2.5 *GNU Privacy Guard* (GPG)

GnuPG adalah software enkripsi pengganti PGP yang lengkap dan bebas (lisensi GPL). Dibuat oleh tim GnuPG yang terdiri dari *Matthew Skala*, *Michael Roth, Niklas Hernaeus, R Guyomarch and Werner Koch. Gael Queri, Gregory Steuck, Janusz A. Urbanowicz, Marco d'Itri, Thiago Jung Bauermann, Urko Lusa and Walter Koch* yang membuat translasi resmi dan *Mike Ashley* yang mengerjakan *GNU privacy handbook*. GnuPG adalah suatu program yang digunakan untuk mengamankan komunikasi dan penyimpanan data. Program ini dapat menyandikan data serta membuat tanda tangan digital. Karena tidak menggunakan algoritma yang dipatenkan, GnuPG dapat digunakan secara bebas.

GnuPG menggunakan kriptografi *Public Key (Public Key Cryptography)* sehingga para penggunanya dapat saling berkomunikasi secara aman. Dalam sistem *Public Key*, setiap pengguna mempunyai sepasang kunci yang terdiri dari *Private key* dan *Public key*. *Private key* dirahasiakan; hanya diketahui oleh pemiliknya, sementara *Public key* dapat diberikan pada siapa saja yang dikehendaki pemilik, sehingga pemilik dapat berkomunikasi dengan pengguna lain yang diberi *Public key* tersebut. Bebas karena tidak menggunakan algoritma enkripsi yang telah dipatenkan sehingga bisa dipakai oleh siapa saja tanpa batasan. GnuPG memenuhi spesifikasi *OpenPGP* RFC2440. *(Purwanto & Informatika, 2020)*

Beberapa fitur yang ditawarkan GnuPG adalah: penggantian penuh terhadap pemakaian PGP.

1. Dapat digunakan sebagai pengganti PGP (yang dipatenkan algoritmanya).
2. Tidak menggunakan algoritma yang dipatenkan.
3. Berlisensi GPL.
4. Ditulis dari nol, sehingga tidak menggunakan kode sumber atau algoritma dari program lainnya.
5. Implementasi penuh *OpenPGP* (RFC 2440)

2.6 Algoritma Pada *GNU Privacy Guard* (GPG)

Dalam melakukan eksperimen ditahap ini peneliti menggunakan GnuPG versi 2.4.3 mendukung beberapa algoritma antara lain :

Algoritma Asimetrik

(RSA, RSA-E, RSA-S, ELG, DSA)

Algoritma kunci umum memiliki dua jenis kunci, yaitu kunci kunci umum dan kunci rahasia

Algoritma Simetrik :

(AES,3DES, BLOWFISH, AES192, AES256, TWOFISH, CAMELLIA)

Algoritma yang menggunakan kunci yang sama untuk proses enkripsi dan proses deskripsi.

Algoritma Hash :

(MD5, SHA1, RIPEMD160, SHA256, SHA512)

Fungsi Hash memproses seluruh dokumen atau pesan sebagai *input* dan menghasilkan nilai Hash, yang sering disebut sebagai *fingerprint* (sidik jari) dari dokumen dengan ukuran khusus, umumnya sebesar 1 atau 20 byte.

Perbandingan *GNU Privacy Guard (GPG) & Pretty Good Privacy (PGP)*

Berikut merupakan perbandingan yang didapat berdasarkan hasil perbandingan antara GPG & PGP Sebagai acuan :

Tabel 2. 1 Perbandingan antara GPG dan PGP

	PGP	GPG
Fungsi	PGP memiliki keterbatasan saat digunakan baik untuk keperluan pribadi maupun komersial.	GPG bisa digunakan baik untuk kepentingan pribadi maupun komersial dengan cara mengunduh tanda tangan digital secara gratis bersama program terenkripsi.
Lisensi	PGP Corporation	GPL *(GNU Public Lisence)*
Algoritma	Program PGP menggunakan algoritma enkripsi RSA dan IDEA.	NIST AED, yang merupakan algoritma enkripsi standar yang lebih mutakhir, digunakan dalam GPG.
Sertifikat Dukungan	X.509, *OpenPGP*	*OpenPGP*, *Web of Trust*

2.7 Seahorse

Seahorse merupakan sebuah aplikasi manajemen kunci dan sertifikat yang umumnya digunakan pada lingkungan desktop GNOME di sistem operasi Linux. Dengan *Seahorse*, pengguna dapat dengan mudah mengelola kunci enkripsi, sertifikat SSL/TLS, dan kunci SSH. Aplikasi ini menyediakan antarmuka grafis yang intuitif, memungkinkan pengguna untuk membuat, mengimpor, mengekspor, dan mengatur kunci serta sertifikat dengan mudah. Keamanan adalah aspek penting dalam penggunaan *Seahorse* dan GPG.

Evaluasi keamanan, penerapan praktik keamanan dan peningkatan fitur keamanan dalam kedua aplikasi tersebut menjadi yang utama. Dalam hal ini dapat membantu bagaimana *Seahorse* dan GPG dapat digunakan untuk

melindungi informasi sensitif dan menjaga kerahasiaan data di lingkungan Linux. *Seahorse* juga memungkinkan pengguna untuk melakukan berbagai operasi enkripsi, seperti mengenkripsi dan mendekripsi *file*, serta mengirim dan menerima pesan yang dienkripsi. Dengan fitur-fitur ini, pengguna dapat menjaga keamanan data sensitif mereka dan berkomunikasi secara aman di lingkungan Linux.

2.8 Keamanan

Saat ini, sistem komputer telah secara luas terhubung dengan jaringan, sehingga memudahkan akses namun juga meningkatkan kerentanan terhadap ancaman keamanan, seperti kejahatan komputer. Terutama dengan masyarakat yang kini bergantung pada komputer untuk pembuatan, penyimpanan, dan pengaturan informasi-informasi penting seperti data keuangan, informasi pribadi, serta data perusahaan. Oleh karena itu, penting bagi pengguna dan pengelola sistem komputer untuk menjaga keamanan perangkat dan informasi mereka agar tidak terancam oleh kerusakan, kehilangan, atau penyalahgunaan. Istilah keamanan (*security*) dan proteksi (*protection*) sering kali digunakan secara bergantian.

Untuk menghindari kekeliruan, istilah keamanan merujuk pada semua aspek keamanan, sedangkan istilah proteksi mengacu pada mekanisme sistem yang digunakan untuk melindungi informasi dalam sistem komputer. Keamanan sistem operasi merupakan bagian integral dari permasalahan sistem komputer secara keseluruhan, namun telah menjadi semakin penting(Wahid, 2019). Langkah-langkah perlindungan secara fisik dengan membatasi akses fisik secara langsung ke fasilitas sistem komputer juga perlu dilakukan. Keamanan pada komputer mencakup berbagai aspek yang meliputi (Abdul et al., 2019):

a) Otentikasi *(Authentication)*:

   Membantu penerima informasi memastikan bahwa pesan tersebut berasal dari sumber yang seharusnya, sehingga keaslian informasi dapat diverifikasi, dan pesan tersebut benar-benar berasal dari pihak yang diharapkan.

b) Integritas *(Integrity)*:

Menjamin bahwa pesan yang dikirim melalui jaringan tetap utuh dan tidak mengalami modifikasi oleh pihak yang tidak berhak selama proses pengiriman.

c) Nonrepudiasi *(Non-repudiation)*:

Mengaitkan pengirim dengan pesan yang dikirimnya sehingga pengirim tidak dapat menyangkal bahwa dia adalah pengirimnya.

d) Otoritas *(Authority)*:

Mencegah pihak yang tidak berhak untuk mengubah informasi yang ada dalam sistem jaringan.

e) Kerahasiaan *(Confidentiality)*:

Berfokus pada usaha menjaga informasi agar tidak dapat diakses oleh pihak yang tidak berhak. Biasanya berkaitan dengan informasi yang diberikan kepada pihak lain.

f) Privasi *(Privacy)*:

Lebih menitikberatkan pada data pribadi yang harus dijaga dari akses yang tidak sah.

g) Ketersediaan *(Availability)*:

Terkait dengan memastikan bahwa informasi dapat diakses ketika dibutuhkan. Serangan atau gangguan terhadap sistem informasi dapat menghambat atau bahkan menghalangi akses ke informasi.

h) Pengendalian Akses *(Access Control)*:

Mengatur cara akses ke informasi. Ini seringkali melibatkan penggunaan kombinasi ID pengguna dan kata sandi atau mekanisme lain untuk mengelola siapa yang memiliki akses ke informasi tertentu.

2.9 Kriptografi

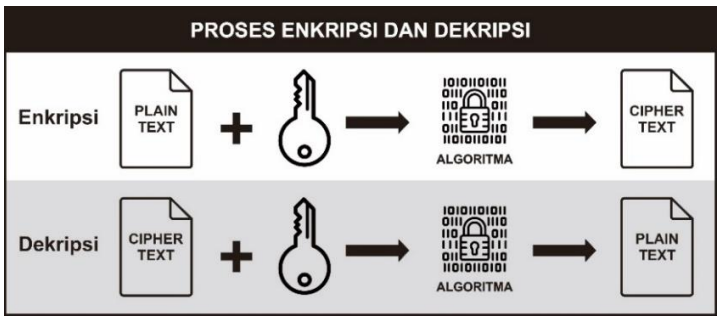
Kriptografi adalah bidang ilmu yang mempelajari teknik penyandian dengan cara mengubah dan merubah teks asli *(Plain Text)* menggunakan kunci dan algoritma tertentu sehingga teks diubah menjadi tidak dapat dikenali atau

dibaca *(Ciphertext)* oleh pihak yang tidak memiliki kunci untuk mengembalikannya ke bentuk asli (Aufia et al., 2021).

Proses mengubah dan mengacak teks asli tersebut disebut enkripsi, sementara dekripsi adalah langkah untuk mengembalikan teks yang sudah diubah menjadi bentuk aslinya. Kedua proses utama kriptografi ini memerlukan kunci untuk merubah dan mengembalikan informasi.

Ketika penerima informasi tidak memiliki kunci, maka dibutuhkan waktu yang sangat lama untuk mengembalikan teks ke bentuk asli, dan ada kemungkinan bahwa tidak dapat dikembalikan sama sekali.

Pada gambar 2.1 di bawah ini merupakan proses enkripsi secara sederhana.



Gambar 2. 1 Proses Enkripsi Dekripsi

2.9.1 Komponen Kriptografi

Pada Umumnya komponen kriptografi terdiri dari beberapa komponen, yaitu:

a) Enkripsi

Merupakan proses mengubah pesan asli (plainteks) menjadi kode yang sulit dipahami. Dalam istilah lain, enkripsi juga disebut sebagai cipherteks atau pesan yang bersifat acak.

b) Dekripsi

Dekripsi merupakan proses mengubah data yang sebelumnya dienkripsi menjadi suatu format tertentu, lalu mengembalikannya ke bentuk asalnya. Contohnya adalah mengonversi kode-kode yang awalnya berupa hash dan biner menjadi bentuk semula.(Herwanto & Kom, 2022)

c) Kunci

Kunci kriptografi adalah elemen penting dalam proses enkripsi dan dekripsi. Ada dua jenis kunci utama: kunci publik *(public key)* dan kunci

privat *(private key)*. Kunci publik digunakan untuk enkripsi, sementara kunci privat digunakan untuk dekripsi. Dalam kriptografi simetris, kunci yang sama digunakan untuk enkripsi dan dekripsi.

d) *Ciphertext*

Merupakan pesan yang telah diacak atau dienkripsi. Pesan yang telah dienkripsi sulit dibaca karena terdiri dari karakter yang tidak memiliki makna langsung.

e) *Plainteks*

Sering disebut *cleartext*, merujuk pada teks asli yang belum mengalami proses apa pun dan mudah dipahami karena memiliki makna yang jelas.

f) Pesan

Berupa data atau informasi yang dikirimkan atau disimpan dalam media perekaman, Pesan ini bisa berupa teks, gambar, audio, atau bentuk data lainnya.

g) *Cryptanalysis*

Merujuk pada upaya membaca teks yang telah dienkripsi, menganalisis sistem kriptografi dengan tujuan untuk mengidentifikasi kelemahan atau mendekripsi pesan tanpa memiliki kunci yang tepat.

2.9.2 Algoritma Kriptografi

Bagian dari algoritma kriptografi meliputi teknik enkripsi dan dekripsi. Teknik enkripsi merupakan proses mengubah pesan asli menjadi bentuk pesan acak yang sulit dipahami, sering kali disebut sebagai cipherteks. Sementara teknik dekripsi adalah langkah untuk mengembalikan pesan yang telah dienkripsi ke dalam bentuk semula, yang dikenal sebagai plainteks (Aufia et al., 2021). Algoritma kriptografi terdiri dari 2 bagian, yaitu :

a. Algoritma Simetris

Algoritma simetris (algoritma kriptografi konvensional) adalah algoritma yang menggunakan kunci yang sama untuk proses enkripsi dan proses deskripsi. Algoritma kriptografi simetris dibagi menjadi dua kategori yaitu algoritma aliran (*Stream Ciphers*) dan algoritma blok (*Block Ciphers*).

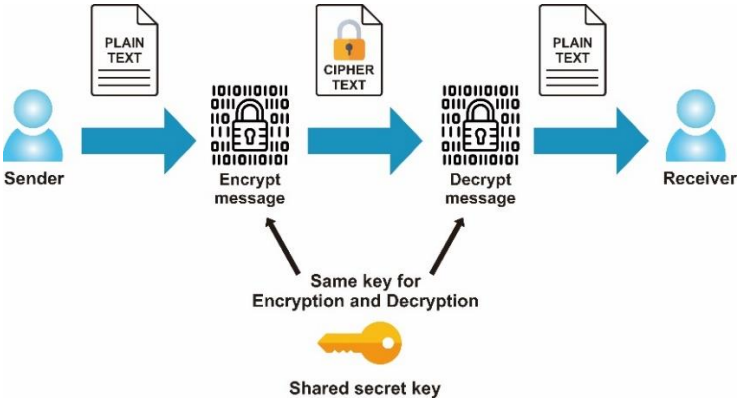
Dimana pada algoritma aliran, proses penyandiannya akan beriorientasi pada satu bit/byte data. Sedangkan pada algoritma blok, proses penyandiannya berorientasi pada sekumpulan bit/byte data (per blok).

(Mu'alimin Arrijal et al., 2016)

Algoritma yang memakai kunsi simetris antara lain adalah :

1. AES (*Advanced Encryption Standard*):
2. DES (*Data Encryption Standard*):
3. 3DES (*Triple Data Encryption Standard*):
4. IDEA (*International Data Encryption Algorithm*):
5. RC2, RC4, RC5, RC6.

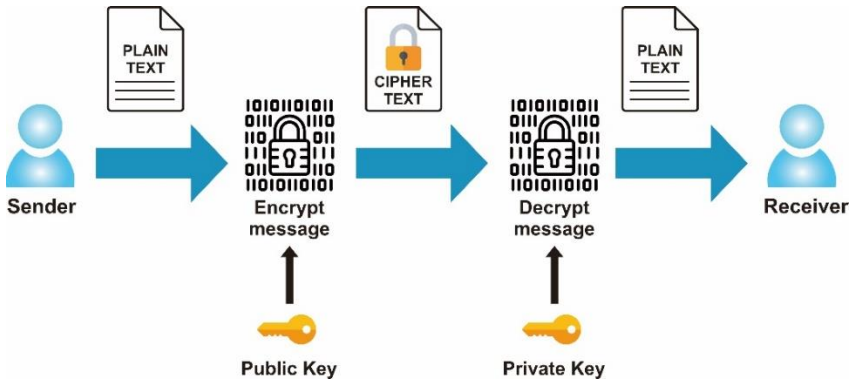
Skema cara kerja kriptografi simetris pada gambar 2.2



Gambar 2. 2 Skema Kriptografi Simetris

b. Algoritma Asimetris

Enkripsi asimetris menggunakan pasangan kunci yang berbeda untuk melaksanakan proses enkripsi dan dekripsi. Ketika melakukan enkripsi, maka kunci publik yang dipakai, sementara untuk dekripsi, kunci privat yang diaplikasikan. Skema cara kerja kriptografi simetris pada gambar 2.3



Gambar 2. 3 Skema Kriptografi Asimetris

Kriptografi asimetris beroperasi dengan cara diawali oleh pihak pengirim *(Sender)* yang mengirimkan data dalam format *plaintext* (berkas dalam keadaan asli). Langkah selanjutnya adalah melalui proses enkripsi menggunakan kunci A (*Public Key*). Berkas yang telah dienkripsi kemudian berubah menjadi *ciphertext* (berkas yang telah diacak). Setelah *ciphertext* diterima oleh pihak penerima, pihak penerima akan mendekripsi berkas *ciphertext* tersebut menggunakan kunci B (*Private Key*), sehingga kembali menjadi *plaintext*. Proses penciptaan kunci B dilakukan berdasarkan suatu algoritma yang dibuat berdasarkan kunci A. Contoh Algoritma yang memakai kunci asimetris antaranya adalah :

1. RSA (Rivest-Shamir-Adleman)
2. ECC (Elliptic Curve Cryptography)
3. DSA (Digital Signature Algorithm)
4. ElGamal
5. Diffie-Hellman Key Exchange

2.9.3 RSA (Rivest-Shamir-Adleman)

Algoritma RSA merupakan salah satu algoritma kunci asimetris. RSA (RivestShamir-Adleman) adalah sebuah kriptografi kunci publik yang berdasarkan pada eksponensial terbatas bilangan bulat (ZN) dimana N adalah sebuah bilangan bulat gabungan dari dua faktor besar yaitu (semiprime). RSA memiliki keamanan yang tinggi dikarenakan penggunaan dua kunci yang berbeda pada proses enkripsi dan dekripsinya dan sulitnya memfaktorkan bilangan menjadi faktor prima dengan tujuan mendapat kunci untuk proses

dekripsi. Algoritma ini digunakan untuk kepentingan autentikasi, yakni dengan kata lain data dan informasi benar-benar berasal dari sumber yang benar. Oleh karena itulah kunci pada algoritma ini berbeda saat enkripsi dan dekripsi. (Ananda et al., 2021)

Besaran yang digunakan pada algoritma RSA :

Tabel 2. 2 Besaran pada Algoritma RSA

p dan q bilangan prima	(rahasia)
n = p . q	(tidak rahasia)
Φ(r) = (p-1)(q-1)	(rahasia)
e (kunci enkripsi)	(tidak rahasia)
d (kunci dekripsi)	(rahasia)
m (*Plaintext*)	(rahasia)
c (*Ciphertext*)	(tidak rahasia)

Untuk menghasilkan pasangan kunci RSA, algoritma yang digunakan adalah sebagai berikut.

1. Dua bilangan prima besar, p dan q, dipilih. Nilai p dan q perlu dijaga kerahasiaannya.
2. Nilai N adalah hasil n = p x q.Besar nilai n tidak memerlukan kerahasiaan.
3. Nilai m dihitung sebagai (p – 1) kali (q – 1).
4. Sebuah kunci publik, disebut e, dipilih sebagai bilangan bulat yang relatif prima terhadap m.

   Artinya, e harus memiliki faktor pembagi terbesar yang sama dengan 1, yang secara matematis disebut gcd (e, d) = 1. Algoritma Euclid dapat digunakan untuk menemukannya.
5. Kunci privat, disebut d, dihitung sedemikian rupa sehingga (d x e) mod m = 1. Algoritma *Extended Euclid* dapat digunakan untuk menemukan nilai d yang sesuai.

   Oleh karena itu, hasil dari algoritma tersebut menghasilkan:

   - Kunci publik merupakan pasangan (e, n).
   - Kunci privat merupakan pasangan (e, d).

2.9.4 Penelitian Terdahulu

Penelitian sebelumnya dilakukan dengan maksud untuk memperoleh materi perbandingan dan referensi. Selain itu, guna mencegah terjadinya kesan kesamaan dengan penelitian ini, maka dalam tinjauan pustaka ini, peneliti memasukkan hasil-hasil penelitian terdahulu sebagaimana yang disajikan berikut.

Tabel 2. 3 Penelitian Terdahulu Terkait dengan Keamanan Enkripsi

Judul, Tahun	Nama Peneliti,Tahun	Metode	Atribut	Hasil Penelitian	Kelebihan	Kekurangan
Analisis Sistem Keamanan Sistem Operasi (Windows, Linux, MacOS)	(Dede Fuji Abdul1, Moh. Ihsan Budiman, Tedi Kurniawan, 2019)	Studi Pustaka	Windows, Linux, MacOS.	fitur keamanan yang disediakan oleh sistem operasi Windows, Linux dan Macintosh memiliki keunggulan dan kegunaanya masing-masing serta dengan cara kerja yang beragam di setiap sistem operasi	Pembahasan keamanan system operasi Meliputi 3 sistem operasi	• Pembahasan penelitian tidak secara spesifik berfokus pada satu sistem operasi, • Penerapan Algoritma pada pembahasan
Implementasi Pengamanan PGP pada Platform Zimbra Mail Server	(Dandy Pramana Hostiadi, Ida Bagus Suradarma,2017)	Skema alur PGP	Mail server, Web browser	Hasil analisa juga menunjukkan bahwa terdapat perbedaan size ukuran dari *file* attachment yang menggunakan pengamanan PGP, dimana size *file* menjadi lebih besar yang disebabkan adanya proses enkripsi dengan kunci *private*.	Hasil akhir data terdapat tabel uji perbandingan data, antara *file* asli, *file* penerima tanpa PGP dan dengan PGP	• Tidak mencantumkan Penjelasan Algoritma yang sedang di kerjakan

Penerapan Keamanan E-Mail Dengan Menggunakan *Gnu Privacy Guard* (Gnupg)	(Hari Purwanto,2020)	DES	Linux	Penerapan Email Enkripsi dengan menggunakan algoritma enkripsi el-gamal dan DES, dengan hasil akhir penyandian dan tanda tangan *file* enkripsi		
Gabungan Advanced Encryption Standard Dan Vigenere Cipher Untuk Pengamanan Dokumen Digital	(Eko Hari Rachmawanto, Christy Atika Sari,2020)	algoritma Vigenere Cipher & AES	Kasiski Test	Penggabungan antara 2 algoritma. data menjadi semakin aman *file* dienkripsi dan iderkipsi dengan baik	Dengan menggabungkan kunci dari algoritma Vigenere dan AES, program dapat memberikan tingkat keamanan data yang tinggi. Kombinasi ini memastikan bahwa tidak ada masalah keamanan yang timbul, dan data tetap aman.	Waktu Yang Lama Pada Proses Enkripsi dan Dekripsi menggunakan kombinasi Algoritma Vigenere dan AES, Terutama pada dokumen Power Point

Enkripsi dan Dekripsi Pesan Menggunakan Metode Vigenere Cipher dan Route Cipher	(Zulfatul Aufia, Turmudi, Evawati Alisah,2021)	Vigenere Cipher dan Route Cipher		• Penerapan 2 Algoritma Enkripsi • Penjelasan Proses Enkripsi menggunakan algoritma Vigenere dan Route Cipher	• Penjelasan Secara Detail Alur Proses Enkripsi Menggunakan Metode	• Tidak Melakukan Penerapan Secara Langsung Pada System Operasi
Analisis Sistem Keamanan Pada Sistem Operasi Microsoft Windows, Linux Dan Macintosh	(Aceng Abdul Wahid Program,2019)	Studi Pustaka	Bitlocker, Firewall, SSH	fitur keamanan yang disediakan oleh sistem operasi Windows, Linux dan Macintosh memiliki keunggulan dan kegunaanya masing-masing	Pembahasan keamanan system operasi Meliputi 3 sistem operasi	• Pembahasan penelitian tidak secara spesifik berfokus pada satu sistem operasi, • Penerapan Algoritma pada pembahasan
Pelatihan Penginstalan Dan Penggunaan Os Linux Yang Berbiaya Murah Dan Legal Bagi Guru Sma Di Kabupaten Ogan Ilir	(Firmansyah Burlian, Irsyadi Yani, Ismail Thamrin, Muhammad Yanis, Yulia Resti,2022)	-	-	-	-	-
Penerapan Algoritma Kriptografi Kunci Simetris Dengan Modifikasi Vigenere	(Arrijal, Rusdi Efendi, Boko Susilo,2016)	Vigenere Cipher	Java, object-oriented-programming(OOP)	menghasilkan suatu proto-type aplikasi yang menerapkan algoritma kriptografi kunci	Metode vigenere cipher dengan algoritma kriptografi kunci simetris yang	

Cipher Dalam Aplikasi Kriptografi Teks				simetris. mengenkripsi setiap subsequence yang dipartisi dari plain text sesuai metode Vigenere Cipher dengan algoritma kriptografi kunci simetris yang dapat saling berbeda, di mana jumlah subsequence yang dihasilkan adalah sebanyak algoritma kriptografi kunci simetris yang digunakan.	dapat saling berbeda, di mana jumlah subsequence yang dihasilkan adalah sebanyak algoritma kriptografi kunci simetris yang digunakan.	
Keamanan Email Menggunakan Metode *Pretty Good Privacy* Dengan Algoritma Rsa	(Ridwan Ighfirlana Ananda, Fauziah, Nur Hayati,2021)	RSA	PGP	8 data email yang dikirim ke beberapa user menunjukan bahwa sistem PGP cukup baik untuk teknik enkripsi.menghasilkan size *file* yang berbeda dari *file* asli jika di bandingkan. Misalnya		Size pada *file* dokumen yang telah dienkripsi memiliki perubahan dari sebelumnya menjadi lebih besar

				pada data dengan format .TXT menunjukan dari size *file* asli 7,402 Bytes dan size *file* yang telah di enkripsi menjadi 7,777 Bytes.		
Penerapan Sistem *File Encryption* Pada Sistem Operasi Linux Untuk Menjaga Kerahasiaan Data Pengguna	(Irfan Adigunanto, 2024)	RSA	Studi Pustaka, GnuPG,*Seahorse*	Penerapan *File* enkripsi menggunakan algoritma RSA, *file* local terenkripsi, *File* terenkripsi dengan baik Penggunaan gnupg dapat diterapkan melalui aplikasi GUI (Graphical User Interface) melalui aplikasi *Seahorse*	Penggunaan aplikasi GnuPG dan *Seahorse* tidak berbayar melainkan *Free (Open Source)* GnuPG pada Linux versi 20.x sudah menjadi satu dengan Instalasi Linux Ubuntu	