Perpaduan Diffie Hellman dan Blowfish sebagai Sistem Keamanan Dokumen

Muhammad Rizka*

Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe

*rizka@pnl.ac.id (penulis korespondensi)

Abstrak — Keamanan data telah menjadi ancaman yang paling sensitive dan sering terjadi ditengah meningkatnya trafik data yang cukup massif dalam jaringan internet. Untuk menjaga keamanan data ada berbagai metode yang telah di usulkan diataranya dengan kriptografi. Kriptografi merupakan teknik matematika untuk menyandikan informasi *(plaintext)* ke dalam bentuk informasi yang tidak bisa di baca atau dipahami *(ciphertext)* yang disebut dengan proses enkripsi. Kriptografi dengan kunci simetri dalam penerapannya menggunakan kunci yang sama untuk proses enkripsi dan dekripsi data sehingga waktu yang dibutuhkan dalam pengoperasiannya relative rendah berbeda halnya dengan kunci asimetri. Algoritma *Blowfish* merupakan salah satu algoritma kriptografi simetri dengan *block cipher* 64bit dan memiliki ukuran kunci yang bervariasi mulai dari 32bit sampai dengan 448bit. Algoritma blowfish sangat sesuai diimplementasikan pada aplikasi yang berbasis desktop (*stand alone*) namun demikian algoritma blowfish memiliki kekurangan saat di implementasikan pada aplikasi yang berbasis jaringan, hal ini disebabkan karena proses pendistribusian kunci yang tidak aman. Proses pendistribusian kunci pada algoritma blowfish memiliki potensi terhadap ancaman serangan MITM (*Man in The Middle Attack*) di dalam jaringan. Potensi tersebut dapat dicegah dengan melibatkan metode Diffie hellman saat proses pembangkitan kunci simetri. Metode Diffie hellman melibatkan dua atau lebih entitas yang saling berkomunikasi dengan bertukar angka dan melakukan perhitungan sederhana untuk mendapatkan angka yang sama dan nantinya akan digunakan sebagai kunci simetri. Kunci simetri yang dibangkitkan dengan metode Diffie hellman digunakan untuk proses enkripsi dan dekripsi dengan algoritma blowfish.

Kata kunci— Algoritma Blowfish, Algoritma Diffie Hellman, Kunci Simetri.

Abstract *— Data security has become the most sensitive threat and often occurs amid the massive increase in data traffic on the internet network. To maintain data security, there are various methods that have been proposed, including cryptography. Cryptography is a mathematical technique for encoding information (plaintext) into a form of information that cannot be read or understood (ciphertext) which is called the encryption process. Cryptography with a symmetric key in its application uses the same key for the encryption and decryption of data so that the time required for operation is relatively low, unlike the asymmetric key. The Blowfish algorithm is a symmetric cryptographic algorithm with a 64-bit block cipher and has a key size that varies from 32-bit to 448-bit. The blowfish algorithm is very suitable to be implemented in desktop-based applications (stand alone), however the blowfish algorithm has drawbacks when implemented in network-based applications, this is due to the insecure key distribution process. The key distribution process in the blowfish algorithm has the potential to attack the threat of MITM (Man in The Middle Attack) in the network. This potential can be prevented by involving the Diffie hellman method during the symmetric key generation process. The Diffie hellman method involves two or more entities communicating with each other by exchanging numbers and performing simple calculations to get the same number which will later be used as a symmetry key. The symmetric key generated by the Diffie hellman method is used for the encryption and decryption process using the blowfish algorithm*.

Keywords — Blowfish Algorithm, Diffie Hellman Algorithm, Simetric Key.

I. PENDAHULUAN

Kriptografi merupakan ilmu dan seni untuk menjaga kerahasiaan informasi. Kriptografi mencegah terhadap ancaman pencurian, perusakan dan penyalahgunaan data yang bersifat rahasia[1]. Dalam penerapannya kriptografi dibagi kedalam dua kelompok utama yaitu kriptografi simetris dan kriptografi asimetris. Kriptografi simetris yaitu kriptografi yang menggunakan kunci yang sama untuk proses enkripsi dan dekripsi. Kriptografi asimetris menggunakan kunci yang berbeda (kunci private publik dan kunci private) untuk melakukan proses enkripsi dan dekripsi. Kriptografi simetris dan kriptografi asimetris memiliki kelebihan masing-masing dan biasanya diaplikasikan pada kasus yang sesuai kebutuhan dan kondisi[2]. Ada beberapa algoritma kriptografi simetri yang memiliki performa tinggi yaitu blowfish, AES (Advance Encryption Standart) dan DES (Data Encryption Standart).

Blowfish merupakan algoritma kriptografi simetri yang berarti penggunaan kunci yang sama untuk proses enkripsi dan dekripsi. Algoritma blowfish beroperasi dengan input dan output blok data (block cipher) yang berukuran 64bit [3]. Untuk ukuran kunci blowfish bervariasi mulai dari 32bit sampai dengan 448bit dan algoritma memiliki kunci yang tidak sering berubah-ubah. Blowfish dapat digunakan untuk melakukan proses enkripsi dan dekripsi data text, gambar, audio dan lain-lain. Proses enkripsi dan dekripsi pada algoritma blowfish terdiri dari 16 kali iterasi dan setiap iterasi terdiri dari permutasi dan subtitusi antara bagian kunci dan inputan data [4].

Untuk menjaga kerahasiaan kunci simetri dari ancaman serangan MITM (*Man in The Midle Attack*) maka proses pertukaran kunci perlu didesain dengan aman yaitu dengan menambahkan algoritma *Diffie Hellman* dalam *key distribution*.

Diffie Hellmen merupakan algoritma pertukaran kunci yang memungkinkan dua pihak yang berkomunikasi melalui jaringan publik untuk dapat membangun kunci bersama yang bersifat rahasia. Kunci bersama tersebut dapat digunakan untuk melakukan enkripsi dan dekripsi dokumen dengan kriptografi simetris [5]. Mekanisme pertukaran kunci dengan Diffie hellman dapat dianalogikan ketika dua pihak yang berkomunikasi katakanlah alice dan bob masing-masing memiliki dokumen yang ingin dibagikan secara rahasia. Untuk melakukan hal tersebut mereka harus menyepakati sebuah informasi awal (public information) dan selanjutnya informasi tersebut dikombinasikan dengan informasi previledge masing-masing serta dikirimkan melewati jaringan yang tidak aman. Alice dan bob masing-masing menerima informasi yang telah dikirimkan tadi dan selanjutnya dikombinasikan lagi dengan informasi rahasia masing-masing. Hasil akhirnya berupa informasi bersama yang bersifat rahasia (*common secret key*). *Common secret key* berfungsi untuk melakukan enkripsi dan dekripsi dokumen sebelum dikirimkan melalui jaringan yang tidak aman.

Dokumen merupakan media tertulis dan dapat dicetak yang dipergunakan sebagai bukti dan keterangan [6]. Pengamanan dokumen saat ini menjadi hal yang perlu diperhatikan agar tidak dimanipulasi, diduplikasi secara tidak sah, dan disalahgunakan oleh pihak-pihak yang tidak bertanggung jawab. Pengamanan dokumen dapat dilakukan melalui berbagai cara, salah satunya adalah dengan menggunakan prinsip-prinsip kriptografi. Kriptoggrafi dapat memproteksi dokumen dari berbagai model manipulasi dan penyalahgunaan yang tidak sah.

Dalam Penelitian ini algoritma *Diffie Hellman* digunakan untuk pertukaran kunci (*key exchange*) diantara user dalam jaringan publik dan algoritma *Blowfish* digunakan untuk proses enkripsi dan proses dekripsi dokumen. Sistem kerja aplikasi dimulai dari proses pembangkitan kunci dengan algoritma Diffie Hellman yang menghasilkan kunci bersama (*common secret key*). Kunci bersama tersebut yang menjadi inputan dalam proses enkripsi dan dekripsi dokumen di sisi *end user*. Cipher blok hasil dari proses enkripsi akan dikirimkan ke tujuan melalui jaringan public.

II. METODOLOGI PENELITIAN

A. Perancangan Sistem Keamanan

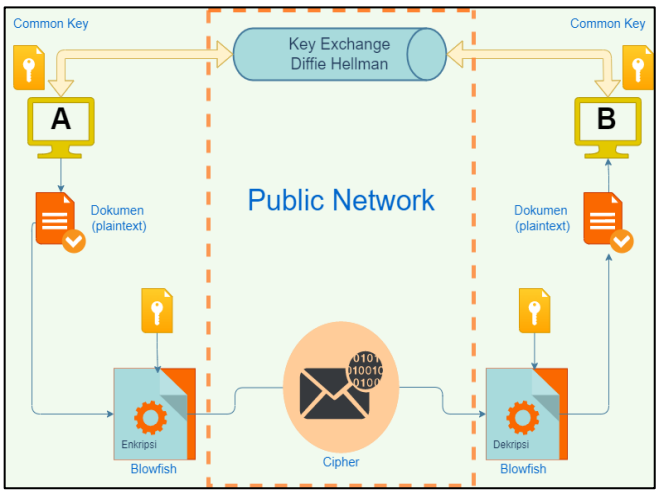
Pada tahap ini merupakan pembuatan sistem keamanan dokumen yang terdiri dari proses pembangkitan kunci simetri dengan algoritma Diffie Hellman, proses enkripsi dan dekripsi dengan algoritma blowfish dengan tipe operasi blok cipher ECB (*Electronic Code Book*) dan padding dengan pkcs5padding. Dokumen yang telah terenkripsi dalam bentuk cipher akan dikirimkan ke tujuan melalui protocol java socket. Untuk lebih jelasnya alur data dari sistem keamanan ditunjukkan pada Gambar 1. Desain Sistem berikut ini.

Gambar 1. Desain Sistem

Pada Gambar 1. Desain Sistem komunikasi diantara Alice dan Bob. yang diawali dengan Key Exchange (pembangkitan kunci bersama) dengan algoritma Diffie Hellman.

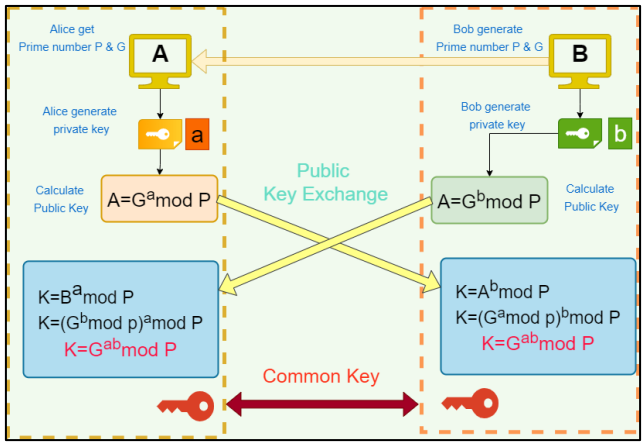
B. *Diffie Hellman Key Exchange*

Pembangkitan kunci bersama dimulai dari device Bob yang membentuk dua angka P dan G (bilangan prima) dan selanjutnya dikirimkan ke Alice. Bilangan P, G dan komponen Diffie Hellman lainnya dibentuk dengan tipe Biginteger supaya dapat mendukung kunci bersama (*common key*) dengan ukuran 128bit. Alice membentuk kunci private a dan Bob membentuk kunci private b dan Selanjutnya masing-masing melakukan perhitungan dengan rumus $A=G^a \bmod P$ dan $B= G^b \bmod P$. Hasil perhitungan tersebut dipertukarkan melalui jaringan public dan masing-masing Alice dan Bob melakukan perhitungan untuk mendapatkan kunci rahasia dengan rumus $K=B^a \bmod p$ dan $K=A^b \bmod p$. Mekanisme pembangkitan kunci Diffie Hellman dapat dilihat pada Gambar 2. Diffie Hellman *Key Exchange*.



Gambar 2. *Diffie Hellman Key Exchange*

Dalam Penelitian ini pembangkitan kunci dengan algoritma Diffie hellman akan menghasilkan kunci simetri dengan Panjang 128 bit. Kunci yang telah dihasilkan selanjutnya dikonversikan dalam format kunci blowfish.



C. Algoritma Blowfish

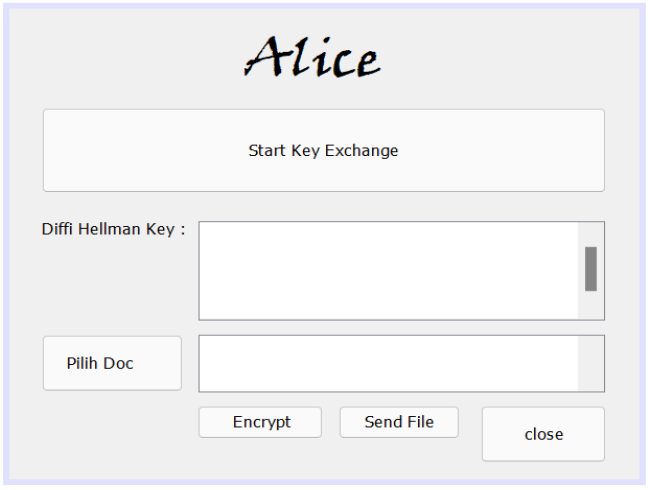
Algoritma Blowfish memiliki Panjang kunci yang bervariasi mulai dari 32bit sampai 448bit dan dalam Penelitian

ini panjang kunci yang di pilih yaitu 128bit. Panjang kunci yang dipilih akan mempengaruhi tingkat keamanan dan konsumsi daya komputasi dari perangkat yang digunakan. Blok cipher dalam algoritma blowfish memiliki ukuran 64bit sehingga setiap dokumen yang dimasukkan kedalam sistem keaman ini akan dipecah-pecah menjadi 64bit. Mode operasi yang digunakan yaitu *Electronic Code Book* (ECB) yang berfungsi untuk proses pemecahan suatu documen menjadi blok cipher dengan ukuran 64bit. Proses pemecahan documen juga melibatkan padding (pkcs5padding) pada blok data yang berukuran kurang dari 64bit. Padding berfungsi sebagai tambahan data agar sebuah blok data yang dienkripis tetap berukuran 64bit.

III.HASIL DAN PEMBAHASAN

A. Tampilan User Interface Alice

Tampilan UI untuk Alice terdiri dari tombol *start key*, Pilih Doc, *Encrypt*, *Send File* dan *Close*. Tombol *Key exchange* berfungsi untuk membuka jalur koneksi dan menunggu permintaan koneksi dari Bob. Saat koneksi sudah terbentuk maka alice akan menerima nilai P dan G dari Bob. Selanjutnya alice dan bob akan saling melakukan *key exchange* hingga kunci bersama terbentuk serta tampil pada layar aplikasi. Tombol Pilih Doc berfungsi untuk menyeleksi documen yang akan dienkripsi dengan algoritma blowfish. Untuk lebih jelasnya dapat dilihat pada Gambar 3. *User Interface* Alice

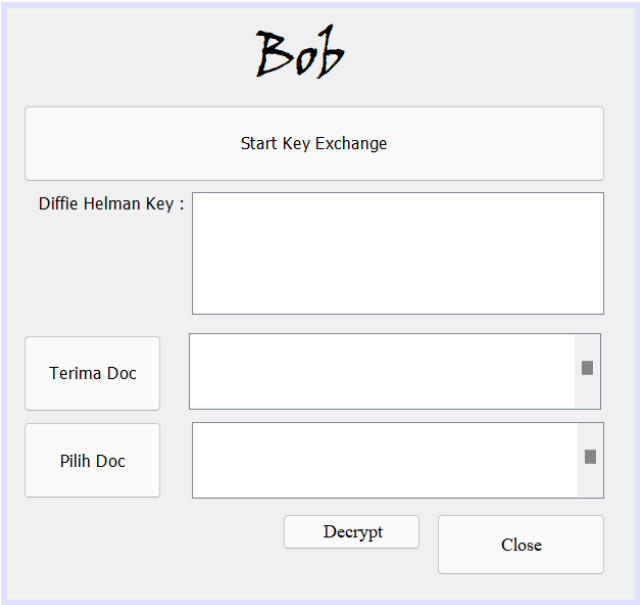


Gambar 3. *User Interface* Alice

Pada Gambar 3. User Interface Alice berisi tombol Encrypt yang berfungsi untuk melaksanakan proses enkripi Document dengan algoritma blowfish.

B. Tampilan User Interface Bob

Tampilan UI untuk Bob terdiri dari tombol start key exchange, Decrypt, Terima Doc, Pilih Doc dan Close. Tombol *Key exchange* berfungsi untuk menghubungi atau meminta koneksi ke Alice. Saat koneksi sudah terbentuk maka Bob akan menggenerate nilai P dan G dan selanjutnya mengirimkannya ke Alice. Alice dan Bob saling melakukan key exchange hingga kunci bersama terbentuk serta tampil pada layar aplikasi. Tombol Pilih Doc berfungsi untuk menyeleksi documen yang akan di dekripsi dengan algoritma blowfish. Untuk lebih jelahnya dapat dilihat pada Gambar 4. *User Interface* Bob.



Gambar 4. User Interface Bob

Pada Gambar 4. *User Interface* Bob berisi tombol Decypt yang berfungsi untuk melakukan dekripsi cipher yang dikirimkan Alice menjadi document yang dapat di baca (plaintext). Dokumen yang sudah berhasil di dekripsi dapat disimpan pada folder yang di inginkan.

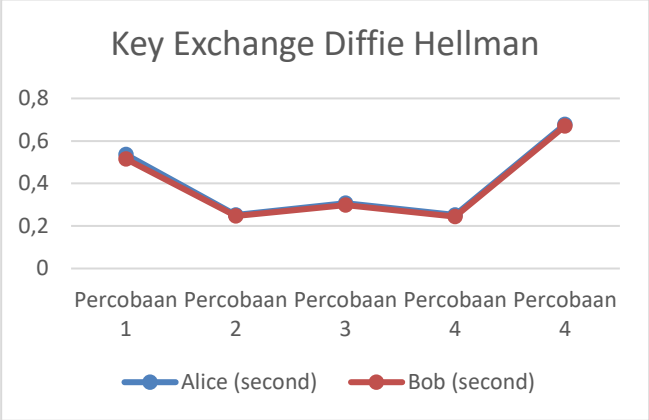
C. Pengujian Key Exchange dengan Diffie Hellman

Dalam Penelitian ini *Key Exchange* dengan Diffie hellman memanfaatkan class networking java untuk membangun koneksi data diantara alice dan bob. Class networking java yang digunakan yaitu serversocket dan socket. Dengan alamat ip localhost (127.0.0.1) dan port 1212. Percobaan yang dilakukan sebanyak lima kali key exchange diantara alice dan bob. Hasil percobaan dapat dilihat pada tabel 1. Kinerja Key Exchange Diffie Hellman.

Tabel 1. Pengujian Key Exchange Diffie Hellman

No.	Pengujian	Alice (detik)	Bob (detik)
1.	Percobaan 1	0.536	0.515
2.	Percobaan 2	0.252	0.247
3.	Percobaan 3	0.307	0.298
4.	Percobaan 4	0.252	0.244
5.	Percobaan 4	0.678	0.67
Rata-rata		0.405	0.395

Dari Tabel 1. Dapat dilihat bahwa perbedaan waktu proses *Key Exchange* 128 bit antara Alice dan Bob tidak terlalu jauh. Rata-rata waktu yang dibutuhkan alice dari lima percobaan yang dilakukan adalah 0.405 detik sedangkan Bob dengan waktu 0.395 detik. Untuk melihat lebih rinci perbedaan waktu komputasi Diffie Hellman antara alice dan bob dapat dilihat pada Gambar 5. Kinerja *Key Exchange Diffie Hellman*.



Gambar 5. Kinerja Key Exchange Diffie Hellman

D. Pengujian *Blowfish Encryption*

Pengujian dilakukan dilakukan terhadap proses enkripsi dan dekripsi dokumen. Pengujian proses enkripsi dan dekripsi dilakukan sebanyak lima kali dengan ukuran dokumen yang berbeda. Dokumnen yang digunakan dalam 89 enelitian ini berukuruan mulai dari 2.192 MB sampai 10.832MB. Hasil pengujian enkripsi dokumen dirangkum dalam Tabel 2. Pengujian Waktu Enkripsi.

Tabel 2. Pengujian Waktu Enkripsi

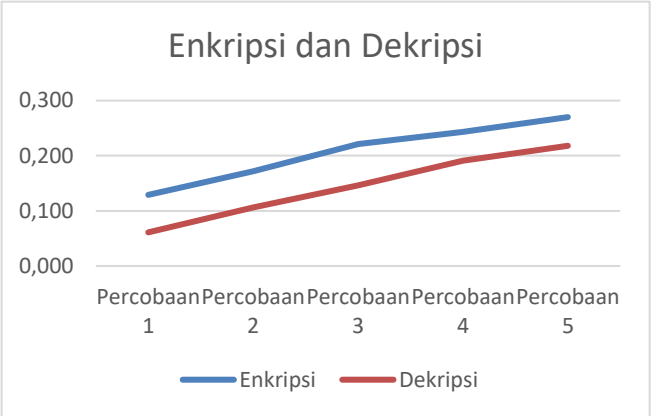
No.	Nama Dokumen *Plaintext*	Ukuran *Plaintext* (MB)	Enkripsi (detik)	Penambahan *byte* (*Ciphertext*)
1	Percobaan 1	2.192	0.129	138
2	Percobaan 2	4.182	0.172	434
3	Percobaan 3	6.146	0.221	984
4	Percobaan 4	8.479	0.243	775
5	Percobaan 5	10.832	0.270	422

Proses enkripsi akan menghasilkan cipher yang tidak dapat dibaca atau dikenali. Cipher selanjutnya akan didekripsi supaya dapat dibaca Kembali. Berikut ini Tabel 3. Pengujian Dekripsi Dokumen.

Tabel 3. Pengujian Dekripsi Dokumen

No.	Nama Dokumen *Plaintext*	Ukuran *Cipher* (MB)	Dekripsi (detik)
1.	Percobaan 1	2.192	0,061
2.	Percobaan 2	4.182	0,106
3.	Percobaan 3	6.147	0,146
4.	Percobaan 4	8.479	0,191
5.	Percobaan 4	10.833	0,218

Berikut ini merupakan grafik yang menunjukkan pengujian waktu proses enkripsi dan dekripsi pada Gambar 6. Kinerja Enkripsi dan Dekripsi.



Gambar 6. Kinerja Enkripsi dan Dekripsi

IV. KESIMPULAN

Berdasarkan hasil dan pembahasan dapat disimpulkan bahwa:

Penggunaan Diffie Hellman dalam pengiriman kunci (*Key Exchange*) berhasil dilakukan dengan waktu rata-rata 0.405 detik (alice) dan 0.395 detik (Bob) dari lima kali percobaan yang dilakukan. Perbedaan waktu yang terjadi dikarenakan ada delay saat pengiriman data lewat jaringan.

Enkripsi dan dekripsi dengan algoritma blowfish juga berhasil diimplementasikan. Waktu Enkripsi dengan dokumen paling kecil (2.192MB) membutuhkan waktu 0.129 detik serta tambahan 138byte pada ciphertextnya. Waktu Enkripsi dokumen paling besar (10.832MB) membutuhkan waktu 0.270 detik serta tambahan 422byte pada ciphertextnya. Penambahan ukuran dokumen paling besar terjadi pada dokumen percobaan 3 dengan penambahan byte sebesar 984byte. Saat dibandingkan dengan proses dekripsi waktu yang dibutuhkan lebih rendah yaitu sekitar 33%. Waktu dekripsi dengan dokumen paling kecil (2.192MB) membutuhkan waktu 0.061 detik sedangkan dokumen paling besar (10.832MB) membutuhkan waktu 0.218 detik.

REFERENSI

[1] Sitinjak, dkk. 2010. Aplikasi Kriptografi Menggunakan Algoritma Blowfish. Yogyakarta: Jurnal Seminar Nasional Informatika 2010 (SemnasIF 2010) UPN "Veteran" Yogyakarta 22 mei 2010: ISSN: 1979-2328

[2] Arrijal, Irham Mu'alimin dkk. 2016. Penerapan Algoritma Kriptografi Kunci Simetris Dengan Modifikasi Vigenere Cipher Dalam Aplikasi Kriptografi Teks. Bengkulu: Jurnal Pseudocode 2016: ISSN: 2355–5920

[3] Abdullah, Dedy dkk. Implementasi Algoritma Blowfish Dan Metode Least Significant Bit Insertion Pada Video Mp4. Bengkulu: Jurnal Pseudocode 2016: ISSN: 2355-5920

[4] Wardoyo, Siswo dkk. Enkripsi Dan Dekripsi File Dengan Algoritma Blowfish Pada Perangkat Mobile Berbasis Android. Banten: Jurnal Nasional Teknik Elektro 2016: ISSN: 2302 - 2949

[5] Security Encyclopedia (n.d). Diffie–Hellman (DH) Algorithm. Di akses pada 22 Maret 2022, dari https://www.hypr.com/diffie-hellman-algorithim.

[6] PrimaDoc (n.d). 3 Definisi Dokumen Menurut Para Ahli. Di akses pada 23 Maret 2022, dari https://primadoc.id/3-definisi-dokumen-menurut-para-ahli/

[7] Munir. Algoritma PertukaranKunci Diffie Hellman. Teknik Informatika STEI – ITB: Bahan Kuliah IF4020 Kriptografi.

[8] Gunaawan. 2013. Penggunaan Algoritma Diffie Hellman dalam Melakukan Pertukaran Kunci. Bandung: Struktur Diskrit – Sem I Tahun 2013.
[9] Susanto. Pemograman Client/Server dengan Java Socket.
[10] Yalisa, dkk. 2018. Algoritma Elgama dengan Pertukaran Kunci Diffie Hellman pada Aplikasi Keamanan Citra Sidik Jari Berbasis Android. Buketrata: Jurnal Procceding Seminar Nasional Politeknik Negeri Lhokseumawe Vol.2 No.1 September 2018: ISSN 2598-3954.
[11] Slideshare. "Kriptografi Algoritma Deffie Hellam. URL: 23 November 2018 https://www.slideshare.net/KuliahKita/kriptografi-algoritma-diffie-hellman, 2018
[12] Putra, dkk. 2016. Sistem Informasi Manjemant Pengarsipan Dengan Menggunakan Algoritma Blowfish. Politeknik Negeri Malang: Volume 2, Edisi 2, Februari 2016.
[13] Pratama. 2013. Aplikasi Pengamanan Dokumen Office Dengan Algoritma Kriptografi Blowfish. Jurnal: Falkultas Ilmu Komputer Universitas Dian Nuswantoro 14 Februari 2013.
[14] Pratama. 2013. Aplikasi Pengamanan Dokumen Office Dengan Algoritma Kriptografi Blowfish. Jurnal: Falkultas Ilmu Komputer Universitas Dian Nuswantoro 14 Februari 2013.