Aplikasi Aritmetika Modulo dalam Metode *Diffie-Hellman Key Exchange*

Kaisar [1)]

1) Teknik Informatika Institut Teknologi Bandung, Bandung, email: kaisar@students.itb.ac.id

Abstract *– Makalah ini membahas mengenai aplikasi aritmetika modulo dalam metode Diffie-Hellman Key Exchange. Metode Diffe-Hellman Key Exchange merupakan metode enkripsi dengan algoritma kunci publik (public key distribution system). Metode Diffie Hellman Key Exchange ini merupakan konsep yang mendasari pembentukan metode enkripsi RSA yang juga menggunakan algoritma nirsimetri. Aritmetika modulo dalam metode ini digunakan oleh kedua belah pihak yang saling bertukar pesan untuk mendapatkan sebuah kunci rahasia yang hanya mungkin diketahui oleh kedua pihak tersebut. Metode Diffie-Hellman Key Exchange merupakan suatu metode enkripsi yang sangat brilian, karena selain konsepnya yang sederhana dan mudah diimplementasikan, metode ini juga memiliki tingkat keamanan yang relatif tinggi*

.

Kata Kunci*: Diffie-Hellman Key Exchange, Masalah Logaritma Diskrit, Bilangan Prima Sophie Germain, Algoritma Pohlig-Herman*

1. PENDAHULUAN

Di bidang matematika diskrit aritmetika modulo, yang juga dikenal sebagai aritmetika modular atau aritmetika jam, memiliki pengertian sebagai suatu sistem aritmetika untuk bilangan *integer* di mana bilangan-bilangan tersebut akan "kembali ke awal" ketika samapai pada nilai tertentu [1]. Aritmetika modulo pertama kali diperkenalkan oleh Carl Friedrich Gauss dalam bukunya *Disquisitiones Arithmeticiae* yang diterbitkan pada tahun 1801.

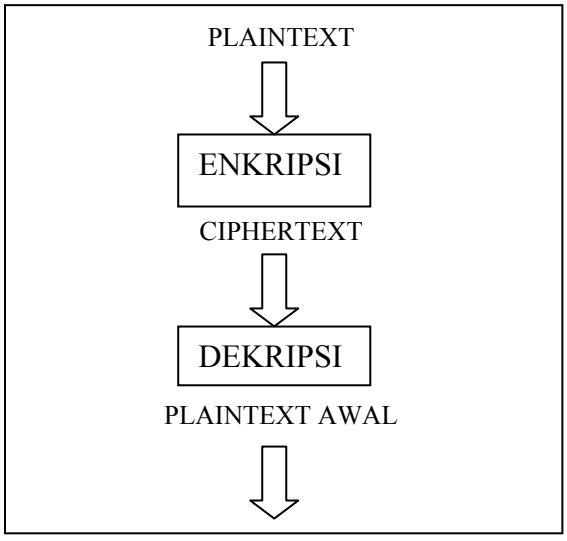
Contoh paling sederhana dari aritmetika modulo adalah sistem jam 24. Aritmetika di mana satu hari dibagi menjadi 24 jam dan diberi nilai 0 sampai dengan 23. Bila sekarang pukul 19.00, maka 8 jam kemudian akan menjadi pukul 03.00. Bila menggunakan aritmetika biasa maka seharusnya 19+8=27, yang mengindikasikan pukul 27.00. Namun karena bilangan jam "kembali ke awal" pada akhir hari, maka hal tersebut tidak berlaku. Artmetika jam ini dikenal juga dengan aritmetika modulo 24.

Operator yang digunakan pada aritmetika modulo adalah *mod*. Operator *mod*, jika digunakan pada pembagian bilangan bulat, akan memberikan sisa pembagian. Misalnya 23 dibagi 5 memberikan hasil = 4 dan sisa = 3, sehingga dapat ditulis 23 mod 5 = 3. [2]

Dalam berbagai bidang, aritmetika modulo memiliki peran dan aplikasi yang beraneka ragam. Salah satu aplikasi terpenting dari aritmetika modulo adalah kriptografi. Kriptografi adalah ilmu sekaligus seni untuk menjaga kerahasiaan pesan (data ataupun infomasi) dengan cara menyamarkannya (*to* crypt artinya menyamar) menjadi bentuk tersandi yang tidak mempunyai makna [3].

Pada zaman modern ini, kerahasiaan merupakan hal yang sudah menjadi tuntutan publik yang mutlak. Setiap individu pasti memiliki informasi-informasi pribadi yang ingin disembunyikan agar tidak diketahui oleh pihak-pihak yang tidak berhak. Di bidang komunikasi kerahasiaan dalam penyampaian pesan juga diinginkan oleh semua individu. Apabila seseorang ingin menyampaikan sesuatu pesan yang bersifat pribadi kepada orang lain yang dipercayainya, tentunya orang tersebut tidak menginginkan adanya pihak ke-tiga yang mendapat pesan tersebut. Oleh karena tuntutan-tuntutan seperti itulah ilmu kriptorafi ada dan berkembang.

Pada dasarnya, konsep kriptografi mengacu pada proses enkripsi dan dekripsi. Proses enkripsi adalah proses penyamaran dari *plaintext* (teks jelas yang dapat dimengerti) menjadi *ciphertext* (teks tersandi). Sedangkan dekripsi adalah proses pembalikan *ciphertext* menjadi *plaintext* asal[4].



Gambar 1.1. Enkripsi dan Dekripsi

Dalam Notasi Matematis, proses enkripsi dan dekripsi dapat dijabarkan menjadi.

$$E(P) = C \qquad (1.1)$$

Di mana fungsi enkripsi E memetakan *plaintext* P ke *ciphertext* C. Sedangkan pada proses kebalikannya

$$D(C) = P \qquad (1.2)$$

Dimana fungsi dekripsi D memetakan *ciphertext* C ke *plaintext* P. Dari kedua fungsi diatas berlaku juga fungsi komposisi

$$D(E(P)) = P \qquad (1.3)$$

Di dunia kriptografi ada berbagai algoritma yang beraneka ragam untuk menjalankan proses enkripsi dan dekripsi. Salah satunya adalah metode *Diffie-Hellman Key Exchange* yang merupakan salah satu kriptografi kunci publik yang menggunakan aritmetika modulo.

2. *DIFFIE-HELLMAN KEY EXCHANGE*

Diffie-Hellman (D-H) *Key Exchange* adalah protokol kriptografik yang memungkinkan kedua pihak yang bertukar informasi, walaupun mereka tidak mengenal satu sama lainnya, dapat secara bersama menciptakan sebuah kunci rahasia bersama melalui sebuah jalur komunikasi yang tak aman sekalipun. Kunci ini kemudian dapat digunakan untuk mengenkripsi pesang menggunakan kunci *cipher* simetris.[5]. *Diffie-Hellman Key Exchange* mempunyai beberapa sinonim. Di antaranya adalah:

- *Diffie-Hellman key agreement*
- *Diffie-Hellman key establishment*
- *Diffie-Hellman key negotiation*
- *Exponential key exchange*

2.1. Sejarah

DIffie-Hellman Key Exchange ditemukan pada tahun 1976 atas hasil kerjasama antara Whitfield Diffie dan Martin Hellman. Metode ini merupakan metude praktikal pertama untuk menciptakan sebuah rahasia bersama antara dua belah pihak melalui sebuah jalur komunikasi yang tidak terjaga.

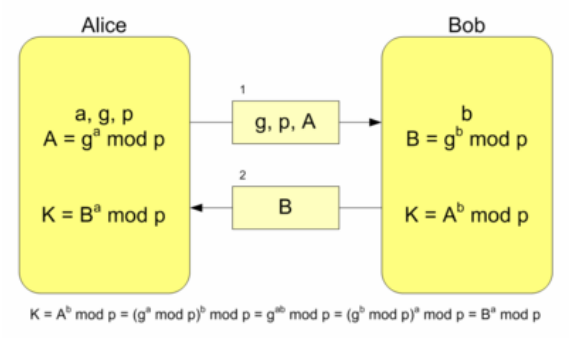
Metode ini amat dipengaruhi oleh karya-karya Ralph Merkle dalam bidang distribusi kunci publik. Sedangkan John Gill jug aturut berkontribusi dalam penciptakan metode ini dengan menyarankan pengaplikasian masalah logaritma diskrit.

Sebenarnya, metode yang sama telah lebih dulu diciptakan oleh Malcolm Williamson dari GCHQ di Inggris. Namun GCHQ memilih untuk menyembunyikannya dari publik sampai pada akhirnya mempublikasikannya pada tahun 1997. Kemudian pada tahun yang sama, metode ini dijadikan sebagai dasar oleh tiga orang dari MIT (*Massachusetts Institute of Technology) yaitu* Ron Rivest, Adi Shamir, dan Len Adleman untuk menciptakan algoritma RSA.

Metode *Diffie-Hillman Key Exchange* dipatenkan dengan U.S Patent no. 4,200,770 dengan mengkreditkan Whitfield Diffie, Martin Hellman, dan Ralph Merkle sebagai penemunya.

2.2. Deskripsi

Secara umum metode *Diffie-Hellman Key Exchange* dapat digambarkan oleh skema diagram berikut:



Gambar 2.1 Skema *Diffie-Hellman Key Exchange*

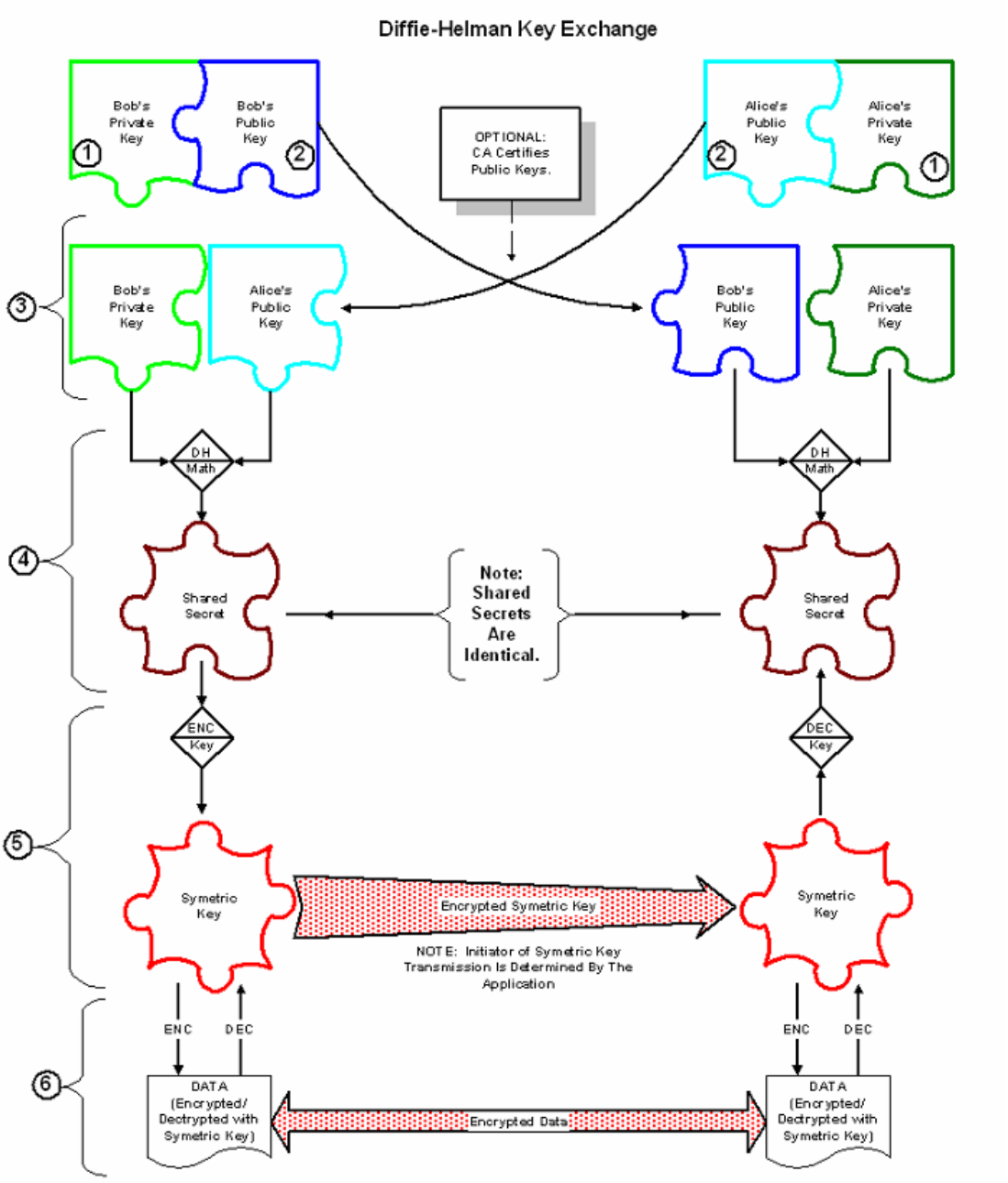
Berikut adalah penjelasan deskriptif mengenai diagram di atas:

1. Alice dan Bob sepakat untuk menggunakan bilangan prima p dan suatu bilangan g yang akan digunakasn sebagai basis.
2. Alice memilih suatu bilangan natural acak *a* dan mengirim (g^a mod p) ke Bob.
3. Bob memilih suatu bilangan natural acak *b* dan mengirim (g^b mod p) ke Alice.
4. Alice menghitung $(g^b \bmod p)^a$.
5. Bob menghitung $(g^a \bmod p)^b$.

Sekarang Alice dan Bob memiliki nilai dari $(g^a \bmod p)^b$dan $(g^b \bmod p)^a$ yang notabene memiliki nilai yang sama berdasarkan sifat asosiatif perpangkatan, yaitu:

$$(g^b)^a = (g^b)^a \qquad (2.1)$$

Nilai tersebut adalah nilai dari kunci yang dapat digunakan oleh Alice dan Bob untuk mengenkripsi dan mendekripsi pesan mereka. Perlu dicatat bahwa hanya nilai a, b, $(g^a \bmod p)^b$,dan,$(g^b \bmod p)^a$ yang perlu dirahasiakan.Tentu saja dengan menggunakan nilai a, b , dan p yang besar, maka akan didapatkan nilai kunci yang lebih sulit untuk dipecahkan oleh pihak ketiga



Gambar 2.2 Diagram Lengkap Proses *Diffie-Hellman Key Exchange* dan Proses Enkripsi serta Dekripsi Kunci Simetri

2.3. Tingkat Sekuritas

Protokol *Diffie-Hellman Key Exchange* dinilai aman dari pihak ke tiga yang bertindak sebagai pengamat yang tidak diinginkan (*eavesdropper*) apabila variabel g, a, b, dan p dipilih secara tepat. Bila angka-angka tersebut dipilih dengan benar, maka pengamat tersebut akan kesulitan untuk memecahkan apa yang dikenal dengan istilah *Diffie-Hellman Problem* untuk menemukan kunci publik yang diinginkan.

Kelemahan dari *Diffie-Hellman Key Exchange* akan terjadi apabila terpenuhinya syarat sehungga pengamat dapat menggunakan Algoritma Pohlig-Hellman (yang dapat diatasi dengan bilangan prima Sophie-Germain) dan adanya *Man in the Middle Attack.*

2.3.1. Algoritma Pohlig-Hellman dan Bilangan Prima yang Aman (Bilangan Prima Sophie-Germain)

Bila pangkat dari g (yaitu a dan b) kecil dan bukan merupakan bilangan prima yang aman (*safe prime*) maka para pengamat dapat menggunakan algoritma Pohlig-Hellman untuk mendapatkan nilai a dan b yang nantinya dapat dengan mudah digunakan untuk mendapatkan kunci publik.

Algoritma Pohlig-Hellman adalah sebuah algoritma yang digunakan untuk mengkomputasi logaritma diskritdalam sebuah kelompok multiplikatif yang pangkatnya bukan merupakan bilangan prima yang aman [6]. Algoritma ini didasari oleh teorema *Chinese Remainder* dan berjalan pada waktu polinomial.

Dengan contoh sederhana dimana semua elemen-elemen dari Z_p adalah ko-prima terhadap p, maka algoritma Pohlig Hellman dapat dijabarkan sebagai berikut:

Input Integer *p*, *g*, dan *e*.
Output Integer *x*, dimana $e \equiv g^x \pmod p$.

- Gunakan *Euler's Totient Function* untuk menentukan faktorisasi prima dari orde kelompok:

$$\varphi(p) = p_1 \cdot p_2 \cdot \ldots \cdot p_n \quad (2.2)$$

- Dari teorema sisa kita ketahui bahwa

$$x = a_1 p_1 + b_1 \qquad (2.3)$$

Kita *sekarang menemukan nilai dari* b_1 di mana persamaan berikut terpenuhi dengan menggunakan algoritma ceepat seperti algoritma *Baby-step giant*-step:

$$\begin{aligned} e^{\varphi(p)/p_1} &\equiv (g^x)^{\varphi(p)/p_1} \pmod p \\ &\equiv (g^{\varphi(p)})^{a_1} g^{b_1 \varphi(p)/p_1} \pmod p \\ &\equiv (g^{\varphi(p)/p_1})^{b_1} \pmod p \end{aligned}$$

(menggunakan teorema Euler)

(2.4)

Perlu dicatat $g^{\varphi(p)/p_1} \equiv 1 \pmod p$ maka orde dari g adalah kurang dari φ(*p*) dan $e^{\varphi(p)/p_1} \bmod p$ haruslah 1 agar terdapat solusi. Dalam kasus ini akan ada lebih dari solusi untuk x kecil dari φ(*p*), namun karena kita tidak mencari semua himpunan solusi maka kita dapat menganggap bahwa b_1=0. Operasi yang sama dilakukan untuk p_2 sampai dengan p_n.

Sebuah modifikasi kecil diperlukan di mana sebuah bilangan prima diulangi. Anggap kita menemui p_i untuk ke *k+1* kalinya. Makan kita telah mengetahui c_i dari persamaan

$$x = a_i p_i^{k+1} + b_i p_i^k + c_i \qquad (2.5)$$

dan kita akan menemukan nilai b_i seperti sebelumnya.

- Pada akhirnya kita akan mendapatkan bentuk-bentuk modulu kongruen yang cukup sehingga x dapat didapatkan dengan mengaplikasikan teorema *Chinese Remainder*.

Untuk mengatasi penggunaan algoritma Pohlig Herman, maka harus digunakan bilangan prima yang aman atau bilangan prima Sophie-Germain untuk menentukan nilai p.

Sebuah bilangan prima dikatakan aman apabila bilangan prima (anggap P) juga prima untuk 2P+1. Beberapa bilangan prima yang pertama adalah: 5, 7, 11, 23, 47, 59, 83, 107, 167, 179, 227, 263, 347, 359, 383, 467, 479, 503, 563, 587, 719, 839, 863, 887, 983, 1019, 1187, 1283, 1307, 1319, 1367, 1439, 1487, 1523, 1619, 1823, 1907.

Dengan 7 sebagai pengecualian, sebuah bilangan pima yang aman P mempunyai bentuk *6k-1* atau ekivalen dengan $p \equiv 5 \pmod 6$. Bentuk ini juga dikenal sebagai bilangan prima Sophie-Germain. Selain itu dengan 5 sebagai pengecualian, sebuah bilangan prima yang aman P mempunyai bentuk *4k-1* atau ekivalen dengan $p \equiv 3 \pmod 4$. Dengan memadukan kedua bentuk tersebut kita mendapatkan bahwa sebuah bilangan prima yang aman p>7 harus memenuhi pesamaan $p = 12k-1$ atau ekivalen dengan $p \equiv 11 \pmod{12}$.[7]

Bilangan-bilangan prima tersebut disebut “aman” karena hubungan mereka dengan bilangan prima kuat. Sebuah bilangan prima p adalah prima kuat apabila p+1 dan p-1 memiliki faktor prima besar. Selain itu apabila p merupakan bilangan prima yang aman kelompok multiplikatif dari angka-angka modulo p akan membunyai sub-kelompok bilangan prima berorde tinggi. Sub-kelompok inilah yang diharapkan. Selain itu alasan digunakannya bilangan yang aman p adalah agar modulusnya bernilai sekecil mungkin relatif terhadap p.

Terhitung sejak Januari 2007, bilangan prima yang aman terbesar adalah $48047305725 \cdot 2^{172404}-1$ yang mempunyai 51910 digit desimal. Bilangan prima yang aman ini, yang juga merupakan bilangan prima Sophie-Germain, ditemukan oleh David Underbakke pada tanggal 25 Januari 2007 menggunakan program TwinGen dan LLR. Rekor sebelumnya dipegang oleh Járai et.al. dengan nilai $137211941292195 \times 2^{171960}-1$, yang mempunyai 51780 digit desimal. [8]

2.3.2. *Man in the Middle Attack*

Metode *Diffie-Hellmann Key Exchange* rentan terhadap *man in the middle attack*. *Man in the middle attack* memiliki pengertian bahwa terdapat pihak ketiga yang mengintersepsi nilai (g^a mod p) milik Alice (merujuk pada contoh kasus di sub bab 2.2) yang dikirim ke Bob dan mengirim nilai (g^a mod p) miliknya sendiri ke Bob. Ketika Bob mengirim nilai (g^b mod p) pihak ketiga ini kembali mengintrsepsinya dan mengirimkan nilai (g^b mod p) miliknya sendiri ke Alice. Sehingga terdapat dua kunci publik yaitu kunci publik pihak ketiga-Alice dan pihak ketiga-Bob.

Setelah proses ini, Alice mengirim pesan, pihak ketiga dapat mendekripsi pesan tersebut, membaca ataupun bahkan mengubahnya, meng-enkripsinya kembali, dan mengirim pesan tersebut ke Bob. Hal ini juga berlaku sebaliknya. Kelemahan ini terjadi karena pada metode *Diffie-Hellman Key Exchange* biasa tidak mengautentifikasi kedua pihak yang berkomunikasi. Agar lebih aman proses autentifikasi perlu diimplementasikan dalam proses *Diffie-Hellman Key Exchange*.

Protokol *Diffie-Hellman Key Exchange* yang terautentifikasi atau *Station to Station Protocol* (STS) dikembangkan oleh Diffie, van Oorschot, dan Wiener pada tahun 1992 untuk mengebalkan metode *Diffie-Hellman Key Exchange* dari ancaman *man in the middle attack.* Kekebalan ini didapatkan dengan mengharuskan kedua belah pihak unuk mengatentifikasi diri mereka dengan penggunaan *digital signatures* dan *public key certificates* [9]. Selain itu, beberapa proses autentifikasi kriptografik juga dapa diimplementasikan ke proses *Diffie-Hellman Key Exchange.*

3. HASIL DAN PEMBAHASAN

Contoh kasus penggunaan metode *Diffie-Hellman Key Exchange* adalah sebagai berikut:

- Alice and Bob setuju untuk menggunakan bilangan prima yang aman p=23 and basis g=5.
- Alice memilih bilangan integer rahasia a=6, kemudian mengirim Bob nilai (g^a mod p)
    - $5^6 \bmod 23 = 8$.
- Bob memilih bilangan integer rahasia b=15, kemudian mengierim Alice nilai (g^b mod p)
    - $5^{15} \bmod 23 = 19$.
- Alice menghitung $(g^b \bmod p)^a \bmod p$
    - $19^6 \bmod 23 = 2$.
- Bob menghitung $(g^a \bmod p)^b \bmod p$
    - $8^{15} \bmod 23 = 2$.

Terlihat bahwa dari proses tersebut kedua pihak mendapatkan sebuah kunci publik yang sama, yaitu s = 2 yang dapat digunakan untuk mengenkripsi maupun mendekripsi pesan yang ingin disampaikan. Sekarang mari kita analisis kasus bila ada pihak-ketiga yang ingin mendapatkan kunci publik Alice dan Bob. Anggap pihak ketiga bernama Jupri.

Anggap s = kunci publik Alice dan Bob. s = 2
Anggap a = Bilangan integer rahasia Alice. a = 6
Anggap b = Bilangan integer rahasia Bob. b = 15
Anggap g = basis publik. g=5
Anggap p = bilangan prima publik. p = 23

Alice	
Tahu	Tidak Tahu
p=23	b = 15
g=5	
a=6	
$5^6 \bmod 23 = 8$	
$5^b \bmod 23 = 19$	
$19^6 \bmod 23 = 2 = s$	
$8^b \bmod 23 = 2 = s$	
$19^6 \bmod 23 = 8^b \bmod 23$	

Tabel 3.1 Hal-Hal yang diketahui pihak pertama

Bob	
Tahu	Tidak Tahu
b = 15	p=23
g=5	
a=6	
$5^{15} \bmod 23 = 19$	
$5^a \bmod 23 = 8$	
$19^a \bmod 23 = 2 = s$	
$8^{15} \bmod 23 = 2 = s$	
$19^a \bmod 23 = 8^{15} \bmod 23$	

Tabel 3.3 Hal-Hal yang diketahui pihak kedua

Jupri	
Tahu	Tidak Tahu
p=23	b = 15
g=5	a=6
	s=2
$5^a \bmod 23 = 8$	
$5^b \bmod 23 = 19$	
$19^a \bmod 23 = s$	
$8^b \bmod 23 = s$	
$19^a \bmod 23 = 8^b \bmod 23$	

Tabel 3.2 Hal-Hal yang diketahui pihak ketiga

Dari tabel-tabel di atas terlihat bahwa Jupri tidak hampir tidak mungkin untuk mengetahui nilai dari kunci publik s karena tidak mengetahui salah satu dari variabel a dan b. Perlu dicatat pula apabila bilangan p adalah bilangan prima yang aman dengan sedikitnya 300 digit desimal dan a serta b setidaknya memiliki 100 digit desimal, maka kunci publik yang dihasilkan tidak mungkin didapatkan oleh pihak ketiga walaupun ia telah menggunakan algoritma komputasi yang terbaik pada saat ini. Masalah ini dikenal sebagai Masalah Matematika Diskrit. Sedangkan untuk bilangan basis b tidak perlu besar dan pada prakteknya biasanya menggunakan bilang 2 atau 5.

4. KESIMPULAN

- Metode *Diffie-Hellman Key Exchange* merupakan metode kriptografi kunci publik simetris yang menggunakan konsep aritmetika modulo
- Kelemahan dari *Diffie-Hellman Key Exchange* akan terjadi apabila terpenuhinya syarat sehungga pengamat dapat menggunakan Algoritma Pohlig-Hellman (yang dapat diatasi dengan bilangan prima Sophie-Germain) dan adanya *Man in the Middle Attack* (yang dapat diatasi dengan proses autentifikasi)
- Metode *Diffie-Hellman Key Exchange* merupakan dasar dari beberapa algoritma kriptografi, salah satu di antaranya adalah RSA.
- Metode *Diffie-Hellman Key Exchange* banyak digunakan sampai saat ini karena konsepnya yang sederhana dan tingkat keamanannya yang cukup tinggi.

DAFTAR REFERENSI

[1] http://en.wikipedia.org/Modulo_arithmetic.htm, 19 Desember 2007

[2] Rinaldi Munir, "Matematika Diskrit", *Program Studi Teknik Elektro dan Informatika ITBl*, Edisi Keempat, 2006, hal.V-13.

[3] Op.Cit. Munir, hal.V-21.

[4] Ibid

[5] http://en.wikipedia.org/Diffie-Hellman_key_exchange.htm, 19 Desember 2007

[6] http://en.wikipedia.org/Pohlig-Hellman_algorithm.htm, 19 Desember 2007

[7] http://en.wikipedia.org/Safe_prime.htm, 19 Desember 2007

[8] http://en.wikipedia.org/Sophie-Germain_prime.htm, 19 Desember 2007

[9] http://www.rsa.com/rsalabs/node.asp?id=2248, 22 Desember 2007