PERANCANGAN DAN IMPLEMENTASI PERANGKAT LUNAK PROXY TUNNEL DENGAN ENKRIPSI KUNCI SIMETRIS-ASIMETRIS UNTUK MANIPULASI PAKET DATA

GAT

Sekolah Tinggi Manajemen Informatika dan Komputer Pontianak
Program Studi Sistem Informasi
Jln. Merdeka N0. 372 Pontianak, Kalimantan Barat
E-mail: **Gutsy0818@yahoo.com**

***Abstracts**: The authors conducted a study on wiretapping in computer networks because of it is very disturbing security and comfort in the surf. Therefore the writer create a proxy tunnel software with symmetric and asymmetric encryption as a security agent that encrypts data communications conducted. Forms of research that the author did was experimental data collection techniques onegroup posttest-pretest, the authors study variable is the security of data packets transmitted between website and browser. Design method used is the Extreme Programming with 4 ways to apply the practical development of XP, namely coding, testing, listening and designing. The result is a software design software called "Crypto Tunnel Proxy", according to the function of encryption, tunneling and proxy. The software engineering author uses Visual Basic 6.0 and uses 2 types of encryption, namely symmetrical and asymmetrical. The authors also apply the hash function as a fingerprint or fingerptint of keys that are sent. This application can improve security in computer networks. By encrypting all data packets between the browser and the website are symmetrical and asymmetrical. Tests conducted with several commonly used browsers.*
Keywords: RC4 and RSA Encryptiom, Tunneling, Proxy

1. PENDAHULUAN

Masalah keamanan dari penyadapan data mengakibatkan dikembangkannya enkripsi halaman website yang disebut SSL(*Secure Socket Layer*). SSL ini adalah enkripsi halaman website dengan metode kunci asimetris, yang mana sertifikat SSL yang diterima browser berisi *public key* untuk mengenkripsi halaman yang akan dikirim ke server website.

Untuk melindungi browser dari sertifikat SSL palsu, diterapkan 2 metode oleh developer browser, yaitu CA(*Certificate Authority*) dan OCSP(*Online Certificate Status Protokol*). CA (Certificate Authority) adalah sebuah entitas yang mengeluarkan sertifikat digital yang dapat digunakan oleh pihak-pihak lainnya. Sejauh ini CA memberikan keamanan yang tinggi terhadap browsing, kecuali user memaksa menggunakan sertifikat SSL yang *error*. Kekurangannya adalah pada browser hanya terdaftar CA sertifikat SSL website-website tertentu. OCSP adalah metode validasi sertifikat SSL secara online ke sebuah server penyedia layanan OCSP, berbeda dengan CA, OCSP ini bisa mendaftarkan lebih banyak *fingerprint* sertifikat valid. Fitur CA dan OCSP ada pada browser-browser internet terbaru, tetapi tidak ada dalam internet explorer 5 yang menjadi browser default pada windows XP serta browser-browser lama lainnya, sehingga browser-browser versi lama sangat *vulnerable* terhadap *certificate spoofing*.

Solusi diusulkan sebuah proxy tunnel dengan enkripsi kunci simetris dan asimetris sebagai solusi alternatif, yang mana untuk penerapannya dibutuhkan server di titik/node di luar jaringan yang berperan sebagai tunnel server, server bisa berada sebelum gateway maupun sesudah gateway (server di internet).

Pada model OSI, perangkat lunak proxy tunnel ini bekerja pada 3 Layer yaitu *layer session*, *layer transport* dan *layer presentation.* Pada *layer session*, proxy tunnel client menciptakan sebuah sesi koneksi *full duplex* ke proxy tunnel server, karena tidak memungkinkan digunakan komunikasi *half duplex* sebab seringkali proxy tunnel client berada di belakang gateway atau firewall. Pada layer transport, proxy tunnel memanipulasi port yang digunakan, hal ini merupakan kemampuan dasar dari sebuah tunnel yang dapat menggunakan celah pada firewall jaringan untuk mendapatkan akses yang lebih luas. Pada *layer presentation,* proxy tunnel memiliki mesin enkripsi sendiri untuk mengenkripsi paket data yang masuk. Enkripsi yang digunakan adalah Algoritma Simetris RC4 dan Asimetris RSA. Adapun permasalahannya adalah bagaimana menghasilkan sebuah perangkat lunak proxy tunnel dengan emkripsi kunci simetris dan asimetris untuk manipulasi paket data. Tujuan dari penelitian ini untuk mengamankan komunikasi data dalam jaringan komputer yang rawan aktifitas hacking.

2. TINJAUAN PUSTAKA

2.1 Kriptografi

Kriptografi berasal dari bahasa yunani yaitu: kripto dan graphia, kripto berarti secret (rahasia) dan graphia berarti writing (tulisan). Menurut Ariyus (2008:13), kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ke tempat lain.

Algoritma kriptografi dibagi menjadi tiga bagian berdasarkan kunci yang dipakainya (Ariyus, 2008:44), yaitu: a) Algoritma Simetri, disebut algoritma klasik karena memakai kunci yang sama untuk kegiatan deskripsi dan enkripsi. Algoritma ini sudah ada sejak 4000 tahun yang lalu. Bila mengirim pesan dengan algoritma ini, penerima pesan harus diberitahu kunci dari pesan agar bisa mendeskripsikan pesan yang dikirim. Keamanan pesan yang menggunakan algoritma ini tergantung pada kunci. Jika kunci itu diketahui orang lain maka orang tersebut akan dapat melakukan enkripsi dan deskripsi terhadap pesan. Algoritma yang memakai enkripsi ini adalah Data Encryption Standard (DES), RC2, RC4, RC5, RC6, International Data Encryption Algoritm (IDEA), Advance Encryption Standard, One Time PAD, Blowfish, A5, dan sebagainya; b) Algoritma Asimetri, disebut dengan algoritma kunci publik, dengan arti kata kunci digunakan untuk enkripsi dan deskripsi berbeda. Pada algoritma asimetri kunci terbagi dalam dua bagian yaitu Kunci Umum (public key): Kunci yang boleh semua orang tahu (dipubliksikan) dan Kunci Rahasia (private key) : Kunci yang dirahasiakan (hanya boleh diketahui satu orang).

Algoritma asimetri bisa mengirimkan pesan lebih aman dari pada algoritma simetri. Algoritma yang menggunakan kunci publik di antaranya adalah: Digital Signature Algoritm, RSA, Diffie-Hellman(DH), Elliptic Curve Criptography(ECC), Kriptografi Quantum, dan lain sebagainya.

2.2 Enkripsi RC4 dan RSA

Menurut Deris Stiawan (2005:72), "RC4 merupakan salah satu algoritma kunci simetris yang berbentuk stream cipher". Algoritma RC4 merupakan salah satu jenis stream cipher, yaitu memproses unit atau input data pada satu saat. Dengan cara ini

enkripsi atau dekripsi dapat dilaksanakan pada panjang yang variabel. Algoritma ini tidak harus menunggu sejumlah input data tertentu sebelum diproses, atau menambahkan byte tambahan untuk mengenkripsi.

Algoritma RC4 memiliki dua fase, *setup* kunci dan pengenkripsian. Setup untuk kunci adalah fase pertama dan yang paling sulit dalam algoritma ini. Dalam setup N-bit kunci (N merupakan panjang dari kunci), kunci enkripsi digunakan untuk menghasilkan variabel enkripsi yang menggunakan dua buah array, state dan kunci, dan sejumlah-N hasil dari operasi penggabungan. Operasi penggabungan ini terdiri dari pemindahan(swapping) byte, operasi modulo, dan rumus lain. Operasi modulo merupakan proses yang menghasilkan nilai sisa dari satu pembagian. Sebagai contoh,berikut akan Algoritma setup kunci RC4 dalam bahasa C:

```
i := 0
j := 0
while GeneratingOutput:
    i := (i + 1) mod 256
    j := (j + S[i]) mod 256
    swap(&S[i],&S[j])
    output S[(S[i] + S[j]) mod 256]
endwhile
```

Untuk menunjukan cara kerja dari algoritma RC4, berikut akan Algoritma RC4 dalam bahasa C:

```
unsigned char S[256];
unsigned int i, j;

void swap(unsigned char *s, unsigned int i, unsigned int j) {
    unsigned char temp = s[i];
    s[i] = s[j];
    s[j] = temp;
}
void rc4_init(unsigned char *key, unsigned int key_length) {
    for (i = 0; i < 256; i++)
        S[i] = i;
    for (i = j = 0; i < 256; i++) {
        j = (j + key[i % key_length] + S[i]) & 255;
        swap(S, i, j);
    }
    i = j = 0;
}
/* PRGA */
unsigned char rc4_output() {
    i = (i + 1) & 255;
    j = (j + S[i]) & 255;
    swap(S, i, j);
    return S[(S[i] + S[j]) & 255];
}
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

#define ARRAY_SIZE(a) (sizeof(a)/sizeof(a[0]))

int main() {
    unsigned char *test_vectors[][2] =
    {
        {"Key", "Plaintext"},
        {"Wiki", "pedia"},
        {"Secret", "Attack at dawn"}
    };
    int x;
    for (x = 0; x < ARRAY_SIZE(test_vectors); x++) {
        int y;
        rc4_init(test_vectors[x][0], strlen((char*)test_vectors[x][0]));

        for (y = 0; y < strlen((char*)test_vectors[x][1]); y++)
            printf("%02X", test_vectors[x][1][y] ^ rc4_output());
        printf("\n");
```

```
    }
    return 0;
}
```

Metode RSA digagas oleh Ron Rivest, Adi Shamir,dan Leonard Adleman dari MIT pada tahun 1977. Walaupun sudah berumur lebih dari 30 tahun, metode ini masih banyak dipakai untuk merahasiakan data. Metode perahasiaan lain juga bisa dipakai bersamaan dengan RSA, membuatnya lebih aman, dan memang sekarang sistem RSA dianjurkan untuk dikombinasikan dengan metode lain (seperti Padding) agar keamanannya terjamin.

3. METODOLOGI PENELITIAN

3.1 Bentuk Penelitian

Bentuk penelitian yang penulis terapkan adalah tes awal – tes akhir kelompok tunggal *(one group pretest posttest)* dengan metode eksperimental.

Metode eksperimen yang dilakukan adalah dengan melakukan penyadapan pada jaringan komputer. Dalam penyadapan ini menggunakan tool *Wireshark* dan *Cain & Abel.* Yang mana wireshark memberikan penyadapan mentah*(raw data)* dan *Cain & Abel* memberikan hasil berupa penyadapan terfilter.

3.2 Metode Pengumpulan data

Dalam penelitian ini pengumpulan data dilakukan dengan mengambil data penyadapan yang diuji pada login HTTP dan login SSL(terenkripsi), Penyadapan yang dilakukan adalah penyadapan pasif dan aktif. Penyadapan pasif dikenal juga dengan *promiscuous mode*, yang mana sistem memerintahkan LAN atau WLAN card untuk mengambil paket data yang tidak ditujukan kepada sistem. Hampir semua LAN card mendukung *promiscuous mode,* tetapi tidak semua WLAN Card bisa melakukan *promiscuous mode*. Penyadapan pasif sangat efektif pada website HTTP.

Penyadapan aktif dikenal juga dengan nama *man in the middle attack* atau secara teknis disebut *arp spoofing.* Penyadapan aktif biasa dilakukan dengan me-*route* paket data ke komputer lain, biasanya dilakukan di jaringan switch maupun wireless karena *promiciuos mode* (penyadapan pasif) tidak bisa bekerja pada jaringan switch dan tidak optimal pada jaringan wireless karena tidak semua WLAN Card bisa mengaktifkan *promiscuous mode.* Selain itu penyadapan Aktif sering dikolaborasikan dengan *certificate spoofing* untuk melakukan penyadapan pada website SSL yang terenkripsi.

3.3 Metode Pengembangan Perangkat Lunak

Metode yang penulis gunakan dalam merancang perangkat lunak Proxy Tunnel ini adalah Metode *Extreme Programming.* Bahasa pemrograman yang digunakan adalah Visual basic 6.

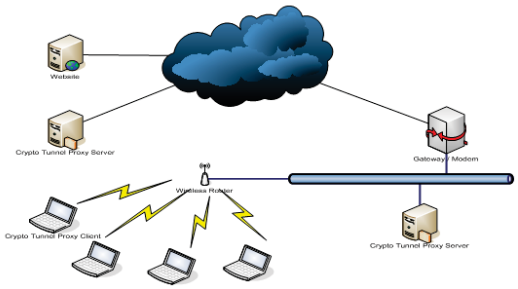
4. HASIL PENELITIAN

4.1. Perancangan Perangkat Lunak

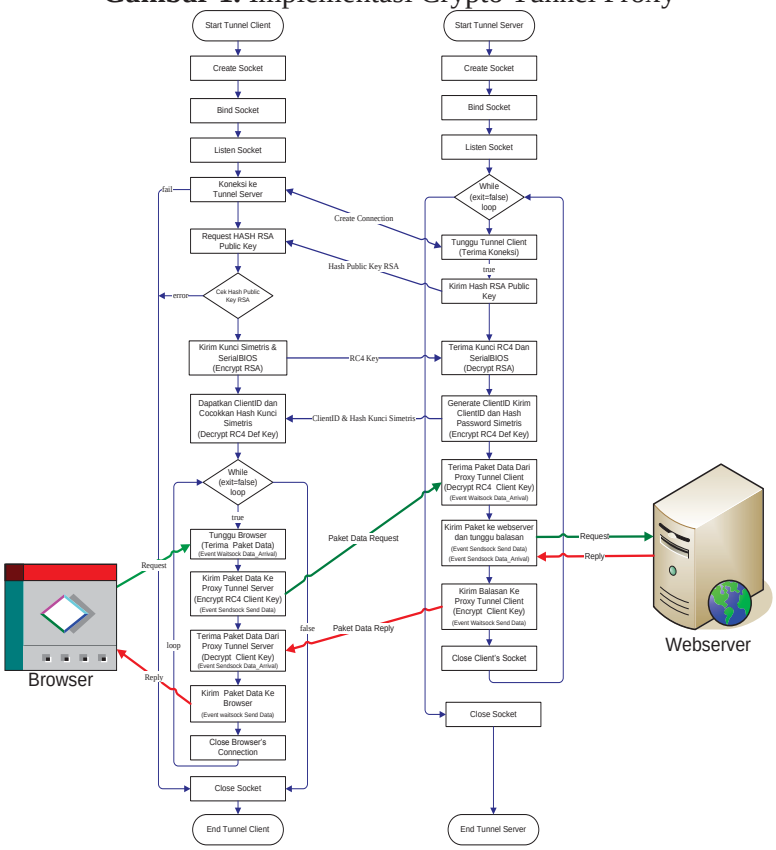
Sesuai dengan fungsi perangkat lunak yang penulis buat, perangkat lunak yang dihasilkan diberi nama *"Crypto Tunnel Proxy"* (CTP), yang terdiri dari Client dan Server. Dalam proses pengiriman data, *Crypto Tunnel Proxy* menggunakan *passenger protocol* IPv4, dan enkripsi yang dilakukan adalah pada pada level *socket*. Dalam perancangan hanya digunakan protocol TCP karena pada komponen winsock visual basic 6.0, protokol TCP memiliki kemampuan *Full Duplex* sedangkan protokol UDP tidak.

Metode enkripsi yang digunakan adalah metode enkripsi simetris (RC4) dan asimetris (RSA). Enkripsi RSA penulis gunakan untuk pertukaran kunci simetris, sedangkan enkripsi RC4 penulis gunakan untuk pertukaran data. Untuk mencegah pemalsuan kunci public RSA oleh pihak yang tidak bertanggung jawab, proxy tunnel client akan mengecek hash MD5 dan SHA-1 dari kunci public RSA milik proxy tunnel server untuk dicocokkan dengan kunci yang tersimpan di client.

Penggunaan perangkat lunak Crypto Tunnel Proxy dapat di server lokal maupun server di internet. Untuk implementasi agar perangkat lunak proxy tunnel dapat digunakan dan diuji secara umum, penulis menyewa sebuah VPS(Virtual Private Server) dengan Sistem Operasi Microsoft Windows Server 2003 untuk kemudian penulis instalasikan Crypto Tunnel Proxy Server. VPS ini penulis sewa selama satu bulan dari www.swvps.com. Akan tetapi penulis lebih menyarankan menginstalasi Crypto Tunnel Proxy Server dalam server lokal. Contoh implementasi Crypto Tunnel Proxy Server dapat di lihat pada gambar 1.



Gambar 1. Implementasi Crypto Tunnel Proxy



Gambar 2. Flowchart Aplikasi Proxy Tunnel

4.2. Pengujian Perangkat Lunak

Evaluasi perangkat lunak Crypto Tunnel Proxy ini didapatkan dari hasil pengujian perangkat lunak, dengan cara menguji perangkat lunak dengan menggunakannya pada aplikasi browser internet, dan browser internet digunakan untuk mengakses 2 jenis website yaitu SSL dan HTTP. Selain itu penulis juga menguji pada 3 jenis jaringan, yaitu wireless, LAN Hub dan LAN Switch, pengujian ini adalah pengujian terhadap penyadapan, website yang coba disadap adalah website HTTP.

4.2.1 Pengujian Pada Browser Internet

Pengujian ini dilakukan untuk melihat apakah tunneling dan enkripsi yang dilakukan berjalan sebagaimana mestinya, website HTTP dan SSL memiliki cara yang berbeda dalam komunikasi datanya, sehingga memerlukan prosedur yang berbeda dalam penanganannya.

Pada browser terdapat settingan network untuk proxy, settingan ini diisi dengan alamat proxy "127.0.0.1" dan port 8080(default), setelah Crypto Tunnel Proxy dan browser di konfigurasi maka dapat dilakukan pengujian.

Tabel 1
Pengujian Koneksi

No	Nama Browser	HTTP (www.google.co.id)	SSL (https://www.facebook.com)
1	Mozilla Firefox 3.5	OK	OK
2	Internet Explorer 8	OK	OK
3	Opera 10	OK	OK
4	Safari	OK	OK
5	Google Chrome	OK	OK

4.2.2 Pengujian Penyadapan Pada Jaringan

Penulis melakukan pengujian penyadapan password saat login menggunakan tool "Cain & Abel", website yang di uji adalah website HTTP dan SSL. Untuk pengujian ini penulis mengambi sample www.friendster.com sebagai website HTTP dan www.facebook.com sebagai website SSL yang akan disadap passwordnya saat user melakukan login, facebook tidak secara "penuh" menggunakan SSL tetapi login sudah menggunakan SSL.

Jenis penyadapan yang penulis lakukan adalah pasif dan aktif, penyadapan pasif adalah penyadapan tanpa melakukan *intervensi* terhadap komunikasi data yang terjadi, sedangkan penyadapan aktif melakukan *intervensi* terhadap komunikasi data yang terjadi. Teknik penyadapan aktif yang penulis lakukan adalah *arp poisoning* dan *certificate spoofing.*

Dalam pengujian ini, penulis menggunakan 2 kategori browser, yaitu browser yang sudah memiliki fitur CA(*Certificate Authority*) dan yang tidak memiliki fitur CA. Penulis memilih Mozilla Firefox untuk mewakili browser yang memiliki fitur CA yang baik, serta IE 5 (Browser default Windows) mewakili browser versi lama yang tidak memiliki fitur CA untuk kemudian dilakukan uji coba

Tabel 2
Pengujian Penyadapan Website HTTP (Firefox 3.5 dan IE5)

No	Jenis Jaringan	Normal		Menggunakan CTP	
		Pasif	Aktif	Pasif	Aktif
1	Wireless	Tersadap	Tersadap	Aman	Aman
2	LAN HUB	Tersadap	Tersadap	Aman	Aman
3	LAN SWITCH	Aman	Tersadap	Aman	Aman

Tabel 3
Pengujian Penyadapan Website SSL
Dengan Browser yang tidak memiliki fitur CA(IE5)

No	Jenis Jaringan	Normal		Menggunakan CTP	
		Pasif	Aktif	Pasif	Aktif
1	Wireless	Aman	Tersadap	Aman	Aman
2	LAN HUB	Aman	Tersadap	Aman	Aman
3	LAN SWITCH	Aman	Tersadap	Aman	Aman

Tabel 4
Pengujian Penyadapan Website SSL
Dengan Browser yang memiliki fitur CA(Firefox 3.5)

No	Jenis Jaringan	Normal		Menggunakan CTP	
		Pasif	Aktif	Pasif	Aktif
1	Wireless	Aman	Certificate Error	Aman	Aman
2	LAN HUB	Aman	Certificate Error	Aman	Aman
3	LAN SWITCH	Aman	Certificate Error	Aman	Aman

Tabel 5
Tabel Hasil Implementasi dan Pengujian pada Browser

No	Browser	Penyadapan		Keterangan
		Pasif	Aktif	
1	IE5	Aman	Aman	OK + Bug pada HTTP 1.0
2	IE7	Aman	Aman	OK
3	Mozilla Firefox 3.5	Aman	Aman	OK
4	Opera	Aman	Aman	OK
5	*Netscape Communicator*	Aman	Aman	OK
6	Safari 2.0	Aman	Aman	OK
7	SeaMonkey 2.0	Aman	Aman	OK

5. KESIMPULAN

Adapun kesimpulan yang dapat diambil dari penelitian yang penulis lakukan adalah perangkat lunak Crypto Tunnel Proxy dapat meningkatkan keamanan melakukan kegiatan berinternet. Penggabungan enkripsi simetris dan asimetris yang penulis rancang memberikan keamanan yang kuat tanpa mengorbankan terlalu banyak kecepatan. Saran yang diberikan penulis dari hasil penelitian yang dilakukan adalah sedapat mungkin menggunakan browser yang memiliki fitur *Certificate Authority*(CA) untuk melakukan browsing. Penulis dalam hal ini merekomendasikan Mozilla Firefox 3.5 dan Internet Explorer 8. Hindari browsing menggunakan browser versi lama karena walau terdapat fitur CA, tetapi telah kadaluarsa.

DAFTAR RUJUKAN

Ariyus, Dony., 2008, *Pengantar Ilmu Kriptografi*, Andi Offset, Yogyakarta.

Beck, Kent., 1999, *Extreme Programming Explained: Embrace Change*, Addison-Wesley

Brenton, Chris., Hunt, Cameron., 2003, *Network Security*, Diterjemahkan oleh : Jhoni Hidayat. Jakarta : PT. Elex Media Komputindo.

Kristianto, Andi. 2003, *Keamanan Data pada Jaringan Komputer*, Gava Media : Yogyakarta

Kruse, Robert L., Tindo, Clovis L, Leung, Bruce P., 1997, *Data Structure and Program Design in C*, Second Edition, USA : Prentice Hall International.

Stiawan, Deris., 2005, *Sistem Keamanan Komputer*, Jakarta: PT. Elex Media Komputindo

Syafrizal, Melwin., 2005, *Pengantar Jaringan Komputer*, Yogyakarta: Andi.