ANALISIS DAN IMPLEMENTASI SSL DENGAN METODE PERTUKARAN KUNCI DIFFIE-HELLMAN PADA NAGIOS NETWORK MONITORING SYSTEM

Saharuddin
Program Studi Sistem Informasi
STMIK Profesional Makassar
email : saharuddin@stmikprofesional.ac.id

Abstrak

SSL (Secure Socket Layer) adalah protokol keamanan yang didesain untuk dijalankan pada TCP/IP (Transmission Control Protocol/Internet Protocol) dan dengan mudah dapat digantikan dengan API soket UNIX-style standar yang digunakan oleh hampir semua perangkat lunak jaringan. Keamanan dijamin dengan menggunakan kombinasi dari kriptografi kunci publik dan iptografi kunci simetri bersamaan dengan sebuah infrastruktur sertifikat. Pada penelitian ini digunakan perangkat lunak open-source yaitu nagios sebagai Network Management Station. Alasan pemilihan nagios yaitu source code yang tersedia, telah mendukung dua tipe sistem pengawasan yaitu SNMP dan agen, dan memberikan kebebasan kepada pengembang untuk membuat sendiri program pengecekan yang akan terhubung ke modul utama dari nagios tersebut. Tipe sistem monitoring pada Nagios yang digunakan adalah berbasis Agen NRPE yang mendukung enkripsi SSL dengan metode pertukaran kunci Diffie-Hellman. Pertukaran kunci Diffie-Hellman terjadi setelah adanya pertukaran kunci publik yang dilakukan antara server monitoring dengan client. Kunci publik ini berupa nilai bilangan basis dan prima yang sudah ditentukan oleh Nagios. Berdasarkan hasil pengujian metode black box, implementasi SSL berjalan dengan baik dan sudah tidak adanya paket data yang berisi informasi penting yang dapat dibaca oleh Wireshark network protocol analyzer. Dengan menggunakan sistem monitoring jaringan nagios ini, suatu perusahaan yang memiliki banyak server dapat mengawasi aliran data pada jaringan perusahaannya.

Kata kunci : *SSL, SNMP, Sistem monitoring, Nagios, Client-server.*

A. PENDAHULUAN

Perkembangan teknologi informasi, khususnya jaringan memungkinkan terjadinya pertukaran informasi yang cepat dan semakin kompleks. Pengaturan jaringan yang baik tentu akan memaksimalkan pemanfaatan informasi tersebut. Oleh sebab itu jaringan harus dimonitoring sehingga kelancaran pengiriman informasi dapat berjalan dengan baik. Semakin besar dan luas sistem jaringan, semakin sulit untuk memonitoringnya.

Menjamin berjalannya semua infrastruktur sistem jaringan tersebut maka diimplementasikan sistem

monitoring untuk mempercepat diagnosis dan jika terjadi permasalahan akan mempercepat aksi untuk menghindari kerugian yang lebih banyak.

Salah satu contoh sistem *monitoring* jaringan adalah The dude yang dikembangkan oleh Mikrotik dan Nagios. The dude menggunakan *SNMP* (*Simple Network Management Protocol)* sebagai agen untuk melakukan *monitoring*. Akan tetapi, *SNMP* (*Simple Network Management Protocol)* yang merupakan protokol standar untuk *monitoring* bukanlah sebuah protokol yang didesain untuk keamanan. Hal ini ditambah dengan kenyataan data hasil *monitoring* yang dialirkan melalui protokol *SNMP* (*Simple Network Management Protocol)* merupakan *plainteks* yang dapat dilihat dengan mudah dengan suatu program *sniffer*. Kunci *SNMP* (*Simple Network Management Protocol)* dapat mudah ditemukan dan berbasis *plainteks*. Sedangkan Nagios dapat menggunakan *SNMP* (*Simple Network Management Protocol)* dan juga *NRPE (Nagios Remote Plugin Executor)* sebagai agennya. *NRPE (Nagios Remote Plugin Executor)* menggunakan *SSL (Secure Socket Layer)* untuk mengamankan komunikasi data antara *client server*.

SSL (Secure Socket Layer) adalah *protokol* keamanan yang didesain untuk dijalankan pada *TCP/IP* (*Transmission Control Protocol*/*Internet Protocol*) dan dengan mudah dapat digantikan dengan *API soket UNIX-style* standar yang digunakan oleh hampir semua perangkat lunak jaringan. Keamanan dijamin dengan menggunakan kombinasi dari *kriptografi kunci publik* dan *kriptografi kunci simetri* bersamaan dengan sebuah infrastruktur sertifikat. Sehingga dalam Tugas Akhir ini dikembangkan sebuah teknik untuk mengimplementasikan pengamanan data *monitoring* menggunakan *SSL* dengan pertukaran kunci *Diffie- Hellman* pada Nagios *Network Monitoring System*.

B. METODE PENELITIAN

Penelitian yang penulis lakukan berjudul Analisis dan Implementasi *SSL* dengan Metode Pertukaran Kunci *Diffie-Hellman* pada Nagios *Network Monitoring* System. Dalam impelementasi sistem ini penulis menggunakan Metodologi *System Development Life Cycle* (Siklus Hidup Pengembangan Sistem). Disebut SDLC, karena terdiri dari beberapa tahapan-tahapan pengembangan sistem yang membentuk suatu siklus hidup yaitu tahap analisis, desain, implementasi dan perawatan (Jogiyanto, 2005 : 9)

Dalam sebuah siklus SDLC terdapat 7 tahap umum (Hartono, 2004 : 18-19). Siklus hidup pengembangan ini dapat diuraikan tahapan-tahapannya sebagai berikut :

1. Tahap Perencanaan (*Planning*)
   Pada tahap ini dilakukan *feasibility study*, lokasi waktu, dan cakupan dari aplikasi yang akan dikembangkan.
2. Tahap Analisa (*Analysis*)
   Pada tahap ini akan diuraikan mengenai keadaan sistem sekarang, analisis proses komunikasi *client-server monitoring*, identifikasi masalah dan solusi pemecahan masalah.
3. Tahap Perancangan *(Design*)
   Tahap ini untuk menggambarkan topologi jaringan yang digunakan.
4. Tahap Pengembangan (*Development*)
   Pada tahap ini penulis melakukan pengembangan dengan instalasi dan konfigurasi terhadap komponen-komponen sistem yang diperlukan.
5. Tahap Ujicoba (Testing)
   Pengujian dilakukan dengan metode *Black Box* terhadap sistem yang telah selesai dibangun.
6. Tahap Implementasi (*Implementation*)
   Implementasi dilakukan dengan menerapkan sistem yang telah selesai melalui tahap pengujian untuk digunakan oleh user.
7. Tahap Pengoperasian dan Pemeliharaan (*Operations and Maintenace*)

   Pada tahap terakhir ini yang dilakukan adalah kegiatan-kegiatan untuk mendukung beroperasinya sistem yang akan dilakukan oleh admin.

Siklus SDLC ini dijalankan secara berurutan, mulai dari tahap 1 hingga tahap 7. Setiap tahap yang telah selesai harus dikaji ulang (*review*), kadang-kadang bersama *expert user*, terutama dalam langkah perencanaan dan desain untuk memastikan bahwa langkah-langkah dikerjakan dengan benar dan sesuai dengan harapan. Jika tidak maka langkah tersebut perlu diulangi lagi atau kembali ke langkah sebelumnya. Berikut ini akan diuraikan secara garis besar mengenai tahapan-tahapan siklus SDLC model waterfall.

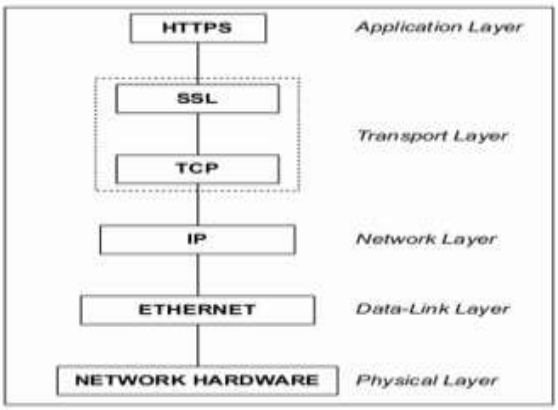
Secure Socket Layer

SSL adalah *protokol* keamanan yang digunakan pada hampir semua transaksi aman pada *internet*. *SSL* mengubah suatu *protokol* transport seperti TCP menjadi sebuah saluran komunikasi aman yang cocok untuk transaksi yang sensitif.

Protokol SSL mendefinisikan metode yang digunakan untuk membangun sebuah saluran komunikasi yang aman

dan tidak tergantung pada algoritma *kriptografi* mana yang digunakan. *SSL* mendukung berbagai macam algoritma *kriptografi*, dan berlaku sebagai sebuah *framework* di mana *kriptografi* dapat digunakan dengan cara yang tepat dan terdistribusi.

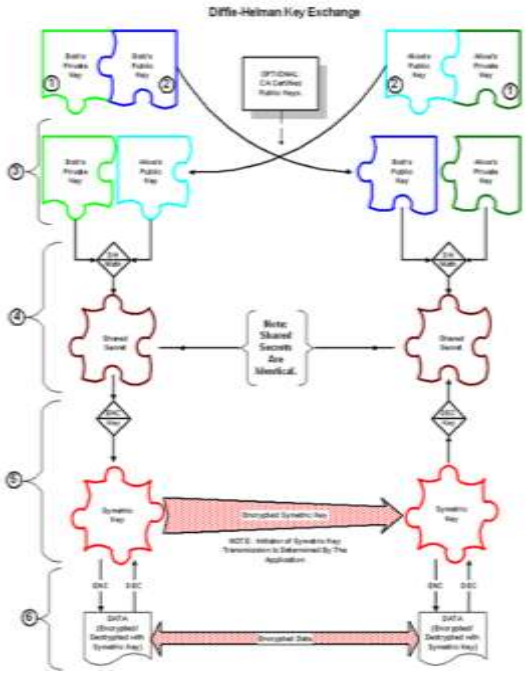
Penggunaan *SSL* sangat luas. Aplikasi yang membutuhkan pengiriman data melalui sebuah jaringan yang tidak aman seperti *internet* atau *intranet* perusahaan adalah salah satu aplikasi yang berpotensi untuk memanfaatkan *SSL*. *SSL* menyediakan keamanan, dan yang lebih penting adalah ketenangan. Dengan menggunakan ssl, kita dapat memastikan bahwa data kita aman dari pihak-pihak yang tidak berhak mengakses. SSL didesain untuk dijalankan pada TCP-IP.



Gambar 1. SSL Pada Model Referensih TCP-IP

Diffie-Hellman.

Diffie-Hellman bukan metode *enkripsi* dan tidak dapat digunakan untuk *enkripsi* data. Ini merupakan metode pertukaran kunci sekuritas dari *enkripsi* data. *Diffie-Hellman* mengkompilasi pertukaran sekuritas dengan membuat "*shared secret*" atau disebut dengan "*Key Encryption Key*" (KEK). antara dua perangkat. *Shared secret* kemudian di*enkripsi* dengan *kunci simetri*s untuk sekuritas pengiriman. *Kunci simetri*s terkadang disebut dengan *Traffic Encryption Key* (TEK) atau *Data Encryption Key* (DEK). Terkadang KEK digunakan untuk sekuritas pengiriman dalam TEK.

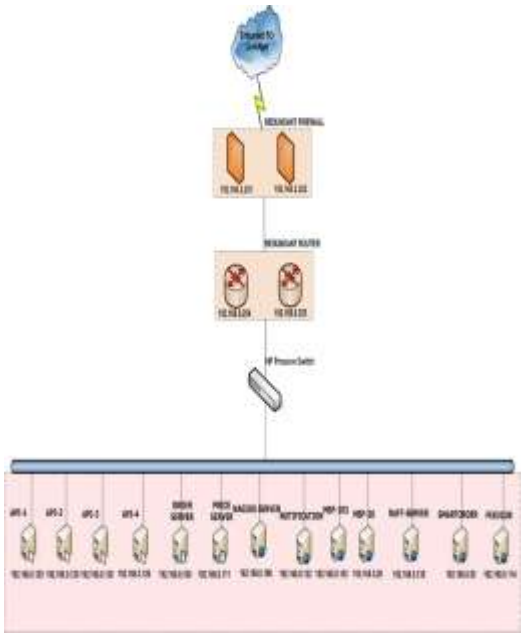


Gambar 2. Proses Pertukaran *Kunci publik* dan Pertukaran Kunci *Diffie-Hellman*

C. HASIL DAN PEMBAHASAN

Setelah dilakukannya analisa, maka yang dilakukan berikutnya adalah

menentukan perancangan topologi jaringan dan protokol pertukaran kunci Diffie-Hellman.



Gambar 3. Skema Topologi Jaringan

Proses Pertukaran Kunci Diffie-Hellman

Alghoritma DIFFIE-HELLMAN melibatkan dua kunci yaitu Private Key dan Public Key.

Data yang telah di enkripsi oleh Public Key hanya dapat di dekripsi oleh Private Key, walaupun Public Key diketahui oleh pihak lain. Pembuatan Private dan Public key terdiri dari 5 tahap, diantaranya:

1. Cari 2 Bilangan Prima secara acak dan simpan dalam variabel p dan q, dengan catatan jumlah bit untuk bilangan ini sama. Nilai p harus lebih besar dari q dan direkomendasikan minimal untuk menggunakan bilangan di atas 128bit/2 = 64bit *bila akan membuat kunci dengan bit-length sebesar 128 bit* (min 64bit hex = 0x8000000000000000; min 64bit decimal = 9223372036854775808).
2. Hitung n = p*q; Dimana nilai n ini akan digunakan untuk modulus pada private dan public key.
3. Hitung p q = (p-1) * (q-1); Untuk digunakan sebagai pencarian nilai private key.
4. Pilih nilai e untuk public key dengan syarat (1< e <pq). Nilai e ini biasanya merupakan nilai yang relative kecil, yang paling sering digunakan adalah 0x10001 = 65537. Bila kriteria e tidak cocok dengan syarat di atas, maka harus dicari nilai e lain yang sesuai, atau bila e sudah ditentukan dengan 0x10001, maka yang harus dicari kembali adalah nilai p, q, n dan pq seperti pada tahap awal.
5. Pilih nilai d, dengan syarat nilai d memenuhi: (d*e) mod pq = 1 Contoh Kasus Pembuatan Kunci dengan bilangan yang kecil.
6. Pilih 2 bilangan prima yang berbeda untuk p dan q. Misalnya: p=61 dan q=53
7. Hitung n=pq 61*53 = 3233
8. Hitung pq=(p-1) * (q-1); (61-1)*(53-

1) = 3120;

9. Pilih bilangan e dengan syarat (1 < e < 3120) dan gcd(e,3120)=1, kita ambil e=17, dimana 17 memenuhi syarat: (1 < 17 < 3120) dan (gcd(17,3120)=1).
10. Pilih nilai d, dimana (d*e) mod pq = 1. Kita ambil d=2753 dimana: (2753*17) mod 3120 = 1 46801 mod 3120 = 1

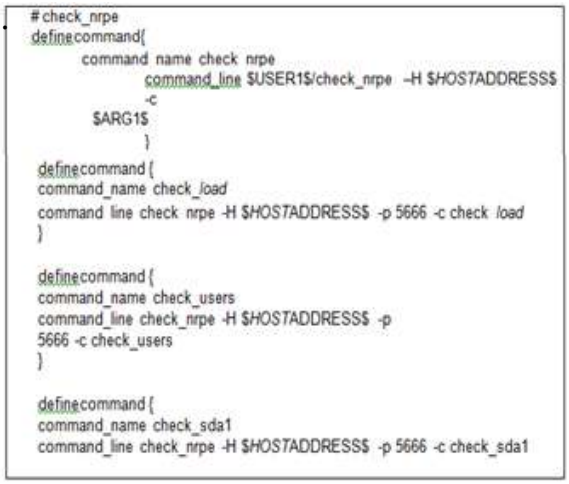
Dengan perhitungan tersebut maka telah mendapatkan Private dan Public Key, dimana Private Key adalah (n=3233 dan d=2753) dan Public Key adalah (n=3233 dan e=17).

Tahap hasil merupakan tahap dimana dilakukan instalasi Wireshark yang merupakan software untuk menguji tingkat keamanan dari sistem yang telah dibangun. Setelah implementasi *SSL* dengan metode pertukaran kunci *Diffie-Hellman* dibangun pada *Nagios Network Monitoring System*, maka diharapkan tidak ada data maupun informasi yang dapat terbaca oleh pihak luar.

Setelah proses instalasi selesai, hal yang pertama kali dilakukan adalah buka *web browser* dan akses ke url ***http://192.168.30.8*** Selanjutnya buka halaman nagios dan klik link tactical overwiew. Disana akan terlihat host yang sedang up adalah *localhost* dengan *service* standar yang dimonitoring seperti *current load, current users, HTTP, ping root partition,SSH, swap, usage, dan total processes.*

Setelah melakukan pembuatan file macosx-server.cfg, langkah selanjutnya adalah mendefenisikan file tersebut di dalam Nagios.cfg yang terletak pada /etc/Nagios/.

Gambar 4. Konfirmasi file Nagios. cfg



Gambar 5. Konfigurasi file Commands.cfg

D. KESIMPULAN

Dengan menggunakan sistem monitoring jaringan nagios ini, suatu perusahaan yang memiliki banyak server

dapat mengawasi aliran data pada jaringan perusahaannya.

Mengembangkan sistem *monitoring* jaringan yang mengimplementasikan *SSL* dengan metode pertukaran kunci *Diffie-Hellman* pada Nagios *Network Monitoring System* sebagai suatu solusi yang tepat untuk aliran data pengawasan dalam jaringan. Tentu saja selain keamanan yang terjamin solusi ini juga memiliki kecepatan dan kemudahan implementasi.

REFERENSI

[1] Alshamsi. Abdel Nasir dan Takamichi Saito. 2004. *A Technical Compration of IPsec and SSL*. Tokyo : Tokyo University of Technology.

[2] Amalia Rahma. 2010. *Analisis Pengamanan Data Menggunakan SSL/SSH Dengan Algoritma ECC Pada Transaksi Ecommerce.* Skripsi. Bandung : Program Strata I Jurusan Teknik Infomatika Institut Teknologi Bandung

[3] Joshua davies. 2011. Implementing SSL/TLS using cryptography and PKI. New York : Penerbit Wiley

[4] Privar Chandra, Matt Messier, John Viega. 2002. *Network Security with Open SSL*. Amerika Serikat : O'Reilly.

[5] Raf Knowledge. 2010 *Trik Memonitor jaringan*. Jakarta : Penerbit PT Elex Media Komputindo

[6] Ridwan. 2013. *Metode dan Teknik Menyusun Proposal Penelitian*. Bandung : Penerbit Alfabeta.

[7] Michael Ingga Gunawan. 2012. *Penggunaan Algoritma Diffie-Hellman dalam Melakukan Pertukaran Kunci.* Skripsi. Bandung : Program Strata I Jurusan Teknik Infomatika Institut Teknologi Bandung.

[8] Munir, Rinaldi. 2004. *Sistem Kriptografi Kunci Publik*. Bandung : Penerbit Informatika.