PROTOKOL PENGIRIMAN PAKET DENGAN DIGITAL SIGNATURE

William – NIM : 13506085
Program Studi Teknik Informatika, Institut Teknologi Bandung
Jl. Ganesha 10, Bandung
E-mail : if116085@students.if.itb.ac.id

Abstrak

Keamanan jaringan adalah sesuatu yang sangat penting, tetapi juga sesuatu yang sangat rentan terhadap serangan-serangan, seperti serangan *man in the middle attack*, *wormhole attack*, dll. Jaringan yang terkena serangan tentu saja tidak akan melakukan fungsinya dengan baik, jaringan tersebut menjadi tidak dapat mengirimkan paket yang benar kepada orang yang tepat. Bayangkan bila kita memiliki sebuah dokumen yang penting, lalu kita mengirimkannya melalui sebuah jaringan yang tingkat keamanannya rendah, maka dokumen tersebut dapat dengan mudah dilihat bahkan dicuri oleh orang lain yang dapat mengakses jaringan tersebut. Selain itu orang tersebut dapat mengambil dokumen tersebut dengan dokumen yang palsu. Pihak penerima tidak akan dapat memverifikasikan paket tersebut.

Dari cerita di atas kita masih dapat berargumen bahwa kriptografi dan tanda tangan digital pada dokumen digital akan menyelesaikan masalah tersebut. Hal ini memang benar, tetapi tidak sepenuhnya benar pada tingkat jaringan. Terkadang kita perlu untuk mengamankan file yang berbahaya pada tingkat jaringan dan apabila file itu baru dicek pada tingkat penerima bisa saja file tersebut membahayakan komputer penerima. Sehingga kita perlu sebuah protokol keamanan yang baik pada tingkat jaringan.

Pada makalah ini akan dibahas implementasi tanda tangan digital(*digital signature*) untuk melindungi jaringan pada tingkat pengiriman paket yang tentu saja menambah keamanan pengguna jaringan, karena jaringan sudah menjadi sebuah unsur yang sangat penting bagi semua pengguna teknologi modern saat ini,
***Kata kunci**: digital signature*, jaringan, paket, protokol, keamanan

1. Pendahuluan

Setiap tahun jumlah hacker terus meningkat, hal ini disebabkan semakin banyak pula tutorial dan buku-buku yang mengajarkan tentang ilmu hacking. Walaupun protokol-protokol keamanan juga semakin maju, tetapi router-router, switch, maupun server yang banyak digunakan di masyarakat bukanlah alat-alat dan protokol yang dilindungi oleh protokol keamanan yang maju tersebut. Umumnya elemen-elemen pembangun jaringan yang memiliki protokol keamanan yang baik harganya sangat mahal, seperti router dan switch keluaran Sisco.

Banyak pendekatan untuk melindungi jaringan dari serangan-serangan para hacker, salah satunya adalah dengan memberi tanda tangan digital pada setiap paket yang dikirimkan.

2. Dasar Teori

Untuk melihat lebih lanjut mengenai solusi pemakaian tanda tangan digital pada protokol pengiriman paket yang diajukan pada makalah ini, lebih baik kalau pembaca mengetahui beberapa pengetahuan tentang *security*, *network/socket programming*, dan tanda tangan digital itu sendiri.

2.1 Security

Definisi dari keamanan sangatlah besar, oleh karena itu saya mengklasifikasikan keamanan menjadi beberapa hal berikut ini :

- Kerahasiaan
  Pihak yang tidak diinginkan tidak dapat melihat/memata-matai suatu informasi tertentu.
- Integritas
  Pihak yang tidak diinginkan tidak dapat melakukan perubahan pada sebuah data/informasi tertentu.
- Autentikasi
  Pengguna harus dapat mengidentifikasikan dirinya kepada sistem.
- Otorisasi
  Sistem harus dapat menentukan hal-hal apa saja yang dapat dilakukan pengguna.
- Mencegah Penyangkalan
  Pengguna tidak dapat menyangkal akan hal-hal yang telah mereka perbuat.
- Pencatatan

Sistem harus memiliki log untuk mencatat hal-hal apa saja yang telah dilakukan para pengguna.

Sebenarnya pada sebuah jaringan kecil, seperti LAN penggunanya relatif sedikit dan keamanan jaringan masih dapat dikendalikan dengan mudah. Tetapi hal ini tidak berlaku pada sebuah jaringan besar, seperti internet. Ada beberapa hal yang harus diperhatikan untuk menjaga keamanan di jaringan :

- Di internet, kita tidak dapat mengendalikan siapa penggunanya.
- Kita tidak dapat mengendalikan infrastruktur dari internet itu sendiri.

2.2 *Network/Socket Programming*

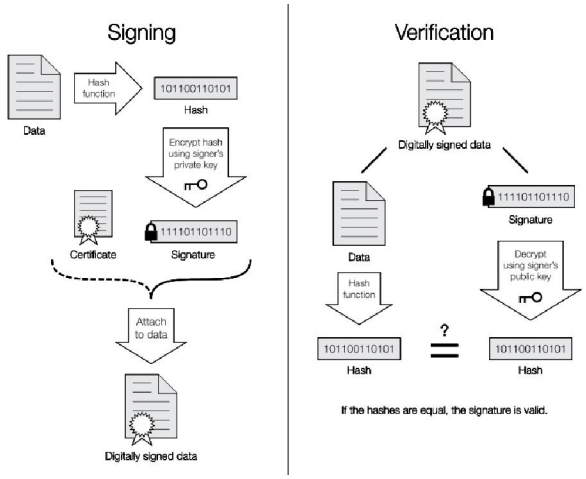
Pada dunia komputer, pemograman jaringan lebih dikenal dengan pemograman soket. Sebuah soket adalah sebuah sistem dari perangkat lunak yang membangun komunikasi dua arah antara aplikasi server dan aplikasi klien. Aplikasi server menyediakan *resources* untuk jaringan dimana aplikasi klien berada. Sedangkan aplikasi klien mengirim *request* ke aplikasi server, dan aplikasi server merespon *request* tersebut.

2.3 *Digital Signature*

Seperti prinsip dari sebuah tanda tangan biasa, tanda tangan digital digunakan untuk mengamankan sebuah file atau pesan yang dianggap penting. File atau pesan tersebut ditandatangani oleh pengirim, lalu diverifikasi oleh penerima. Dengan adanya tanda tangan digital tersebut adalah sebuah penanda untuk pihak penerima bahwa file atau pesan tersebut asli.

Untuk memakai tanda tangan digital dengan baik biasanya dibutuhkan sebuah skema. Skema tersebut terdiri dari tiga buah algoritma :

- Algoritma penggenerasi kunci
  Algoritma ini menggenerasi sebuah kunci privat secara random, lalu menghasilkan kunci public yang berkoresponden dengan kunci prifat tersebut.
- Algoritma penandatangan
  Algoritma ini akan menghasilkan tanda tangan digital pada sebuah pesan atau file, bila diberikan kunci prifat.
- Algoritma verifikasi tanda tangan
  Algoritma ini memverifikasi apakah sebuah file yang memiliki tanda tangan digital merupakan sebuah file yang asli, bila diberikan kunci publik.



Gambar 1 Skema Tanda Tangan Digital

3. Perbandingan Dengan Protokol yang Sudah Ada

Apa yang membedakan protokol yang diajukan di makalah ini dengan protokol keamanan yang sudah ada ? Bila kita melihat beberapa protokol keamanan yang sudah ada seperti :

- PGP : enkripsi/dekripsi/tanda tangan pada email.
- SSL : Secure Socket Layer
- SSH : Secure (remote) Shell
- HTTPS : HTTP over SSL

Sebenarnya protokol-protokol ini hanya bekerja pada tingkat aplikasi server dan aplikasi klien. Tetapi protokol keamanan yang diajukan pada makalah ini adalah tanda tangan digital yang diimplementasikan pada tingkat jaringan dan pengiriman paket. Dengan protokol ini, maka paket akan tidak ditandatangani di tingkat aplikasi server dan tidak juga diverfikasi pada aplikasi klien. Paket akan ditandatangani dan diverifikasi pada tingkat jaringan, untuk melakukannya router memegang peranan yang sangat penting. Protokol ini akan diimplementasikan pada router, seperti halnya pada servis proxy squid.

3.1 Kelebihan

Kelebihan dari protokol ini dibandingkan dengan protokol keamanan lain adalah keamanan di aplikasi klien maupun komputer atau perangkat keras klien itu sendiri meningkat. Peningkatan keamanan ini disebabkan satu hal yang sepele, yaitu

tidak masuknya paket palsu yang merupakan virus yang berbahaya ke komputer atau perangkat keras klien.

Pada protokol lain, karena paket diverifikasi pada tingkat aplikasi klien, maka pada intinya paket tersebut sudah ada pada komputer ataupun perangkat keras klien tersebut walaupun paket itu teridentifikasi sebagai paket palsu oleh aplikasi klien. Tidak ada jaminan bahwa aplikasi antivirus pada klien dapat mengatasi virus yang masuk ini, bahkan mungkin klien tidak memiliki aplikasi antivirus apapun.

Lalu, apakah paket yang merupakan virus itu aman pada router kita ? karena paket tersebut diverifikasi pada tingkat router, jadi virus itu sudah ada di router. Pada saat ini, hal tersbut tidak akan menimbulkan masalah karena biasanya virus dirancang untuk sebuah infrastruktur tertentu, dalam hal ini OS(*Operating System*).

3.2 Kelemahan

Kelemahan dari protokol ini ada pada dua hal, yang pertama adalah akan melambatnya kecepatan pengiriman paket. Hal ini pasti terjadi karena paket akan ditandatangani dan diverfikasi pada tingkat jaringan.

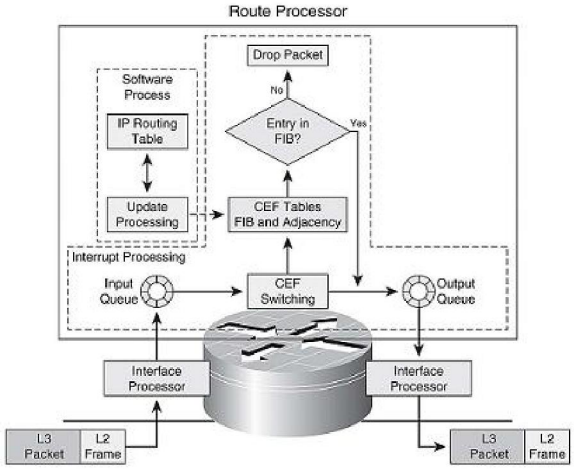
Kelemahan kedua adalah daya tahan router. Router dapat mengalami hang dikarenakan semua paket yang lewat akan melalui router tersebut dan router harus memverifikasi semua paket yang lewat. Hal ini berarti router harus memiliki daya tahan yang kuat baik itu secara fisik ataupun pengelolaan protokol pada router tersebut. Tentu kelemahan kedua ini kurang dirasakan bila komputer yang terhubung ke router sedikit, tetapi bila komputer yang terhubung ke router(terhubung secara tidak langsung, bisa melalui switch dan hub) berjumlah besar, maka router akan sangat rentan untuk berhenti berfungsi.

4. Implementasi

Sebenarnya tidak ada yang khusus dari tanda tangan digital yang akan digunakan pada makalah ini. Skema tanda tangan digital yang akan digunakan adalah skema umum. Saya akan lebih menekankan bagaimana skema tanda tangan digital ini diimplementasikan menjadi sebuah protokol pada router.

4.1 Struktur Router

Mungkin mengimplementasikan suatu protokol pada sebuah router bukanlah sesuatu yang biasa bagi kebanyakan orang. Diperlukan pemahaman mengenai infrastruktur router itu sendiri untuk mengetahui bagaimana kita membangun protokol tanda tangan digital pada pengiriman paket dan juga protokol memverfikasi paket tersebut.



Gambar 2 Proses Switching Paket Pada Router

Gambar di atas bisa digunakan untuk memahami proses switching paket yg sangat sederhana. Paket datang dari media network dan kabel tentunya dgn Layer 3 dan Layer 2 dari standar TCP/IP stack. *Interface* prosesor di router mampu untuk mengambil paket tsb, memeriksa header Layer 2 sekaligus membuang header tsb, dan mengirimkan paket tadi ke route processor utk diproses lebih lanjut. Sambil menunggu route processor melakukan lookup atau pencarian di *routing table* (dan *forwarding table*) tentunya paket itu harus disimpan di suatu *buffer* atau *queue*. Setelah *next hop* dari tujuan si paket ditemukan di tabel, maka route processor sekarang tahu ke interface mana paket harus dikirimkan. Kemudian paket dapat dipindahkan ke *output queue*, tempat untuk menunggu sebelum paket bisa dikirimkan

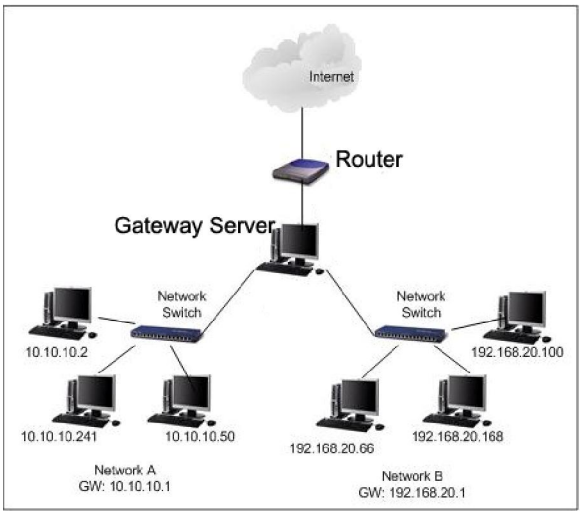
ke media *network*, dan paket akan di re-write atau mendapat layer 2 *header* yang baru yang berisi informasi untuk *next hop* berikutnya, kemudian paket keluar dari router melalui interface. *Input queue* atau *output queue* ini bisa virtual, jadi si paket sebenarnya berada di *memory* fisik yang sama dan tidak pernah berpindah. Tapi dengan membuat dua kondisi yang berbeda ketika paket berada di input queue (sebelum dilakukan *lookup*) dan ketika sesudah berada di output (setelah *lookup* dan tahu paket harus dikirim ke *interface* yang mana), maka router bisa menjalankan fitur atau melakukan perlakuan yang berbeda terhadap si paket di dua kondisi tersebut.

Bila kita perhatikan sebenarnya hanya ada dua jalur utama pada router yaitu jalur input dimana router menerima paket dari luar, dan jalur output dimana router mengirim paket yang ia terima ke luar. Jadi protokol penandatangan paket diletakkan pada jalur output dan protokol verifikasi paket diletakkan pada jalur input.

4.2 Cara Implementasi

Saya sudah berusaha mencoba untuk menambahkan protokol baru pada sebuah router, tetapi sampai makalah ini dibuat, usaha saya masih menghadapi kegagalan. Kegagalan ini dikarenakan oleh router yang tidak terbuka bagi pemakai untuk menambahkan protokol baru. Pemakai hanya dapat mengaktif atau menonaktifkan protokol-protokol yang sudah tersedia di dalam router tersebut.

Oleh karena itu saya mencoba cara lain, yaitu dengan menambah sebuah server *gateway* sebagai gerbang akhir sebelum semua komputer pada jaringan tersambung ke router. Jadi router tidak terkoneksi secara langsung ke switch, hub ataupun komputer lainnya pada jaringan ,melainkan ke *gateway* server tersebut. Tentu hal ini menyebabkan protokol yang diusulkan menjadi semakin mahal. Tetapi dengan cara ini implementasi protokol baru ini menjadi jauh lebih mudah dibandingkan dengan menambahkannya langsung ke dalam router.



Gambar 3 Skema yang Digunakan

Setelah membangun server *gateway*, langkah selanjutnya adalah melakukan pengaturan pada setiap komputer di jaringan, untuk melihat server tersebut sebagai sebuah proxy server. Dengan cara ini, maka skema protokol kita akan sangat mirip dengan skema server proxy.

Sekarang kita membangun aplikasi di server *gateway* ini, dianjurkan untuk memakai bahasa pemograman Java atau C karena kedua bahasa ini relatif cukup memudahkan pemogram untuk melakukan pemograman soket dan pemograman tingkat rendah. Aplikasi yang dibuat tidak perlu memiliki antarmuka, sehingga dibuat aplikasi sederhana yang dijalankan di *command line*.

Untuk tanda tangan digital, ada 3 kelas, yaitu :

- KeyGenerator.java
- Sign.java
- Validate.java

Untuk aplikasi servernya, ada 4 kelas yaitu :

- TCPServer.java
- TCPClient.java
- TCPServerThread.java
- TCPClientThread.java

TCPServer.java adalah kelas yang selalu dijalankan server gateway dan akan membangkitkan TCPServerThread.java untuk setiap ip di dalam jaringan lokal. Thread dibutuhkan sehingga setiap ip

memiliki threadnya sendiri. TCPServerThread.java adalah kelas yang bila dijalankan akan menerima paket dan pesan dari sebuah ip yang terkait dengannya, lalu kelas ini akan memanggil Sign.java dan menggunakan kunci prifat yang sudah dibangkitkan KeyGenerator.java . Kunci publik untuk meverifikasi paket yang sudah diberi tanda tangan kemudian diselipkan sebagai *footer* dari paket tersebut. Kemudian barulah paket tersebut dikirimkan ke router.

TCPClient.java adalah kelas yang selalu dijalankan server gateway dan akan memanggil kelas TCPClientThread.java bila ia menerima paket dari jaringan luar. Thread dibutuhkan sehingga setiap ip memiliki threadnya sendiri. TCPClientThread.java bila dijalankan akan menerima paket dari router ,lalu memanggil kelas Validate.java. Kelas Validate.java akan mengambil kunci publik yang menjadi footer pada paket dan memverifikasi paket tersebut. Bila paket dinyatakan valid ,maka paket akan dikirim ke ip yang bersangkutan.

5. Hasil Uji Coba

Uji coba yang dilakukan adalah uji coba sederhana dengan menggunakan 3 komputer (2 komputer biasa dan 1 komputer sebagai server gateway) dan sebuah router pada sebuah jaringan lokal. Dibuat sebuah aplikasi pengiriman paket sederhana, aplikasi ini hanya mengirim sebuah paket berformat .txt biasa.

Karena uji coba ini dilakukan pada jaringan lokal dan sebuah paket yang relatif kecil dan dalam kuantitas yang kecil, maka hasilnya cukup baik tanpa terasa kelambatan pengiriman paket. Tetapi karena ternyata percobaan ini dapat berjalan, maka dapat disimpulkan bahwa protokol ini mungkin untuk diimplementasikan, walaupun mungkin diperlukan skema yang kompleks untuk menerapkannya pada jaringan yang besar dengan jumlah paket yang sangat besar.

DAFTAR PUSTAKA

[1] Munir, Rinaldi. (2004). Bahan Kuliah IF5054 Kriptografi. Departemen Teknik Informatika, Institut Teknologi Bandung.

[2] Matt Bishop. A Security Analysis of the NTP Protocol Version 2. In Sixth Annual Computer Security Applications Conference,November 1990.

[3] Lundberg, J.; Packet level authentication protocol implementation; In Military Ad Hoc Networks; Series 1, No 19, Helsinki 2004

[4] Leslie Lamport. Password authentication with insecure communication. Communications of the ACM, 24(11):770–772,November 1981.

[5] Mihir Bellare, Ran Canetti, and Hugo Krawczyk. Keying Hash Functions for Message Authentication. In Advances in Cryptology – CRYPTO'96, edited by Neal Koblitz, volume 1109 of Lecture Notes in Computer Science, pages 1–15. Springer-Verlag, Berlin Germany, 1996.