Protokol Kriptografi Secure P2P

Protokol Kriptografi dalam Jaringan Peer To Peer

Andarias Silvanus (13512022)
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10, Bandung 40132, Indonesia
andarias.silvanus@gmail.com

***Abstrak* — Makalah ini membahas mengenai rancangan protokol kriptografi yang beroperasi di protokol komunikasi TCP/IP. Secara umum, protokol ini dapat dibagi menjadi dua sub-protokol utama, yaitu tahap *handshaking* dan tahap pengiriman. Tahap *handshaking* bertujuan untuk memastikan keaslian Bob dari pihak Alice, pertukaran kunci publik antara Alice dan Bob, dan distribusi kunci enkripsi simetris. Sedangkan pada tahap pengiriman, karena dioperasikan di atas TCP/IP, maka pengiriman akan dibagi per paket-paket. Paket-paket tersebut akan memiliki tipe konten (tipe *handshaking*, tipe pengiriman, atau tipe peringatan) yang disambung dengan data yang sudah dienkripsi dengan algoritma enkripsi yang sudah disetujui kedua belah pihak dan dengan menggunakan kunci simetris yang telah didistribusi sebelumnya. Data ini mengandung hash MAC atas pesan yang dikirimkan dengan menggunakan kunci simetris yang disambung dengan pesan yang hendak dikirimkan.**

Kata kunci — kriptografi, protokol, AES, MAC, RSA, SHA-3, peer to peer

I. PENDAHULUAN

Di tengah berjalannya era digital ini, pertukaran informasi digital berlangsung di berbagai tempat hingga melintasi antar negara dan benua. Sampai sejauh ini, pendistribusian data banyak menggunakan protokol komunikasi TCP/IP. Dalam protokol TCP/IP, setiap data yang hendak dikirimkan akan dipotong-potong menjadi paket-paket lalu dikirimkan menuju komputer tujuan. Di penerima sendiri paket-paket tersebut akan disusun berdasarkan urutannya.

Namun, karena tidak jarang informasi yang didistribusi bersifat privasi dan rahasia, beberapa pihak dapat menyadap dan mengambil pesan yang dikirimkan sehingga pihak yang tidak diinginkan tersebut dapat mengetahui isi pesan rahasia tersebut. Atau beberapa pihak dapat mengambil dan mengubah isi paket menjadi yang ia kehendaki untuk memenuhi tujuan pribadinya. Protokol komunikasi TCP/IP belum menangani masalah ini. TCP/IP tidak dirancang untuk mendeteksi adanya pihak penyadap dan tidak dapat mendeteksi jika adanya paket yang telah diubah.

Untuk mengatasi masalah ini, bidang studi kriptografi dipercaya dapat memberikan solusi atas permasalahan yang ada. Kriptografi sendiri merupakan seni sekaligus teknik pengelabuan pesan untuk menyembunyikan informasi atau pesan aslinya. Dalam perkembangannya, kriptografi terbagi menjadi dua era, yaitu era klasik dan era modern. Kriptografi dalam era modern banyak dipakai dalam pengelabuan pesan digital (enkripsi) yang berbentuk bit. Dampaknya, algoritma-algoritma kriptografi modern banyak dipakai dalam berbagai bentuk, termasuk salah satunya dalam perancangan protokol yang aman dalam pengiriman pesan. Dalam makalah ini, penulis berupaya untuk mengajukan sebuah rancangan protokol kriptografi yang beroperasi di atas *layer* TCP/IP yang berfungsi untuk menyediakan jalur yang aman bagi pengirim dan penerima dalam berkomunikasi.

II. DASAR TEORI

A. Protokol Komunikasi TCP/IP

Konsep dari TCP adalah setiap data yang diterima dari lapisan aplikasi akan dibagi per bagian (paket) dan lalu akan diteruskan pada lapisan jaringan. Setiap menerima paket, TCP akan mengirimkan paket lain yang berisikan pemberitahuan kalau paket tersebut telah diterima (*acknowledgement*/ACK). Dalam TCP, setiap paket yang dikirimkan akan disusun berurutan di tempat penerima untuk menjaga keutuhan data.

Protokol TCP/IP terbagi menjadi 4 lapisan, yaitu lapisan aplikasi (*application layer*), lapisan transportasi (*transport layer*), lapisan jaringan (*network layer*), dan *data link layer*.

1. *Application Layer*

   Lapisan ini adalah lapisan teratas dalam protokol TCP/IP. Lapisan ini mengandung pemrosesan yang melibatkan lapisan transportasi untuk mengirimkan data pada komputer yang dituju.

2. *Transport Layer*

   Lapisan ini bertindak sebagai *backbone* dari aliran data yang melibatkan dua komputer (penerima dan pengirim). T*ransport layer* menerima data dari *application layer*.

3. *Network Layer / Internet Layer*

   Fungsi utama dari lapisan ini adalah untuk mengorganisir aliran data pada jaringan. Dengan adanya aliran data, aliran tersebut harus dikontrol dengan menggunakan *routing* data pada jaringan.

Protokol utama yang digunakan dari lapisan ini adalah protokol IP.

4. *Data Link Layer*

   Lapisan ini terdiri dari *driver* dari perangkat yang dalam sistem operasi, dan perangkat penghubung jaringan, seperti *ethernet card*. Dari perangkat penghubung jaringan dan *driver* diadakan komunikasi agar pesan dapat diolah di sistem operasi.

Dalam protokol ini, setiap paket yang dikirimkan dari satu komputer ke komputer lainnya akan melewati lapisan-lapisan di atas secara berurutan. Contohnya, komputer pengirim yang hendak mengirimkan pesan akan ditangani pertama oleh lapisan aplikasi, lalu lapisan transportasi, lapisan jaringan, dan lapisan *data link*. Selanjutnya pesan akan sampai di komputer penerima yang akan ditangani pertama oleh lapisan *data link*, yang akan dibawa menuju lapisan jaringan, lapisan transportasi, dan akhirnya menuju lapisan aplikasi. Dalam setiap lapisannya, paket akan ditambah *header* tertentu bila berada di komputer pengirim dan *header* tersebut akan diolah bila sedang dibaca dalam komputer penerima.

B. Jenis Serangan pada Komputer

Untuk membuat protokol kriptografi, protokol tersebut harus kuat dalam menghadapi berbagai serangan yang akan terjadi. Oleh karena itu, perlu dipahami adanya beberapa jenis serangan yang umum terjadi dalam dunia komputer sebagai berikut:

1. Virus, *spyware, worm, trojan horse,*
2. Dampak yang disebabkan oleh virus yang dapat menyebabkan *vulnerability* terhadap komputer yang telah diinjeksi. Hal ini menyebabkan komputer tersebut menjadi lebih rentan terhadap serangan, misalnya serangan *back-door.*
3. *Hacking* dengan memanfaat celah dan berbagai teknik, seperti SQL *injection* dan *buffer overflow*.
4. *Phishing* yang berarti memancing/menipu pengguna untuk menuju ke arah yang salah atau menuju tempat yang diinginkan oleh *attacker*.
5. Serangan *denial-of-service*, yang merupakan serangan yang bertujuan untuk membuat sebuah komputer melayani berbagai permintaan yang tidak perlu atau membuat komputer sibuk hingga melewati batas dari kemampuannya. Karena hal inilah maka komputer tersebut akan *hang*. Serangan yang umum terjadi merupakan serangan yang dilakukan oleh banyak komputer (DDOS/ *distributed denial-of-service*) dengan menginjeksi berbagai botnet terhadap banyak komputer yang menjadikan komputer-komputer tersebut sebagai komputer zombie.
6. *Blended attack*, yang merupakan kombinasi dua atau lebih jenis serangan yang bertujuan untuk menyerang dari segala arah dengan lebih brutal agar target lebih mudah dijatuhkan.

III. RANCANGAN PROTOKOL KRIPTOGRAFI

Rancangan protokol kriptografi ini akan beroperasi di atas *application layer* dari protokol komunikasi TCP/IP. Pemilihan TCP/IP disebabkan karena banyaknya distribusi data yang menggunakan protokol tersebut. Secara umum, dalam rancangan protokol kriptografi akan memiliki tiga sub-protokol, yaitu sub-protokol *handshaking*, sub-protokol pengiriman, dan sub-protokol peringatan. Tapi sub-protokol yang utama adalah sub-protokol *handshaking* dan sub-protokol pengiriman.

Setiap pesan yang akan dikirimkan akan dipotong-potong dulu menjadi paket-paket. Di bagian depan setiap paket akan ditempelkan tipe konten untuk kelancaran proses protokol.

A. Sub-protokol Handshaking

Sub-protokol ini dijalankan ketika Alice (pengirim) pertama kali meminta jalur koneksi aman pada Bob (penerima). Secara garis besar, hal yang dilakukan dalam sub-protokol ini adalah verifikasi identitas kedua belah pihak, pertukaran kunci publik antara Alice dan Bob, dan persetujuan penggunaan algoritma enkripsi simetris beserta kuncinya. Tahap-tahap secara detailnya adalah sebagai berikut:

1. Ketika Alice hendak meminta jalur koneksi aman, Alice terlebih dahulu membangkitkan kunci privat dan kunci publik yang dibangkitkan dari *seed* yang diambil dari angka acak, yang terdiri dari IP dinamis Alice, tanggal dan waktu pengaksesan, dan angka acak yang dibangkitkan oleh algoritma tertentu.
2. Alice mengirimkan pesan halo pada Bob dalam keadaan belum terenkripsi dengan konten sebagai berikut:

   a. versi protokol kriptografi ini yang dapat Alice gunakan,

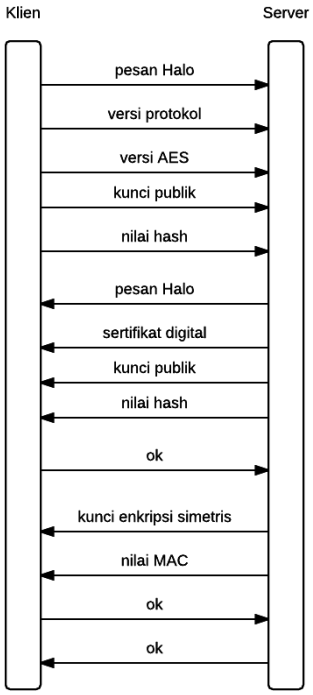
   b. versi AES yang dapat Alice dalam gunakan. Secara *default*, versi yang akan digunakan adalah AES-128. Alice harus bisa menjalankan algoritma enkripsi AES dan RSA. Pertimbangan keharusan ini didasarkan atas algoritma enkripsi standar yang telah diakui dan sudah banyak *browser* yang semakin canggih yang sudah mendukung kedua algoritma enkripsi standar tersebut. Apabila Alice tidak *support* salah satu dari algoritma enkripsi tersebut, maka proses akan dibatalkan dan menggunakan protokol SSL.

   c. kunci publik Alice,

   d. tanda tangan digital Alice,

   e. nilai *hash* atas keempat komponen di atas yang digabungkan sesuai urutan sebelumnya dengan menggunakan algoritma SHA-3. Tujuan dari ditempelkannya nilai *hash* adalah untuk menjaga integritas data dan untuk pengecekan apakah data yang diterima berubah atau tidak selama perjalanan.

3. Bob menerima pesan halo dari Alice dan melakukan fungsi *hash* atas keempat komponen pertama dari pesan halo, lalu membandingkannya dengan nilai *hash* yang terdapat pada pesan halo.
4. Bob membangkitkan kunci privat dan kunci publik dari *seed* angka acak yang juga dibangkitkan oleh Bob.
5. Bob mengirimkan pesan halo pada Alice dengan keadaan seudah terenkripsi oleh kunci publik Alice. Konten yang terdapat dalam pesan halo Bob adalah sebagai berikut:
    a. tanda tangan digital Bob,
    b. kunci publik Bob,
    c. nilai *hash* atas kedua komponen diatas yang digabungkan sesuai urutan.
6. Alice menerima pesan halo dari Bob, lalu mendekripsi pesan tersebut dengan menggunakan kunci privat dari Alice. Setelah itu Alice akan mengotentikasi kesamaan nilai *hash* antara fungsi *hash* yang dilakukan terhadap kedua komponen pertama dengan nilai *hash* yang ditempelkan di akhir pesan halo. Jika sama, Alice akan mengecek keaslian Bob berdasarkan tanda tangan digital yang telah didapatkan.
7. Alice akan mengirimkan pesan ok pada Bob jika tidak didapati masalah. Mulai dari sini, setiap pengiriman yang dikirimkan dari Alice sudah dienkripsi dengan kunci publik Bob, dan begitu juga dengan Bob yang mengenkripsi pesan dengan menggunakan kunci publik Alice.
8. Bob yang menerima pesan ok akan membangkitkan kunci enkripsi simetris. Selanjutnya Bob akan mengirimkan pesan pada Alice dengan konten sebagai berikut:
    a. kunci enkripsi simetris AES,
    b. nilai MAC atas kunci enkripsi simetris yang dibawa yang dijalankan dengan menggunakan kunci enkripsi simetris tersebut.
9. Alice menerima pesan Bob, mendekripsinya, dan mengotentikasi nilai MAC atas kunci enkripsi simetris. Jika tidak ada masalah, Alice akan mengirimkan pesan ok.
10. Bob yang menerima pesan ok akan mengirimkan pesan ok sebagai tanda untuk memasuki fase pengiriman.



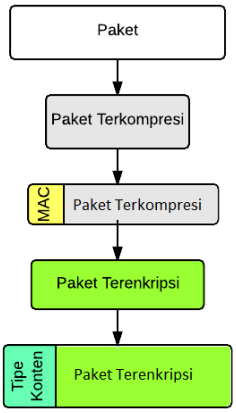
Ilustrasi Sub-protokol Handshake

Jika Alice tidak dapat mendekripsi pesan dari Bob, atau dapat didekripsi namun tidak sesuai dengan nilai *hash*, Alice akan meminta Bob untuk kembali mengirimkan pesannya. Namun jika hal ini terjadi lebih dari 10 kali, maka Alice akan memperingati pengguna terkait adanya kemungkinan penyadap / *middle-man*. Hal yang sama juga terjadi di pihak Bob. Jika terjadi lebih dari 10 kali, Bob akan mengirimkan pesan pada Alice yang memberitahukan kemungkinan adanya penyadap. Bob akan mengirimkan maksimal sampai 5 kali sampai Bob mendapat balasan dari Alice. Jika tidak mendapat balasan, Alice akan mendapat *timeout* dan memperingati pengguna.

Setiap paket *handshake*, sesudah dari tipe konten yang ditempelkan, akan ditempelkan tipe pesan. Tipe pesan itu mencakup pesan halo, pesan kunci, pesan ok, dan pesan peringatan yang akan diolah oleh masing-masing Alice dan Bob.

B. Sub-protokol Pengiriman

Setiap paket yang telah dibagi akan dikompres untuk mengurangi besar ukuran paket yang akan membuat pengiriman berlangsung lebih cepat. Setelah itu, fungsi MAC akan dijalankan dengan menggunakan kunci enkripsi simetris yang dimiliki kedua belah pihak terhadap isi pesan yang hendak dikirimkan. Nilai dari fungsi MAC tersebut akan ditempelkan ke bagian depan dari paket tersebut. Selanjutnya, keseluruhan paket akan dienkripsi dengan algoritma AES. Tipe konten akan ditempelkan di bagian terdepan dari paket.



Ilustrasi Sub-protokol Pengiriman

Bob yang menerima akan mengolah paket dengan membalikkan proses yang dilakukan Alice. Pertama, Bob akan mengecek tipe konten dari paket yang dikirim oleh Alice. Selanjutnya, Bob akan mengambil isi dari paket tersebut dan mendekripsinya dengan kunci simetri. Karena panjang dari nilai MAC sendiri adalah tetap, maka Bob hanya perlu mengambil nilai MAC dan mengkalkulasi letak dari isi pesan (plainteks) yang telah dikompresi. Bob akan membandingkan kesamaan antara nilai MAC yang didapat dengan nilai dari fungsi MAC yang dikalkulasi Bob terhadap pesan kompresi tersebut untuk mengecek integritas data. Jika sama, Bob perlu melakukan dekompresi pesan untuk mengembalikan isi dari pesan asli. Namun jika berbeda, Bob akan memberitahu Alice untuk mengirim ulang paket dengan nomor urut tersebut.

C. Sub-protokol Peringatan

Paket dengan jenis ini dikirimkan oleh Bob pada Alice jika terdapat masalah, seperti ketidakcocokan nilai *hash* dari yang telah dikalkulasi dengan nilai *hash* yang dikirimkan setelah Bob mengkalkulasi lebih dari 10 kali terhadap paket yang berjenis sama yang dikirimkan berkali-kali oleh Alice karena adanya ketidakcocokan nilai *hash*. Paket jenis ini akan mengandung tipe konten yang disambung dengan tipe peringatan. Paket jenis ini juga akan tetap dienkripsi dengan kunci publik Alice.

D. Penanganan Akan Serangan

Jika telah diketahui kemungkinan adanya *attacker*, maka Bob akan memutuskan jalur hubungan secara otomatis dan Alice akan mencoba menghubungi Bob melalui jaringan yang berbeda untuk menghindari keberadaan dari *attacker* tersebut.

IV. ANALISIS

4.1 Analisis Terhadap Ancaman

Saat Alice ingin meminta jalur koneksi yang aman pada Bob, maka hal pertama yang Alice lakukan adalah mengirimkan pesan halo pada Bob dalam keadaan yang belum terenkripsi dalam fase *handshake*. Apabila ternyata *attacker* sudah ada di tahap ini, *attacker* dapat dengan mudah membaca isi dari pesan yang belum terenkripsi tersebut. Namun *attacker* tidak akan mendapat apa-apa selain kunci publik Alice dan nilai *hash*. Jika seandainya dalam kasus terburuk dari skenario ini, *attacker* adalah orang yang memiliki kekuasaan dalam mengontrol *traffic* jaringan dan ia dapat mengganti paket yang dikirim Alice dengan paket yang mengandung kunci publiknya ditambah nilai *hash*-nya dan dikirimkan pada Bob. Bob akan membaca dan mengirimkan paket halo pada Alice dengan kunci publik *attacker*, namun saat paket berada di tangan Alice, Alice tidak dapat mendekripsi paket tersebut dengan kunci privatnya, sehingga Alice akan meminta lagi paket halo Bob. Jika hal ini terjadi lebih dari 10 kali, sesuai aturan protokol, Bob akan mengirimkan paket peringatan pada Alice terkait adanya kemungkinan *attacker* di tengah-tengah mereka. Namun jika seandainya paket peringatan tersebut dicuri sehingga tidak pernah sampai ke tangan Alice, Alice akan terkena *timeout* secara terus menerus yang akan melindungi pengguna secara otomatis.

Jika seandainya *attacker* menyelinap saat Alice mengirimkan pesan halo pada Bob, mengambil pesan halo tersebut sehingga Bob tidak pernah mendapatkan pesannya, dan *attacker* bertindak seolah-olah sebagai Bob. Alice yang kemudian mendapat pesan halo dari *attacker* akan membandingkan antara alamat IP yang dituju (IP Bob) dengan alamat IP dari dikirimkannya paket tersebut (IP *attacker*). Jika ternyata tidak sama, jelas terdapat *attacker* di tengah-tengah mereka dan *attacker* pun dapat dilacak karena telah didapatkan alamat IP-nya.

Jika *attacker* menyerang pada waktu lain, setelah keadaan paket telah terenkripsi oleh kunci publik masing-masing pihak. Maka jika keadaan paket berubah masing-masing pihak akan langsung mengetahui adanya perubahan pada paket tersebut karena dapat terdeteksi dari melihat kesamaan nilai MAC/*hash* dari yang ditempelkan pada paket dengan nilai dari fungsi MAC/*hash* yang telah dikalkulasi sendiri oleh masing-masing pihak. Jika hal ini terjadi, maka satu pihak akan meminta pengiriman ulang paket yang sama terhadap pihak lainnya. Dan jika seandainya *attacker* terus menyerang paket, akan terus terjadi ketidaksinkronan nilai MAC/*hash* yang mengakibatkan setelah permintaan terjadi lebih dari 10 kali, maka masing-masing pihak akan mengetahui kemungkinan adanya *attacker* di antara mereka dan langsung dapat memberi tahu pengguna.

Jika *attacker* menyerang pada saat fase pengiriman, di saat keadaan paket telah terenkripsi dengan kunci simetris, maka penanganan yang berlangsung akan sama seperti penanganan terhadap ancaman terhadap paket terenkripsi oleh kunci publik dalam fase *handshaking*.

.

4.2 Perbandingan Hasil

Dengan diimplementasikannya rancangan protokol kriptografi ini, tentu akan menimbulkan kelebihan dan kekurangannya. Kelebihannya adalah:

1. Penanganan terhadap ancaman yang cukup kuat, dapat dilihat dari enkripsi paket yang telah menggunakan sejumlah algoritma enkripsi yang telah teruji (AES dan RSA).

2. Pengidentifikasian aktor yang jelas, hal ini terjadi karena didukung oleh mekanisme pengecekan perbandingan alamat IP tujuan dengan alamat IP diterimanya pesan, serta tanda tangan digital yang dibubuhkan.
3. Integritas data paket, keintegritasan data selalu dijaga dengan pengecekan nilai *hash* maupun nilai MAC.
4. Ukuran *header* paket yang cukup kecil. Hal ini terjadi karena informasi yang berada di dalam *header* cukup ringkas, dan kalkulasi terhadap letak pesan asli dapat dihitung dalam protokol karena panjang *header* yang tetap. *Header* tersebut adalah tipe konten (2 bit) dan tipe pesan (3 bit). Dalam fase pengiriman, tidak dibutuhkan *header* tambahan untuk mengetahui letak pesan terkompresi karena diketahui panjang nilai MAC yang selalu tetap. Karena ukuran *header* yang cukup kecil, data yang dikirimkan dapat semakin banyak.

Namun adapun kelemahan dari protokol ini adalah jika seandainya terjadi penyerangan, membutuhkan waktu yang cukup lama untuk membentuk jalur koneksi yang aman. Hal ini dikarenakan kedua belah pihak harus menunggu selama 10 kali dari pengiriman paket untuk memastikan adanya kemungkinan *attacker*, dan setelah itu Alice harus mencari jaringan yang berbeda untuk menghindari keberadaan *attacker* sekaligus sebagai jalur alternatif untuk menghubungi Bob. Fase *handshake* perlu diulangi lagi dalam tahap ini.

V. KESIMPULAN

Rancangan protokol kriptografi ini dapat menjadi alternatif dalam jaringan *peer to peer*. Walaupun terdapat kekurangan dalam rancangan protokol ini, namun diharapkan dengan pengembangan selanjutnya, kekurangan ini dapat semakin ditekan sehingga dapat benar-benar diimplementasikan dalam jaringan *peer to peer*.

REFERENSI

[1] Rinaldi Munir, 10 Mei 2015, 14:32, Tersedia dari http://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/

[2] TheGeekStuff, 10 Mei 15:20, Tersedia dari http://www.thegeekstuff.com/2011/11/tcp-ip-fundamentals/

[3] TCPIPGuide, 10 Mei 15:45, Tersedia dari http://www.tcpipguide.com/free/t_InternetProtocolConceptsandOverview.htm

[4] Symantec, Norton 10 Mei 16:25, Tersedia dari http://www.symantec-norton.com/11-most-common-computer-security-threats_k13.aspx