Studi Kasus Metode Enkripsi Dibalik Lalu Lintas "BitTorrent"

Mohammad Faizal Hitobeli – NIM 13506057[1]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]if16057@students.if.itb.ac.id

Abstrak – Makalah ini membahas tentang perkembangan teknologi peer-to-peer file sharing di internet yang menggunakan enkripsi protokol. Pembahasan kali ini dibatasi pada fenomena perkembangan BitTorrent, yaitu salah satu peer-to-peer program. Program ini memungkinkan penggunanya untuk memindahkan data dalam jumah besar antar komputer melalui fasilitas internet dengan fitur protocol encryption.

Protocol encryption adalah poin yang akan dibahas dalam makalah ini. Data yang dikirim oleh para pengguna secara langsung dienkripsi oleh program agar lalu lintas pengiriman data sulit dideteksi oleh pihak ketiga, termasuk juga bagi ISP (Internet Service Provider). Metode tersebut dapat menjaga kerahasiaan suatu aktifitas transfer data. Protocol encryption menggunakan Diffie-Hellman key dan RC4 cipher.

Protocol encryption masih dapat ditembus dengan metode tertentu. Perkiraan kekuatan enkripsi ini berada pada kisaran 60 s.d. 80 bits key length. Angka tersebut tergolong kecil untuk sekarang, tetapi terus digunakan karena protokol ini memerlukan waktu proses pada CPU yg relatif cepat, sehingga aktivitas pengiriman file dapat berlangsung cepat pula.

Kata Kunci - Enkripsi, BitTorrent, lalu lintas, protocol encryption.

I. PENDAHULUAN

CacheLogic, suatu firma di Cambridge yang menganalisa lalu lintas internet, memaparkan bahwa sepertiga data yang dikirim oleh pengguna internet merupakan lalu lintas BitTorrent. Apa sebenarnya BitTorrent, dan yang menyebabkan program tersebut dapat mendominasi internet? Statistik tersebut tentu tidak pasti merepresentasikan banyaknya pengguna internet yang menggunakan program tersebut, karena yang dianalisa adalah besar byte data yang ditransfer secara keseluruhan. Lain halnya bila yang dianalisa adalah address-address html yang dikunjungi atau ip address.

Dapat diketahui dari berbagai artikel BitTorrent, diciptakan oleh Baron Cohen, adalah suatu program peer-to-peer yang memungkinkan file transfer dalam jumlah besar. Hal ini menyebabkan program ini banyak digunakan dan disukai oleh pengguna internet. Pada awalnya Cohen memaparkan program open source ini diharapkan membantu orang-orang yang ingin berbagi software berbasis operasi Linux secara online. Tetapi akhirnya banyak yang memanfaatkan untuk berbagi file-file musik, film, dan berbagai karya seni audio maupun audio-visual yang memiliki paten.

Hal ini menjawab pertanyaan pertama di atas, namun menimbulkan pertanyaan baru. Apa yang menyebabkan pengguna dapat dengan bebas membagi file-file tersebut? Mengapa pihak ISP tidak dapat mengendalikan file transfer tersebut? Masalahnya terletak pada protokol enkripsi yang digunakan Cohen pada program BitTorrent dalam prosedur transfer file melalui internet. Hal ini dimaksudkan untuk mencegah serangan dari pihak ketiga yang mencoba mencuri informasi yang sedang dipertukarkan antara dua komputer melalui lalu lintas BitTorrent pada internet. Tidak terkecuali pihak ISP, sehingga mereka relatif sulit untuk melacak apa saja yang sedang dipertukarkan, misal antara komputer X dan Y.

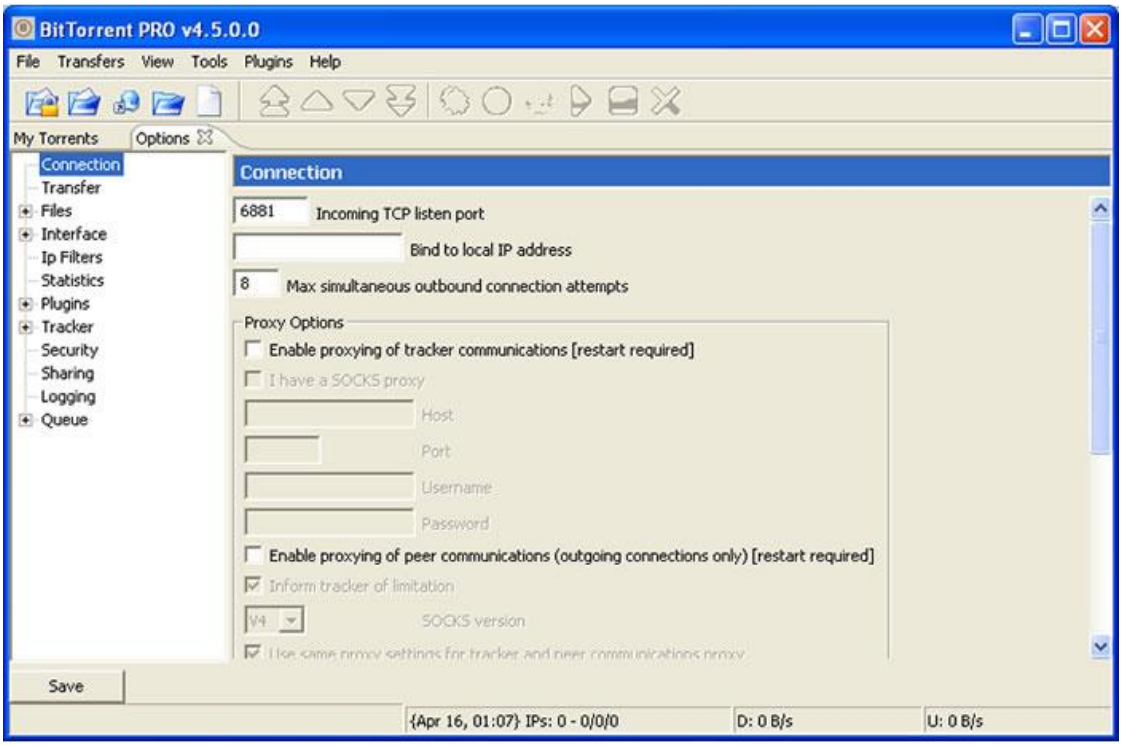
Metode enkripsi yang digunakan BitTorrent, dan sedikit penjelasan mengenai program BitTorrent sendiri selanjutnya akan dibahas lebih dalam pada bab 2.

II. STUDI PROTOKOL ENKRIPSI "BitTorrent"

Protocol header encryption (PHE) diterapkan terlebih dahulu untuk mengamankan pertukaran file melalui lalu lintas BitComet, client dari BitTorrent. Namun masih dapat ditembus karena hanya sebagian dari file yang dienkripsi. PHE selanjutnya tidak diterapkan.

Untuk menanggulanginya para developer program merancang suatu metode baru untuk mengamankan pertukaran data yang disebut message stream encryption (MSE). Masih kekurangan beberapa fitur penting, selanjutnya MSE diupgrade dan akhirnya muncul metode yang disebut protocol encryption (PE). Versi PHE tidak kompatibel dengan MSE/PE sehinnga sudah ditinggalkan. Sedangkan MSE dan PE sebenarnya metode yang tidak jauh berbeda. Hanya masalah penamaan, sebelum PE lengkap, metode tersebut dipanggil MSE. Setelah disempurnakan para

developer menyebutnya PE.



Gambar 1. Interface BitTorrent pada windows

Gambar 2. Interface BitTorrent OSX

PE menggunakan Diffie-Hellman key exchange yang kemudian dipadu dengan info-info pengguna torrent, yang tersusun seperti tabel hash dinamik, untuk menentukan kunci komunikasi tertentu. Kemudian data dienkripsi menggunakan RC4, baik headernya saja maupun seluruh koneksi, tergantung opsi yang dipilih pengguna. Mengenkripsi seluruh koneksi membutuhkan cpu processing time yang lebih lama.

Diffie-Hellman mencegah komunikasi diambil oleh pihak lain, atau sering disebut passive listener. Info hash digunakan untuk mencegah pihak lain memanipulasi data yang dikirim. RC4 dipilih karena metode ini membutuhkan waktu proses yang relatif cepat.

2.1. Diffie-Hellman Key Exchange

Diffie-Hellman key exchange memungkinkan dua pihak, yang tidak memiliki informasi mengenai satu dengan yang lainnya, bergabung membangun sebuah komunikasi menggunakan kunci simetris. Diperkenalkan oleh Whitfield Diffie dan Martin Hellman pada tahun 1976. Terdapat beberapa sinonim untuk diffie-hellman key exchange, yaitu:

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

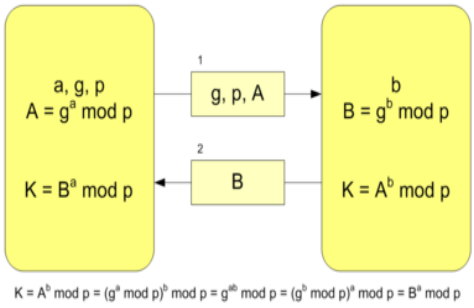
Ralph Merkle cukup berpengaruh dalam metode ini di bidang distribusi kunci publik. Selain itu John Gill turut berkontribusi dengan saran logaritma diskritnya.

Diffie-Hellman key exchange dapat dijelaskan sebagai berikut:

1. A dan B menggunakan bilangan prima p dan suatu bilangan g yang akan digunakasn sebagai basis.

2. A memilih suatu bilangan integer *a* dan mengirim B (g^a mod p).
3. B memilih suatu bilangan integer *b* dan mengirim A (g^b mod p).
4. A menghitung (g^b mod p)^a.
5. B menghitung (g^a mod p)^b.

Dapat digambarkan seperti beikut.



Gambar 3. Alur Diffie-Hellman Key Exchange

A dan B memiliki nilai dari (g^a mod p)^{b}dan (g^b mod p)^a yang sama karena sifat asosiatif perpangkatan, $(g^b)^a$ = $(g^b)^a$. Nilai itu adalah kunci untuk mengenkripsi dan mendekripsi pesan A dan B. Dengan menggunakan nilai p, a, dan b yang lebih besar maka kunci yang didapat akan lebih sulit dipecahkan oleh pihak ketiga.

Diffie-Hellman key exchange termasuk aman bila pangkat dari g dipilih secara tepat. Kunci yang terdiri dari bilangan prima sulit ditembus, karena terdapat sebuah algoritma yang dapat mencari logaritma diskrit dikenal dengan algoritma Pohlig-Hellman

Bagaimanapun juga Diffie-Hellman key exchange rentan terhadap serangan pihak ketiga, yang sering disebut *man-in-the-middle-attack*. Pihak ketiga membajak nilai (g^a mod p) milik A yang dikirim ke B dan mengirim nilai (g^a mod p) miliknya sendiri ke B. Begitu pula dengan B yang mengirim nilai (g^b mod p) pihak ketiga juga membajaknya dan mengirim nilai (g^b mod p) miliknya sendiri ke A. Sehingga kunci publik yang dipakai oleh A dan B sebenarnya adalah kunci pihak-ketiga. Jadi ketika A mengirim pesan, pihak ketiga dapat mengubahnya, mengenkripsinya kembali, dan mengirimnya ke B. Hal ini juga berlaku sebaliknya. Kelemahan dikarenakan pada metode *Diffie-Hellman Key Exchange*, tidak ditanya autentifikasi dari kedua pihak yang berkomunikasi. Agar lebih aman proses autentifikasi perlu diimplementasikan dalam proses *Diffie-Hellman Key Exchange*.

2.2. Tabel Hash Dinamik/Dynamic Hash Table (DHT)

Merupakan sistem terdistribusi yang memiliki nama dan nilai untuk setiap informasi, terstruktur dalam bentuk simpul-simpul. Simpul dapat mengambil suatu nilai yang diasosiasikan terhadap suatu nama tertentu.

DHT membentuk infrastruktur sistem yang kompleks dan dapat digunakan untuk pertukaran data antar simpul. Sistem ini dipakai pada BitTorrent.

Perkembangan DHT sebagian merupakan kontribusi dari peer-to-peer sistem seperti napster dan gnutella, dimana mereka mengambil keuntungan dari sistem ini untuk program aplikasinya. DHT mempunyai karakteristik sebagai berikut:

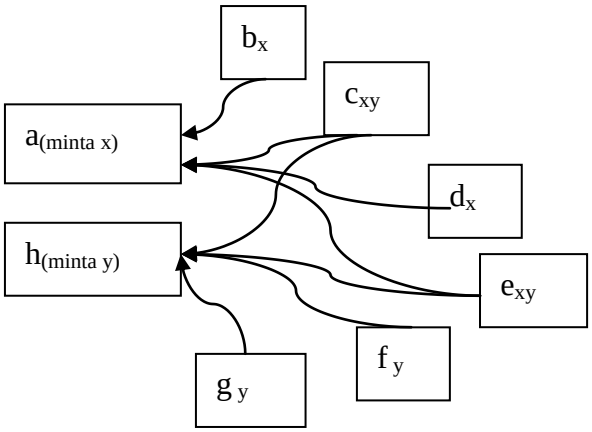
- Desentralisasi: Simpul-simpul membentuk infrastruktur sistem tanpa ada koordinasi terpusat.
- Skalabilitas: Sistem dapat bekerja secara efisien walaupun terdapat simpul yang sangat banyak.
- Toleransi kesalahan: Sistem harus dapat bekerja dengan baik walaupun simpul secara terus menerus bertambah dan berkurang.

Dalam BitTorrent, DHT digunakan sebagai pelacak distribusi untuk mempertemukan pengguna-pengguna yang ingin mempertukarkan file-file tertentu. Struktur DHT terdiri dari abstract keyspace dan overlay network.

Abstract keyspace membagi kunci-kunci kepada anggota-anggota simpul. Selanjutnya overlay network menghubungkan anggota simpul dengan kunci tertentu yang ingin membangun jaringan pertukaran.

Misal suatu jaringan dengan pengguna a,b,c,d,e,f,g,h, seperti yang terlihat dibawah, ingin membangun pertukaran data. Katakan “a” meminta suatu file dengan nama *file1* dan “h” meminta suatu file dengan nama *file2* maka DHT memberikan abstract keyspace melalui fungsi hash tertentu kepada tracker.

Contoh *file1 = x* dan *file2 = y*. *x* adalah kunci dari fungsi hash untuk *file1* dan y adalah kunci dari fungsi hash untuk *file2*. Overlay network lalu mengirim pesan kesuluruh simpul berdasar informasi dari tracker dan anggotanya yang mempunyai kunci x dan y.



Gambar 4. Pemisalan jaringan komunikasi.

Program BitTorrent mencari tidak hanya satu anggota yang mempunyai file yang sama sesuai dengan yang diminta, tetapi seluruh anggota yang memiliki file tersebut. Kemudian donor memberikan bagian dari file, dan donor lainnya memberikan bagian lainnya hingga lengkap.

Semakin banyak anggota yang memiliki file tersebut, maka semakin cepat file dapat didownload. Oleh karena kemungkinan anggota dapat mencapai jutaan, maka BitTorrent mencari kunci-kunci yang sama pada simpul terdekat terlebih dahulu. Overlay network kemudian membentuk jaringan yang terdekat dari tempat peminta.

2.3. RC4

RC4 didesain oleh Ron Rivest dari RSA Security pada tahun 1987. RC4 sendiri merupakan singkatan dari Rivest Cipher 4, meskipun ada pula yang sering menyebutnya Ron's Code 4. RC4 digunaka pula untuk protokol enkripsi pada wireless equivalent privacy (WEP) dan wi-fi protected access (WPA). RC4 banyak digunakan karena tergolong cepat dan sederhana dalam implementasi di hardware maupun software.

RC4 menghasilkan *pseudorandom stream of bits* yang jika digunakan dalam enkripsi digabungkan dengan teks biasa menggunakan XOR. Untuk menghasilkan keystream, cipher menggunakan keadaan internal khusus yang terdiri dari dua bagian:

1. Permutasi dari seluruh 256 byte yang mungkin.
2. Dua 8-bit penunjuk indeks.

Permutasi dimulai dengan mennggunakan *key-scheduling algorithm* (KSA). Setelah selesai bit-bit informasi dihasilkan menggunakan *pseudo-random generation algorithm* (PRGA).

Contoh key-scheduling algorithm:

```
for i from 0 to 255
    S[i] := i
endfor
j := 0
for i from 0 to 255
    j := (j + S[i] + key[i mod
keylength]) mod 256
    swap(S[i],S[j])
endfor
```

Gambar 5. Contoh KSA

Key-scheduling algorithm digunakan untuk memulai permutasi pada array "S". Kemudian diproses 256 kali iterasi mirip seperti algoritma utama PRGA tetapi juga digabungkan dengan kunci byte pada waktu yang sama.

Contoh pseudo-random generation algorithm:

```
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
```

Gambar 6. Contoh PRGA

Untuk setiap iterasi yang dibutuhkan, PRGA memodifikiasi keadaan dan output sebuah byte dari keystream. Pada setiap iterasi, penambahan satu I pada PRGA, menambah nilai S yang menunjuk pada i ke j, mengganti nilai S[i] dan S[j], dan nilai output di S pada lokasi S[i] + S[j] (modulo 256). Setiap nilai S ditukar paling tidak sekali setiap 256 iterasi.

III. ANALISIS DAN PEMBAHASAN

Client pada BitTorrent yang ingin berkomunikasi tanpa disadari melalui beberapa tahap protokol enkripsi berikut. Untuk saling menukar informasi, metode Diffie-Hellman perlu menghasilkan suatu kunci, agar tidak ada pihak ketiga yang mengetahui, atau passive listener. Masalah sebelumnya pada Diffie-Hellman adalah autentifikasi. Pada BitTorrent, setiap pengguna di internet tentu mempunyai suatu ip tersendiri, sehingga bila ada pihak ketiga yang mencoba mengubah Diffie-Hellman key akan terdeteksi. Tabel hash dinamik membantu mengamankan komunikasi ini dari *man-in-the-middle attack*.

BitTorrent mengkombinasikan pertukaran kunci Diffie-Hellman dengan abstract keyspace pada tabel hash dinamik untuk menjalin komunikasi. Kekuatan enkripsi berkisar antara 60-80 bits, untuk cipher simetris pada metode Diffie-Hellman.

Tahap selanjutnya adalah mengenkripsi data yang dikirim menggunakan metode RC4. PRGA dalam metode RC4 rentan terhadap serangan pada byte-byte pertamanya.untuk mencegah hal tersebut, kilobyte pertama pada pada keluaran RC4 dibuang. Dalam program BitTorrent sendiri, setiap pengguna juga dapat mengatur apakah enkripsi dilakukan pada header saja, atau seluruh data.

Pihak ISP mencoba untuk mencegah lalu lintas BitTorrent yang terlalu memakan kapasitas internet. Beberapa pendekatan yang dilakukan untuk mengganggu lalu lintas adalah dengan cara membajak komunikasi dari anggota ke tracker pada DHT, untuk mengetahui anggota tersebut berdasarkan alamat ip yang diberikan tracker. Setelah selanjutnya terjalin komunikasi, langsung diputus dari alamat ip tersebut oleh ISP dengan cara mengirimkan TCP reset palsu. Dengan demikian metode selanjutnya, yaitu enkripsi data dengan RC4 tidak dapat dilakukan dengan demikian komunikasi gagal.

Sekarang banyak internet service provider menggunakan berbagai metode untuk mendeteksi lalu lintas BitTorrent, seperti analisis pattern-timing, kategorisasi berbasis port, dan lainnya. Namun masih banyak ISP yang belum menerapkannya. Sehingga protokol enkripsi BitTorrent masih tergolong efektif.

IV. KESIMPULAN

1. BitTorrent menggunakan protokol enkripsi untuk mencegah serangan pada hubungan komunikasi antar penggunanya dengan metode Diffie-Hellman key exchange, tabel hash dinamik (DHT), dan RC4 cipher.
2. Metode Diffie-Hellman key exchange menggunakan metode kunci publik simetris.
3. Metode Diffie-Hellman digunakan pada protokol enkripsi untuk mencegah adanya passive listener.
4. Info tabel hash dinamik (DHT) digunakan untuk mencegah *man-in-the-middle attack.*
5. Metode RC4 cipher digunakan untuk mengenkripsi data yang dikirim mealui komunikasi BitTorrent.
6. Metode RC4 tetap digunakan walaupun tingkat keamanannya tergolong rendah untuk sekarang, dikarenakan waktu proses cpu yang cepat.

V. REFERENSI

[1]http://en.wikipedia.org/wiki/BitTorrent_protocol_encryption.html, Akses terakhir: 13 Desember 2010,pukul 21.30 WIB.

[2]http://en.wikipedia.org/wiki/RC4.html, Akses terakhir: 14 Desember 2010, pukul 19.10 WIB.

[3]http://en.wikipedia.org/wiki/Diffie-Hellman_key_exchange.html, Akses terakhir: 13 Desember 2010, pukul 19.30 WIB.

[4]http://en.wikipedia.org/wiki/Distributed_hash_table.html, Akses terakhir: 13 Desember 2010, pukul 20.20 WIB.

[5]http://www.wired.com/wired/archive/13.01/bittorrent.html, Akses terakhir: 14 Desember 2010, pukul 20.30 WIB.

[6]http://tools.ietf.org/html/rfc3526#section-8.html, Akses terakhir: 14 Desember 2010, pukul 21.20 WIB.

[7]http://www.rsa.com/rsalabs/node.asp?id=2248.html, Akses terakhir: 13 Desember 2010, pukul 21.33 WIB.

VI. PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 14 Desember 2010

M. Faizal Hitobeli - 13506057