Studi dan Analisis Teknik Keamanan dalam Lingkungan *Mobile System*

(Kriptografi dalam Kehidupan Sehari-hari)

Riani Rilanda – NIM : 13505051

Program Studi Teknik Informatika, Institut Teknologi Bandung
Jl. Ganesha 10, Bandung
E-mail : if11051@students.if.itb.ac.id

Abstrak

Security pada komputasi bergerak menjadi bagian yang sangat penting dimana berbagai *mobile device* saling berinteraksi melalui jaringan wireless. Hal ini semakin urgent mengikuti tingkat ukuran perangkat mobile yang semakin sederhana dan memiliki keterbatasan *power resource*. Pada jaringan *dedicated*, keamanan jaringan sudah berkembang, dengan keberadaan *mobile devices* menjadi sebuah tantangan baru dalam dunia securitas. Terdapat kemungkinan penerapan securitas pada wires system untuk system berdasarkan mobilitas devices.

Public Key Infrastructure sering digunakan sebagai *framework* keamanan yang kuat dan *reliable*, namun hal ini membutuhkan *central point* untuk kontrol jangkauan melalui intrnet untuk menjamin keamanan yang tinggi. Selain penerapan system keamanan ini, terdapat satu hal yang menjadi salah satu keterbatasan umum dalam *system mobile*, yaitu komunikasi *wireless* mempunyai kemungkinan kegagalan system kriptografi berbeda dengan yang sebagaimana yang diterapkan pada *wired system*, system mobilitas membutuhkan metode kriptografi yang kuat . Selain keterbatsan *lifetime battery*, CPU mini cenderung menyebabkan kesulitan dalam operasi system kriptrografi yang dibutuhkan dalam system mobile ini. Jika hal ini terjadi, maka akan menyebabkan *denial-of-service attacks*, yaitu penolakan *service* jaringan *wireless* karena kegagalan sistem kriptografi ini.

***Kata kunci**:* Keamanan, *Mobile system.*

1. Pendahuluan

Perkembangan teknologi jaringan selular broadband saat ini diiringi pertumbuhan berbagai macam aplikasi mobile yang berjalan di atas jaringan tersebut. Berbagai aplikasi tersebut tentunya membutuhkan sistem keamanan yang sangat handal, seperti pada contoh aplikasi *e-banking* dan *e-commerce.* Selain itu, dengan sifat perangkat yang *mobile* maka aplikasi *mobile* dapat digunakan kapan dan dimana pun. Oleh karena itu, sistem keamanannya perlu disesuaikan dengan kedinamisan sistem tersebut.

Aspek keamanan pada aplikasi *mobile* meliputi keamanan jaringan telekomunikasi nirkabel sebagai media transmisi data, sistem transaksi yang digunakan serta keamanan data yang disimpan pada perangkat *mobile*. Salah satu cara untuk meningkatkan keamanan ini adalah dengan menggunakan teknnologi kriptografi, yaitu antara lain dengan menggunakan enkripsi untuk mengacak data.

Salah satu metoda yang mulai umum digunakan adalah pengaman informasi dengan menggunakan publik key sistem. Dengan private ket sistem sebagai alternatif lainnya.

2. Konsep Keamanan

Keamanan memberikan layanan umum yang menjadi sebuah dasar kebutuhan sistem komputasi. Keamanan yang perlu dijamin adalah [1]:

- *Confidentiality* atau dengan kata lain informasi atau data hanya bisa diakses oleh pihak yang memiliki wewenang.
- *Integrity* adalah informasi atau data hanya dapat diubah oleh pihak yang berwenang.
- *Availability* adalah informasi hanya akan tersedia apabila pihak yang berwenang membutuhkan informasi tersebut.
- *Authentication* adalah pihak yang terlibat dengan pertukaran informasi dapat diidentifikasi dengan benar dan ada jaminan bahwa identitas yang didapat tidak palsu.
- *Non-repudiation* adalah pengirim maupun penerima informasi tidak dapat menyangkal pengiriman dan penerimaan pesan.

Selain keamanan yang perlu dijamin, terdapat pula berbagai macam serangan terhadap keamanan yang perlu dilihat, diantaranya :

- *Disclosure attacks* menyerang pada *data confidentiality* dan termasuk *passive attacks.* Dimana seseorang mendapat informasi rahasia tanpa secara aktif melakukan sesuatu. Biasanya serangan ini disebut *passive wiretapping* atau *sniffing.*
- *Unauthorised modification attack point* menyerang *data integrity*, hal ini terjadi apabila data dimodifikasi tanpa ada antitas yang memberikan perhatian. Serangan ini menyebabkan kesalahan data yang disebabkan o leh *active wiretapping (sniffing)*
- *Disruption attacks* meliputi *denial -of-service attack* pada data.
- *Usurpation* atau *unauthorised usages of service*, serangan ini mengenai *confidentiality, integrity dan availibility.*

Sedangkan serangan pada mekanisme *routing*, tingkat serangan pada metode *infrastructureless* sangat tinggi. Serangan tersebut diantaranya :

- *Black Hole attack* terjadi pada saat node-node menarik *routing path* yang kemudian menjatuhkan setiap paket yang melalui *routing path* ini.
- *Selfish nodes* dapat menghemat *resource* atau *power* namun menyebabkan *routing control packet* tapi tidak me-*routing*-kan paket data.
- *Resources consumption attacks*, *resources consumption* ini menghasilkan traffik yang tinggi yang menyebabkan *flooding* pada jaringan sehingga mengkonsumsi *bandwidth* yang sangat tinggi dan juga *power* yang tinggi pula.
- *Sibyl attack*, sebuah node dapat memalsukan identitas node sehingga menghasilkan tingkat kontrol melalui jaringan dan kemungkinan menyebabkan semua traffik diroutingkan ke node tersebut. Hal ini menyebabkan *dropping, sniffing* dan *delay packet* yang dikirimkan.

3. Konsep *Mobile System*

Pencapaian tujuan dari komputasi bergerak (berpindah-pindah) makin bertambah dengan mudah dipenuhi seiring dengan bertambahnya perangkat komputasi yang makin beragam. Namun bagaimanapun, masih terdapat beberapa kendala untuk mengatasi permasalahan ketika menyatukan antara perangkat yang siap pakai dan perangkat *embedded* ke dalam lingkungan komputasi bergerak. Yaitu termasuk dalam pendesaian perangkat yang cerdas berkolaborasi dengan perangkat yang lain, mudah digunakan dan dapat meningkatkan konektivitas diantara perangkat yang berbeda-beda. Untuk konektivitas tinggi, sistem keamanan dalam sistem menjadi faktor utama. Perangkat hanya dapat diakses oleh pemakai yang telah diberikan kuasa dan harus selalu menjaga keamanan komunikasi ketika menerima maupun mengirim informasi yang bersifat pribadi.

Dengan mengimplementasikan bentuk-bentuk pengaman komunikasi yang bersifat pribadi, sulit untuk menggunakan sarana kunci publik pada sembarang perangkat karena memerlukan algoritma kriptografi pada CPU. Algoritma kriptografi kunci publik yang umum digunakan misalnya RSA menggunakan 1024 bit, 43 ms untuk menandai dan 0.6 ms untuk memverifikasi, pada 200 MHz Intel Pentium Pro. Pada beberapa perangkat dengan 8 bit mikrokontroler dengan frekuensi 1-4 MHz, maka kriptografi kunci publik tidak dapat diterapkan pada perangkat ini.

Bagaimanapun kunci publik pada komunikasi antar perangkat dalam suatu jaringan masih sangat diminati.

3.1 Metode Pendekatan

Untuk penyediaan arsitektur model keamanan menggunakan kunci publik dalam jaringan dengan tetap menjaga kesederhanaan perangkat duibuatlah sebuah proxy perangkat lunak untuk masing-masing perangkat. Semua objek dalam sistem, sebagai contoh berbagai peralatan, gadget, software agents dan pengguna, mempercayakan pada proxy software yang dapat dijalankan pada embedded prosesor pada suatu peralatan atau sebuah komputer. Proxy software pada embedded prosesor pada suatu peralatan diasumsikan bahwa perangkat untuk komunikasi secara inheren, aman. Sebagai contoh, pada sebuah video kamera, software yang mengendalikan berbagai aktuator berjalan pada prosesor yang kuat dan proxy pada kamera dapat juga dijalankan pada embedded prosesor. Jika suatu perangkat memiliki kemampuan komputasi yang kecil komunikasi ke software proxy melalui sebuah jaringan dengan atau tanpa kabel. Komunikasi antar proxy menggunakan sebuah proxy yang aman ke protokol proxy berdasarkan SPKI SDSI. Memiliki dua perbedaan protokol yang memungkinkan kita menjalankan komputasi dengan protokol keamanan yang murah pada perangkat sederhana dan sebuah protokol yang rumit untuk sumber yang diberikan kuasa dan komunikasi pada perangkat yang lebih hebat.

3.2 Arsitektur Sistem

Sistem mempunyai tiga komponen utama, yaitu perangkat, proxy dan server. Sebuah perangkat merujuk pada berbagai tipe sumber jaringan yang dibagi baik software maupun hardware. Yang dapat berupa printer, kamera keamanan nirkabel, lampu atau software agent. Karena protokol komunikasi dan badwidth antar perangkat bertambah lebar, masing-masing perangkat memiliki sebuah proxy yang unik untuk dijadikan antarmuka dengan perangkat lain. Server menyediakan fasilitas penamaan dan pencarian untuk berbagai perangkat. Disini diasumsikan korespondensi satu-satu antara perangkat dan proxy, dan masing – masing user dilengkapi dengan K21, dan proxy berjalan pada sebuahkomputer sehingga sistem ini hanya memerlukan perangkat, proxy dan server.

3.3 Perangkat

Tiap perangkat, hardware atau software memiliki keterikatan denagn software proxy. Hardware misalnya, proxy mungkin berjalan pada embedded prosesor dalam perangkat tersebut, atau komputer yang ada dalam satu jaringan dengan perangkat tersebut. Sedang pada perangkat software, perangkat dapat memasukkan proxy software itu sendiri.

Tiap perangkat berkomunikasi dengan proxy masing-masing melalui protokol yang tepat pada suatu perangkat tertentu. Sebuah printer melalui sebuah Ethernet card dapat berkomunikasi dengan proxy menggunakan TCP/IP. Kamera nirkabel menggunakan sebuah protocol nirkabel untuk tujuan yang sama. K21 sebuah perangkat sederhana dengan prosesor ringan, berkomunikasi dengan proxy menggunakan perangkat khusus ke proxy protokol.

3.4 Proxy

Proxy adalah perangkat lunak software yang berjalan pada jaringan maya komputer. Fungsi utamanya adalah membuat keputusan kontrol akses perangkat yang diwakili. Proxy juga dapat berfungsi seperti skrip yang berjalan dari sebuah perangkat dan menjadi antarmuka dengan service yang dituju. Proxy menyediakan sebuah API yang sangat sederhana pada sebuah perangkat. Metode sendtoproxy() dipanggil oleh perangkat lunak untuk mengirimkan pesan ke proxy. Sedangkat metode senttodevice() dipanggil oleh proxy untuk mengirimkan pesan ke perangkat. Jika proxy menerima pesan dari proxy lain, bergantung pada pesannya, proxy akan menerjemahkannya ke dalam bentuk yang dapat dimengerti deang nmenggunakan suatu perangkat khusus. Kemudian meneruskannya ke perangkat yang dituju. Jika proxy menerima pesan dari suatu perangkat, pesan tersebut kemudian diterjemahkan menjadi bentuk umum yang dimengerti semua proxy, kemudian meneruksan ke proxy yang lain. Kapanpun proxy menerima pesan, sebelum menerjemahkan menyampaikan pesan ke suatu perangkat, proxy akan mengadakan kontrol akses terlebih dahulu.

3.5 Perangkat ke Protokol Proxy untuk Perangkat Nirkabel

Dalam hal ini, akan dianggap bahwa perangkat ringan mempunyai konektivitas jaringan nirkabel dengan bandwidth yang sempit, serta CPU yang lambat. Dan perangkat berat dengan bandwidth lebar dan CPU yang cepat. Diasumsikan perangkat berat dapat menjalankan proxy software secara lokal. Dengan proxy lokal, protokol yang rumit untuk komunikasi yang amat antara perangkta ke proxy tidaklah perlu, dengan asumsi bahwa bagian-bagian penting dari perangkat memiliki kemampuan perlawanan. Sedangkan untuk perangkat ringan, proxy harus berjalan dimanapun.

3.6 Komunikasi

Keamanan protokol prototipe layer sistem yang akan dijelaskan adalah melalui sebuah protokol frekuensi radio yang sederhana. Komunikasi frekuensi radio antara sebuah perangkat dengan proxy ditangani oleh sebuah gateway yang menerjemahkan paket-paket komunikasi frekuensi radio menjadi paket-paket UDP IP yang kemudian dirutekan oleh jaringan menuju proxy. Gatewat juga dapat melakukan penerjemahan paket-paket UDP IP menjadi paket-paket frekuensi radio dan menyampaikannya ke suatu perangkat.

3.7 Keamanan

Proxy dan perangkat komunikasi melalui sebuah kanal aman yang mengenkripsi dan mengautentikasi semua pesan. Algortima HMAC MD5 digunakan untuk autentikasi dan RC5 digunakan untuk enkripsi. Kedua algoritma ini menggunakan kunci asimetrik. Proxy dan perangkat berbagi 128 bit kunci.

3.7.1 Autentikasi

HMAC menghasilkan sebuah MAC yang dapat memvalidasi keautentikasian dan integritas dari sebuah pesan. HMAC menggunakan kunci rahasia sehingga hanya seseorang yang tahu kunci khusus dapat membuat MAC khusus pula atau memverifikasi MAC khusus tersebut dengan benar.

HMAC dan fungsi hash MD5 menghasilkan 16 byte MAC, 8 byte MSB diletakkan pada akhir setiap paket. Hal ini tentu saja membatasi jumlah data yang ditransmisikan. Terdapat kelemahan disini yaitu memungkinkan serangan untuk menebak bit yang lebih sedikit untuk menyerang MAC. Akan tetapi cara ini dapat diterima, karena jika ke 16 byte dimasukkan ke dalam tiap paket, maka tiap paket digunakan hanya untuk autentikasi alih-alih data yang ditransmisikan.

3.7.2 Enkripsi

Data dienkripsi menggunakan algoritma RC5. RC5 memiliki kesederhanaan dalam tampilan. Implementasi RC5 berdasarkan pada kode openSSL. RC5 adalah blok chiper, yaitu terbgadi menjadi 8 blok byte data. Akan tetapi, pengimplementasian ini denagn mode output feed back dapat juga dianggap sebagai stream chiper. Hal ini memungkinkan seorang arbiter mengenkripsi data tanpa mengkhawatirkan tentang blok data. Dengan menggunakan mode output feedback hanya dapat diperlukan enkripsi RC5. Mode output feedback menggenarasi sebuah blok enkripsi dari sebuah inisial vektor dan sebuah kunci. Blok enkripsi kemudian di XORkan dengan data untuk menghasilkan cipherteks.

4. Keamanan pada *Mobile System*

Keamanan yang dapat diterapkan pada sebuah sistem, termasuk *mobile system* diantaranya :

4.1 WLAN *security*

WLAN dikembangkan oleh para pionir pada tahun 1985. Berbagai penelitian dilakukan di laboratorium untuk membangun jaringan nirkabel yang menghubungkan berbagai macam peralatan dari komputer, mesin kas dan lain-lain.

Tahun 1997, lahirlah sebuah standar pertama yang dikenal dengan IEEE 802.11b dan disebut dengan *wireless fidelity (WiFi)* dengan frekuensi 2.4 GHz.

4.1.1 WEP

WEP merupakan sebuah algoritma enkripsi yang menggunakan *shared keys* pada proses autentikasi untuk memeriksa pengguna dan untuk mengeknripsi data yang dilewatkan pada segmen jaringan WLAN. WEP merupakan sebuah algoritma sederhana yang menggunakan pseudi-random number generator (PRNG) dan RC4 *stream chiper*. WEP dimaksudkan untuk tujuan keamanan yakni kerahasiaan data, mengatur hak akses dan integritas data.

4.1.2 WEP *Keys*

WEP *Keys* diimplementasikan pada *client* dan infrastruktur yang digunakan pada WLAN. WEP *Keys* ini merupakan *alphanumeric character string* yang memiliki 2 fungsi pada WLAN. Pertama, kunci ini dapat digunakan untuk verifikasi identitas pada stasiun autentikasi. Kedua, kunci ini dapat digunakan untk enkripsi data.

4.1.3 WEP *Usage*

Ketika WEP diinisialisasi, paket data akan dikirimkan dengan menggunakan WEP untuk mengenkripsi. Namun paket header data yang berisi MAC *address* tidak mengalami proses enkripsi. Semua layer 3 yang berisi *source* dan tujuan mengalami enkripsi.

4.1.4 AES

AES merupakan pengganti algoritma RC4 yang digunakan pada WEP. AES menggunakan algoritma Rijndale.

4.1.5 Filtering

Filtering merupakan mekanisme keamanan dasar yang digunakan untuk mendukung WEP atau AES. *Filtering* memiliki arti menutup semua hubungan yang tidak diijinkan dan membuka semua hubungan yang diijinkan. Pada WLAN, *filtering* memiliki tiga jenis yakni, SSID *filtering*, MAC *address filtering*, protocol *filtering.*

4.2 *Wireless* VPN

Keamanan WLAN menyediakan keamanan untuk WLAN di sekitar *access point.* Untuk jarak yang lebih jauh, jangkauan *access point* tidak dapat diaplikasikan. Untuk itu digunakan teknologi *Wireless* VPN (Virtual Public Network). Blok pembangun utama dari VPN adalah *tunneling*, keamanan, QoS, manajemen dan pengadaan. Tujuan dari *tunneling* adalah menciptakan jalur pada jaringan bersama dan membungkus komunikasi dengan header-header paket baru untuk pengirimannya. Tunnel tidak memberikan kerahasiaan, seperti yang diberikan, enkripsi tunnel bisa terjadi pada layer 2 maupun layer 3.

Untuk keamanan VPN, harus ada otorisasi pengguna, otentikasi dan enkripsi data. Otentikasi awal digunakan untuk memverifikasi pengguna/*router* dan mengijinkan dilakukannya tindakan-tindakan tertentu serta menolak tindakan yang lainnya. *Tunneling* VPN biasanya bisa memberikan perlindungan yang cukup, tapi ada beberapa lalulintas yang memerlukan enkripsi IPSec.

4.3 *Mobile* IP

Biasanya penentuan IP *address node* bergantung pada *network link* yang dalam hal ini terhubung. Ini artinya, jika node dirubah ke jaringan lain, IP *address* juga akan berubah. Pada mobile IP, tiap pengguna memiliki *home address* yang diasosiasikan ke mobile node ketika berada pada *home network*. Pada *home network*, juga terdapat entity yang disebut home agent yang lalu lintas jaringannya diforward ke mobile node ketika tidak berada pada *home network*. *Forward* lalulintas jaringan yang lain disebut *foreign agent*. *Foreign agent* ini kemudian diketahui, bagaimana untuk mencapai mobile node. Ketika mobile node mengirim paket ke mesin yang lain, mobile node dapat mengirrim pesan-pesan tersebut langsung melalui *foreign agent* atau juga langsung dihubungkan melalui *home agent.*

VPN dan mobile IP terlihat hampir sama. Keduanya dapat menyediakan keamanan hubungan yang mungkin terjadi ke *home network* dan juga transparansi lokasinya. Bagaimanapun, dalam VPN aspek utama adalah keamanan atau aspek privasi. Sedangkan pada mobile IP, poin utamanya adalah kelayakan dari mobile IP ke internet protokol. Mobile IP dapat menggunakan IPSec untuk mendukung enkripsi data pada layer *network*, maka dari itu disinilah keamanan dapat dicapai. Sayangnya tidak tersedia implementasi yang *up-to-date* dari mobile IP. Kebanyakan adalah hanya sekedar prototipe dan tidak dipergunakan untuk harian. Lainnya bersifat komersil dan tidak disediakan yang gratis.

4.4 Kriptografi *add-on*

Terdapat beberapa simpanan data dan aplikasi yang tersedia yang menyediakan kriptografi. Kriptografi *add-on* memperluas sistem dengan beberapa properti dari kriptografi. SSL/TLS adalah contoh dari simpanan data yang menyediakan keamanan *end-to-end* pada layer network.

SSL merupakan salah satu metode enkripsi dalam komunikasi data yang dibuat oleh Netscape Communication Corporation dimana SSL adalah protocol berlapis. Dalam setiap lapisnya terdiri atas panjang, deskripsi dan isi. SSL mengambil data untuk dikirimkan,

dipecahkan ke dalam blok-blok yang teratur kemudian dikompres jika perlu, menerapkan MAC, dienkripsi dan hasilnya dikirimkan. Di tempat tujuan, data dienkripsi, verifikasi, dekompres dan disusun kembali. Hasilnya dikirimkan ke klien atasnya. SSL hanya mengirimkan data yang dikirimkan melalui http.

5. Kesimpulan

Pada penggunaan energi secara, pembelajaran mengenai algoritma kriptografi dan protokol diberikan. Kebutuhan atas pekerjaan tersebut disebut *better gap*. Komunikasi, daya komputasi, kebutuhan keamanan dan juga kebutuhan energi meningkat secara konstan, sedangkan kapasitas baterai semakin kecil atau sama dengan sebelumnya.

Kita akan melihat bahwa terdapat pertukaran yang tidak sebanding antara kebutuhan daya dan keamanan pada enkripsi dan autentifikasi. Kriptografi asimetrik lebih banyak proses komputasi daripada kriptografi simetrik. Seperti yang sudah kita ketahui dan karena itu, kunci publik dari kriptografi biasanya digunakan pada pendekatan hibrid untuk mendapatkan persetujuan mengakses pada kunci simetris. Fitur ini dapat dieksploitasi dengan cara menyimpan *session key* dan menggunakan itu kembali untuk mencegah negoisasi ulang dari kunci simetris menggunakan *expensive asymmetric operation.* Bagaimanapun juga ini dapat menghasilkan keamanan yang lebih lemah, karena tiap *session key* dimungkinkan untuk hilang atau tertangkap. Jadi ini juga termasuk pertukaran yang tidak sebanding antara *session life time* dan keamanan. Lagi pula, operasi kriptografi dimungkinkan berubah secara dinamis sesuai dengan kebutuhan keamanan dalam rangka untuk menghemat daya. Operasi kriptografi juga dapat dikurangi ketika operasi berada pada lingkungan yang dapat dipercaya atau untuk lebih mudahnya, ketika komunikasi tidak berada pada keamanan yang layak.

Oleh karena itu beberapa protokol, untuk secara langsung SSL/TSL dan IPsec menyediakan banyak keuntungan, keduanya mendukung pemilihan algoritma dengan parameter yang dapat digunakan untuk operasi kriptografi selama perubahan penggunaan kunci. Jadi kurang aman tetapi lebih banyak algoritma dapat di pilih berdasarkan jenis keamanan dan ketersediaan sumber daya. Demikian halnya dengan parameter algoritma yang lain, yang pada umumnya telah ditetapkan, dapat dimodifikasi. Algoritma enkripsi yang simetrik beroperasi dengan menggunakan sejumlah angka tertentu dalam suatu eksekusi algoritma iteratif. Pengurangan hitungan iterasi akan membuat algoritma kurang aman akan tetapi energi yang digunakan akan lebih efisien. Berbeda dengan hal diatas, kriptografi kurva ekliptic kadang lebih efisien dibandingkan dengan algoritma RSA. Sebagai contoh sebuah algoritma yang digunakan untuk membuat tanda tangan dan proses verifikasi dalam kriptografi ini lebih seimbang dibanding RSA-Signature. Juga selama pengubahan kunci, algoritma kurva ekliptic yang berdasar pada algoritma Diffie -Hellman mengkonsumsi lebih sedikit daya daripada algoritma Diffie -Hellman yang normal. Hal ini mungkin akan sama dengan faktanya, bahwa kriptografi kurva ekliptic membutuhkan bit kunci yang lebih kecil untuk menyediakan tingkatan yang sama dalam hal pengamanan daripada algoritma yang biasa digunakan. Jika algoritma asimetric AES menyediakan kelengkapan keamanan yang baik dan dianggap baik untuk analisa kri ptografi.

DAFTAR PUSTAKA

[1] Jay, Ramachandran. *Designing Security Architecture Solution.* John Wiley and Sons. 2002.

[2] http://josh.staff.ugm.ac.id/blog/activity/ Tanggal akses : 8 Mei 2009.

[3] Sinambela, Josua. *Network Security*. http://72.14.235.132/search?q=cache:DoSJH4Nwfv8J:elista.akprind.ac.id/staff/catur/Keamanan%2520Jaringan/07-Keamanan%2520Sistem%2520Jaringan%2520Komputer.pdf+kriptografi+pada+proxy&cd=17&hl=id&ct=clnk&gl=id&client=firefox-a. Tanggal akses : 8 Mei 2009.