# IPsec: Aplikasi Teknik Kriptografi untuk Keamanan Jaringan Komputer

Angga Danimartiawan, Raiza Mahardika N, Wahyu Satryo Nugroho

*Departemen Teknik Informatika*
*Institut Teknologi Bandung*
*Jalan Ganesha 10 Bandung 40132*

*E-mail:*
*if11080@students.if.itb.ac.id, if11048@students.if.itb.ac.id, if11077@students.if.itb.ac.id*

## Abstraksi

Keamanan pada komunikasi melalui jaringan komputer sekarang telah menjadi sebuah persoalan penting. Teknik kriptografi diimplementasikan pada protokol komunikasi IPsec untuk mendapatkan aspek keamanan tersebut. IPsec merupakan serangkaian protokol komunikasi yang menerapkan beberapa teknik kriptografi untuk menjamin keamanan dalam komunikasi melalui jaringan komputer. IPsec merupakan solusi yang transparan terhadap pengguna karena pengguna tidak perlu menyadari keberadaannya karena IPsec membungkus paket-paket IP dengan *header* yang pada akhirnya ditransmisikan sebagai paket-paket IP biasa. Protokol *Authentication Header* (AH) menjamin *data integrity*, sedang protokol *Encapsulating Security Payload* (ESP) selain menjamin *data integrity* juga menjamin data *confidentiality*. IPsec bukanlah protokol komunikasi yang sempurna. Kompleksitasnya yang tinggi, dokumentasinya yang belum sempurna untuk sebuah standar, fitur yang tidak perlu (sehingga menambah kompleksitas) merupakan beberapa kelemahan yang dimilikinya. Tetapi hingga kini, IPsec masih dianggap sebagai protokol keamanan yang paling baik dibanding protokol keamanan IP yang lain.

*Kata kunci: IPsec, keamanan jaringan komputer, teknik kriptografi, ESP, AH,* confidentiality, integrity

## 1. Pendahuluan

Pada awal perkembangannya, jaringan komputer digunakan hanya untuk pengiriman *e-mail* antar perguruan tinggi untuk keperluan riset dan untuk berbagi penggunaan *printer* dalam suatu perusahaan. Untuk memenuhi tujuan tersebut, aspek keamanan jaringan pada saat itu tidak mendapat perhatian penting. Namun kini, saat jaringan komputer juga telah digunakan untuk berbagai aktivitas perbankan dan perdagangan, terutama melalui Internet, aspek keamanan menjadi masalah yang harus mendapat perhatian besar.

Kriptografi merupakan ilmu untuk menyamarkan suatu pesan demi menjaga kerahasiaannya. Suatu pesan (*plain text*) harus melalui proses enkripsi terlebih dulu menjadi bentuk yang tidak berarti (*cipher text*) sebelum dikirimkan ke penerima yang berhak. Hanya pihak yang berhak lah yang dapat melakukan proses dekripsi, yaitu mengubah kembali *cipher text* menjadi *plain text* memakai suatu kunci yang rahasia. *Plain text* akan sulit diturunkan dari *cipher text*-nya oleh orang yang tidak berhak (yang tidak memiliki kunci yang bersesuaian dengan *cipher text* tersebut). Kriptografi menganut prinsip kerahasiaan melalui ketidakjelasan (*secrecy through obscurity*).

Dalam makalah ini, IPsec akan dibahas sebagai salah satu aplikasi teknik kriptografi untuk keamanan jaringan komputer. Aspek keamanan yang disediakan merupakan hasil dari teknik kriptografi yang diimplementasikan dalam rangkaian protokol IPsec. Sejak pendefinisiannya dalam beberapa dokumen RFC (*Request for Comments*), para ahli telah melakukan beberapa analisis untuk mengidentifikasi kelebihan dan kelemahan IPsec, serta memberikan beberapa rekomendasi untuk perbaikan.

## 2. Keamanan Jaringan Komputer

Masalah keamanan jaringan komputer secara umum dapat dibagi menjadi empat kategori yang saling berkaitan [17]:

1. *Secrecy*/*Confidentiality*
   Informasi yang dikirimkan melalui jaringan komputer harus dijaga sedemikian rupa kerahasiaannya sehingga tidak dapat diketahui oleh pihak yang tidak berhak mengetahui informasi tersebut.
2. *Authentication*
   Identifikasi terhadap pihak-pihak yang sedang melakukan komunikasi melalui jaringan harus dapat dilakukan. Pihak yang berkomunikasi melalui jaringan harus dapat memastikan bahwa pihak lain yang diajak berkomunikasi adalah benar-benar pihak yang dikehendaki.
3. *Nonrepudiation*
   Pembuktian korespondensi antara pihak yang mengirimkan suatu informasi dengan informasi yang dikirimkan juga perlu dilakukan dalam komunikasi melalui jaringan komputer. Dengan pembuktian tersebut, identitas pengirim suatu informasi dapat dipastikan dan penyangkalan pihak tersebut atas informasi yang telah dikirimnya tidak dapat dilakukan.
4. *Integrity Control*
   Informasi yang diterima oleh pihak penerima harus sama dengan informasi yang telah dikirim oleh pihak pengirim. Informasi yang telah mengalami perubahan dalam proses pengiriman, misalnya diubah oleh pihak lain, harus dapat diketahui oleh pihak penerima.

Dalam protokol *stack* OSI (*Open Systems Interconnection*) *Reference Model* terdapat beberapa kemungkinan penempatan aspek keamanan jaringan. Terdapat pula kemungkinan bahwa aspek keamanan jaringan tidak hanya ditempatkan pada salah satu *layer* melainkan dikombinasikan pada beberapa *layer* sekaligus karena penempatan pada tiap *layer* memiliki keunggulan masing-masing.

Pada *physical layer*, kabel transmisi dapat diamankan dengan penggunaan tabung pelapis yang berisi gas bertekanan tinggi. Pada *data link layer*, paket pada jalur *point-to-point* dapat dienkripsi ketika meninggalkan sebuah mesin dan didekripsi ketika masuk ke mesin yang lain. Pada *network layer*, penggunaan *firewall* dan protokol IPsec digunakan untuk menjamin keamanan. Pada *transport layer*, koneksi dapat dienkripsi untuk menjamin kemanan antarproses (*end-to-end*). Terakhir, pada *application layer*, aspek autentikasi dan *nonrepudiation* dapat dijamin dengan algoritma pada aplikasi yang digunakan.

## 3. IPsec (*IP Security*)

Sebagaimana telah dijelaskan sebelumnya, masalah utama yang menjadi perhatian dalam mengimplementasikan aspek keamanan dalam jaringan komputer adalah di *layer* mana aspek keamanan tersebut harus diimplementasikan.

Salah satu solusi yang menjamin tingkat keamanan paling tinggi adalah dengan mengimplementasikan aspek keamanan pada *application layer*. Dengan implementasi aspek keamanan pada *layer* ini maka keamanan data dapat dijamin secara *end-to-end* (proses ke proses) sehingga upaya apa pun untuk mengakses atau mengubah data dalam proses pengiriman data dapat dicegah. Namun, pendekatan ini membawa pengaruh yang besar yaitu bahwa semua aplikasi yang dibangun harus ditambahkan dengan aspek keamanan untuk dapat menjamin keamanan pengiriman data.

Pendekatan lain didasarkan bahwa tidak semua pengguna menyadari pentingnya aspek keamanan sehingga mungkin menyebabkan mereka tidak dapat menggunakan fitur keamanan pada aplikasi dengan benar. Selain itu, tidak semua pengembang aplikasi memiliki kemauan untuk menambahkan aspek keamanan pada aplikasi mereka. Oleh karena itu, aspek keamanan ditambahkan pada *network layer* sehingga fitur kemanan dapat dipenuhi tanpa campur tangan pengguna atau pengembang aplikasi.

Pada akhirnya pendekatan kedua mendapat dukungan lebih banyak daripada pendekatan pertama sehingga dibuat sebuah standar keamanan *network layer* yang salah satu

desainnya yaitu IPsec. IPsec merupakan kumpulan protokol yang dikembangkan oleh IETF (*Internet Engineering Task Force*) untuk mendukung pertukaran paket yang aman melalui IP *layer*.

IPsec didesain untuk menyediakan keamanan berbasis kriptografi yang memiliki karakteristik *interoperable* dan berkualitas. Layanan keamanan yang disediakan mencakup *access control*, *connectionless integrity*, *data origin authentication*, proteksi dari *replay attack* (*sequence integrity*), *data confidentiality* dan *traffic flow confidentiality*. Layanan tersebut disediakan pada IP *layer* sehingga mendukung proteksi untuk IP *layer* dan *layer* lain di atasnya [7, 18)].

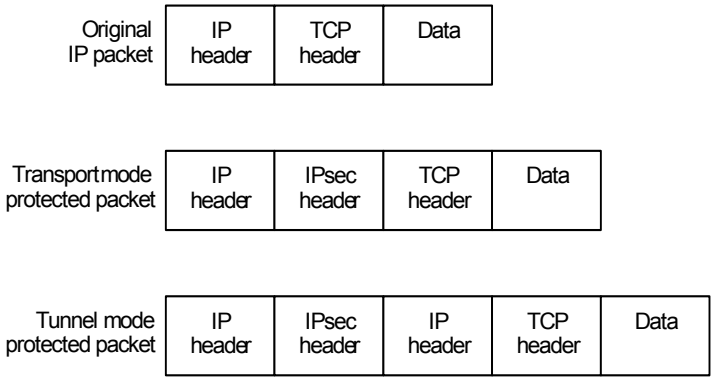
Secara teknis, IPsec terdiri atas dua bagian utama. Bagian pertama mendeskripsikan dua protokol untuk penambahan *header* pada paket yang membawa *security identifier*, data mengenai *integrity control*, dan informasi keamanan lain. Bagian kedua berkaitan dengan protokol pembangkitan dan distribusi kunci.

Bagian pertama IPsec adalah implementasi dua protokol keamanan yaitu:

1. *Authentication Header* (AH) [8)] menyediakan *data integrity*, *data origin authentication* dan proteksi terhadap *replay attack*.
2. *Encapsulating Security Payload* (ESP) [9)] menyediakan layanan yang disediakan oleh AH ditambah layanan data *confidentiality* dan *traffic flow confidentiality*.

Kedua protokol di atas dapat diaplikasikan masing-masing atau secara bersamaan untuk menyediakan layanan keamanan yang dibutuhkan pada IPv4 dan IPv6. Setiap protokol mendukung dua mode penggunaan: *transport mode* dan *tunnel mode*.

Pada *transport mode*, protokol menyediakan proteksi terhadap layer di atas IP *layer*. Layanan keamanan pada mode ini dilakukan dengan penambahan sebuah IPsec *header* antara IP *header* dengan *header* protokol *layer* di atas IP yang diproteksi. Sedangkan pada *tunnel mode*, protokol diaplikasikan untuk menyediakan proteksi pada paket IP sehingga sekaligus melindungi *layer* di atas IP *layer*. Hal ini dilakukan dengan mengenkapsulasi paket IP yang akan diproteksi pada sebuah IP datagram yang lain. Penambahan *header* pada kedua mode protokol dapat dilihat pada Gambar 1 [3)].




**Gambar 1 *Transport* dan *Tunnel Mode* Protokol IPsec**

Sedang bagian kedua IPsec adalah implementasi protokol IKE (*Internet Key Exchange*) [6)] yang berfungsi dalam pembangkitan dan pertukaran *criptographic key* secara otomatis. *Criptographic key* digunakan dalam autentikasi *node* yang berkomunikasi dan proses enkripsi dan dekripsi paket yang dikirimkan. IKE tidak dibahas secara detil pada makalah ini.

Sebuah implementasi IPsec beroperasi pada sebuah *host* atau *security gateway* untuk menyediakan proteksi pada *traffic* IP. Proteksi yang ditawarkan berdasarkan kebutuhan didefinisikan melalui *Security Policy Database* (SPD) yang ditetapkan dan dipelihara oleh administrator sistem. Berdasarkan SPD tersebut, paket-paket diproses melalui salah satu dari tiga mode pemrosesan berdasarkan informasi pada *header* paket, yaitu [3)]:

1. Paket diberikan layanan keamanan IPsec (*apply security*).
2. Paket tidak diberikan layanan IPsec dan dibuang (*drop*/*discard*).
3. Paket diperbolehkan melewati protokol keamanan IPsec tanpa memberikan layanan keamanan IPsec (*bypass*).

## 4. *Data Integrity* dan *Authentication* pada IPsec

Integritas data yang dikirimkan melalui jaringan komunikasi dijamin oleh IPsec melalui metode autentikasi *digital signature* atas informasi yang dikirimkan secara paket-per-paket. Layanan jaminan integritas data ini disediakan dengan menggunakan algoritma HMAC (*Hash Message Authentication Code*) baik oleh protokol AH (*Authentication Header*) maupun oleh protokol ESP (*Encapsulating Security Payload*).

### 4.1. *Hash Message Authentication Code* (HMAC)

Untuk menjamin integritas paket, IPsec menggunakan *Hash Message Authentication Code* (HMAC) [10]. HMAC adalah algoritma autentikasi menggunakan kunci rahasia. Integritas data dan autentikasi asal data yang disediakan oleh HMAC bergantung pada penyebaran kunci rahasia yang digunakan. Jika hanya sumber (pengirim) dan tujuan (penerima) yang mengetahui kunci HMAC, maka autentikasi asal data dan integritas data untuk paket-paket yang dikirim antara kedua pihak tersebut dijamin.

HMAC menggunakan fungsi *hash* satu arah, H, dan kunci rahasia K. Beberapa fungsi *hash* yang digunakan di antaranya adalah: MD5 dan SHA-1. Untuk memperjelas fungsi *hash* yang digunakan, digunakan notasi HMAC-H. Contohnya, HMAC-MD5 menyatakan HMAC yang menggunakan fungsi *hash* MD5. H merupakan fungsi *hash* yang melakukan *hashing* dengan iterasi suatu fungsi kompresi pada blok-blok data. Panjang blok data dalam byte, B (B=64 untuk MD5 dan SHA-1), dan panjang *output hash* dalam byte, L (L=16 untuk MD5, L=20 untuk SHA-1), masing-masing menjadi batas atas dan batas bawah panjang kunci K. Bila panjang K melebihi B, maka yang digunakan sebagai kunci adalah H(K).

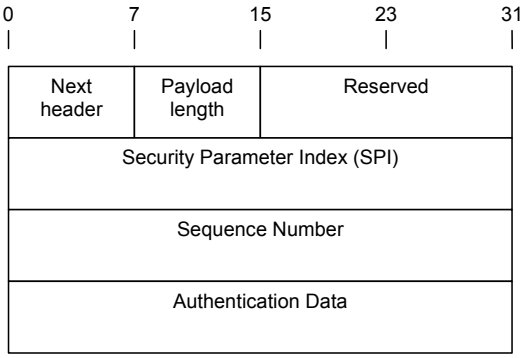
Menghitung HMAC sebuah data 'text' berarti melakukan perhitungan H(K XOR opad, H(K XOR ipad, text)), dengan ipad = byte 0x36 diulangi sebanyak B kali, dan opad = byte 0x5C diulangi sebanyak B kali. Algoritma HMAC adalah sebagai berikut:

1. *Append* nol pada akhir K untuk membuat *string* sepanjang B byte.
2. Lakukan XOR string sepanjang B byte hasil langkah 1 dengan ipad.
3. *Append* data 'text' dengan *string* sepanjang B byte hasil langkah 2.
4. Terapkan fungsi H pada hasil langkah 3.
5. XOR *string* sepanjang B byte yang dihasilkan langkah 1 dengan opad.
6. *Append* hasil H dari langkah 4 dengan *string* sepanjang B byte hasil langkah 5.
7. Terapkan fungsi H pada hasil langkah 6 dan kembalikan hasilnya sebagai HMAC 'text'.

Salah satu praktek kriptografi menyangkut HMAC adalah hanya menggunakan sebagian bit (sepanjang t bit) paling kiri dari keluaran algoritma HMAC. Notasi yang digunakan untuk menyatakan penerapan praktek ini yaitu HMAC-H-t. Contohnya, HMAC-MD5-96 menyatakan HMAC yang menggunakan fungsi *hash* MD5 dan hanya menggunakan 96 bit paling kiri dari hasil keluaran algoritma HMAC. Penggunaan sebagian keluaran ini mempunyai keuntungan yaitu lebih sedikit informasi *hash* paket yang bisa didapat oleh penyerang.

### 4.2. *Authentication Header* (AH)

AH didefinisikan sebagai protokol IP yang diberi nomor protokol 51. Dengan demikian, *field* protokol pada *header* paket IP yang diproteksi akan memiliki nilai 51 yang menunjukkan bahwa *header* yang mengikutinya adalah sebuah *header* AH. *Field* pada *header* AH diperlihatkan pada Gambar 2 [3, 8].




**Gambar 2 *Header* AH**

Pesan berisi informasi yang akan dikirim melalui jaringan akan dipecah-pecah menjadi paket-paket IP. Hal yang pertama dilakukan oleh IPsec dalam pemrosesan paket tersebut adalah memeriksa apakah paket tersebut akan diberikan layanan keamanan dengan protokol AH. Hal ini dilakukan dengan mencocokan paket yang bersangkutan dengan entri pada *Security Policy Database* (SPD).

Apabila terdapat entri pada SPD, IPsec akan menentukan sebuah *Security Association* (SA) berisi informasi mengenai algoritma MAC (*Message Authentication Code*) yang digunakan untuk paket tersebut sesuai dengan *source* dan *destination* paket yang bersangkutan. Setiap pasangan *source* dan *destination* memiliki Security Association yang berbeda yang ditentukan secara manual sebelumnya maupun secara otomatis melalui protokol IKE (*Internet Key Exchange*).

Informasi mengenai algoritma MAC yang digunakan dalam SA ditambahkan pada *header* AH yaitu pada *field* `SPI` (`Security Parameter Index`) setelah sebelumnya memberi nilai pada *field* `Next header`, `Payload length` dan `Reserved`. *Field* `Next header` diberi nilai numerik dari tipe protokol dari data yang diproteksi (misalnya TCP atau UDP pada *transport mode*, atau IP pada *tunnel mode*). Selain itu, *field* `Payload length` diberi nilai sesuai panjang *header* AH, kemudian *field* `Reserved` diberi nilai kosong (*null value*).

*Field* `Sequence number` pada awalnya diberi nilai kosong, untuk kemudian ditambahkan satu-per-satu untuk setiap paket terproteksi yang dikirimkan. `Sequence number` ini tidak dapat berulang sehingga apabila sudah mencapai nilai maksimum $2^{32}$ maka SA untuk pengiriman pesan yang berjalan harus dimatikan terlebih dahulu dan dibentuk SA baru untuk pengiriman pesan lanjutan. Hal ini dimaksudkan untuk mencegah terjadinya *replay attack* atas paket pesan.

*Field* `Authentication data` adalah *field* yang panjangnya variabel berisi hasil dari fungsi pengecekan integritas yang disebut ICV (*Integrity Check Value*). Protokol AH pada IPsec tidak mendefinisikan (algoritma) *authenticator* tertentu tetapi mengharuskan implementasi *authenticator* untuk menjamin interoperabilitas antara implementasi IPsec yang berbeda-beda. Dua *authenticator* yang harus diimplementasikan adalah HMAC-SHA-1-96 [13] dan HMAC-MD5-96 [14]. Kedua fungsi merupakan fungsi MAC dengan kunci rahasia yang keluarannya di-*truncate* menjadi 96 bit .

ICV dihitung dengan menjalankan fungsi yang didefinisikan oleh *authenticator* pada SA dengan parameter kunci rahasia dan seluruh paket IP yang terproteksi termasuk *header* AH. Namun demikian, *field-field* pada *header* IP yang sifatnya berubah-ubah seperti TTL (*Time To Live*) dan *header checksum* serta *field* `Authentication data` itu sendiri diberi nilai kosong terlebih dahulu sebelum dimasukkan ke dalam fungsi. Setelah nilai ICV tersebut didapat maka nilainya dimasukkan dalam *field* `Authentication data`. Nilai-nilai *field* dari *header* IP yang berubah juga dimasukkan kembali ke dalam *field* masing-masing.

Pemrosesan AH selesai sampai titik ini dan paket IP yang terproteksi siap untuk dikirimkan. Bergantung pada ukuran paket yang dihasilkan, paket tersebut mungkin terfragmentasi dalam perjalanannya. Hal ini tidak menjadi masalah dan akan ditangani oleh pihak penerima.

Pihak penerima pesan harus melakukan *reassembly* atas paket yang terfragmentasi. Setelah paket utuh diterima, hal pertama yang dilakukan adalah menentukan SA yang digunakan untuk memproteksi paket tersebut. SA yang digunakan ditentukan melalui *field*

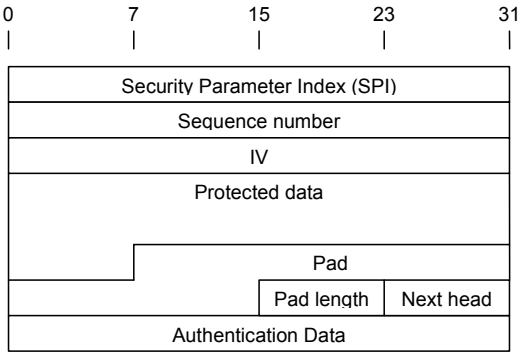
`Destination` dan *field* `Protocol` pada header IP serta *field* `SPI` pada *header* AH. Bila tidak ditemukan SA yang sesuai, maka paket tersebut di-*discard*.

Kemudian, pengecekan `Sequence number` dilakukan. Sama halnya dengan sebelumnya, apabila paket yang diperiksa gagal memenuhinya maka paket tersebut di-*discard*. Paket yang gagal memenuhi pengecekan `Sequence number` menunjukkan bahwa paket terproteksi tersebut telah di-*replay* oleh pihak lain.

Proses yang teakhir dilakukan adalah pengecekan ICV. ICV yang terdapat pada *field* `Authentication data` pada *header* AH disimpan dan *field* tersebut diberi nilai kosong. *Field* pada *header* IP yang berubah-ubah juga diberi nilai kosong. Setelah itu, algoritma *authenticator* yang didefinisikan SA dijalankan pada paket tersebut dan hasilnya dibandingkan dengan ICV yang disimpan sebelumnya. Jika hasilnya sama maka paket tersebut terautentikasi dan paket IP yang diproteksi dapat di-*restore* dan diperlakukan sebagaimana paket IP biasanya.

## 4.3. *Encapsulating Security Payload* (ESP)

ESP didefinisikan sebagai protokol IP yang diberi nomor protokol 50. Dengan demikian, *field* protokol pada *header* paket IP yang diproteksi akan memiliki nilai 50 yang menunjukkan bahwa *header* yang mengikutinya adalah sebuah *header* ESP. *Field* pada *header* ESP diperlihatkan pada Gambar 3 [3, 9].




**Gambar 3 *Header* ESP**

Sebagaimana halnya pada protokol AH, paket yang akan diproses harus ditentukan terlebih dahulu apakah akan diberi layanan keamanan IPsec. Serupa dengan yang dilakukan pada AH, informasi paket yang diterima dicocokan pada entri di SPD (*Security Policy Database*). Apabila paket tersebut terdapat pada entri SPD maka paket tersebut akan diberikan layanan keamanan ESP. Sebuah *Security Association* (SA) kemudian akan ditentukan dari definisi yang ditentukan secara manual sebelumnya maupun secara otomatis dengan protokol IKE. Namun demikian, berbeda dengan protokol AH yang hanya menentukan *authenticator* (algoritma autentikasi yang digunakan), SA pada protokol ESP juga menentukan sebuah *encryptor* (algoritma enkripsi paket yang digunakan). Hal ini disebabkan ESP tidak hanya menyediakan fitur *data integrity* tetapi juga fitur data *confidentiality*. Fitur *data confidentiality* akan dijelaskan pada bagian selanjutnya.

Nilai numerik informasi mengenai *Security Association* (SA) yang telah ditentukan kemudian dimasukkan ke dalam *field* `Security Parameter Index` (`SPI`) sebagaimana dilakukan pada protokol AH. Demikian pula dengan *field* `Sequence number` akan diinisiasi dengan nilai nol dan di-*increment* setiap pemrosesan paket pesan yang diproteksi. Seperti pada protokol AH, *field* ini diperlukan untuk untuk menjamin keamanan pengiriman paket dari *replay-attack*.

*Field* `IV` (`Initialization Vector`), `Protected data`, `Pad`, dan `Pad length` akan berisi nilai hasil pemrosesan *data confidentiality*. *Field-field* ini akan dijelaskan pada bagian berikutnya. Untuk sementara, *field-field* tersebut diasumsikan telah diisi nilai sebagaimana mestinya.

Kemudian, *field* `Next header` diberi nilai sesuai dengan nilai numerik protokol dari *payload* paket yang terproteksi. Dengan demikian, apabila ESP digunakan dalam *transport mode* dengan *payload* berupa paket TCP maka akan memiliki nilai 6 dan bila digunakan dalam *tunnel mode* (*payload* berupa paket IP) maka akan memiliki nilai 4.

Terakhir, sebagaimana dalam protokol AH, *field* `Authentication data` akan berisi nilai yang digunakan dalam pengecekan integritas paket

atau ICV (*Integrity Check Value*). Seperti halnya pada protokol AH, *authenticator* yang harus diimplementasi dalam protokol ESP adalah HMAC-MD5-96 dan HMAC-SHA-96 untuk menjamin interoperabilitas antar implementasi IPsec yang berbeda-beda. Semua *field* dari *field* `SPI` hingga *field* `Next header` kemudian dimasukkan ke dalam fungsi *hash* yang telah ditentukan pada SA dan hasilnya dimasukkan ke dalam *field* `Authentication data`.

Sampai di sini pemrosesan fitur *data integrity* dari paket protokol ESP selesai dijalankan. Paket kemudian siap untuk dikirimkan melalui saluran komunikasi. Fragmentasi paket juga mungkin terjadi sebagaimana pada paket AH dan akan ditangani pula oleh pihak penerima paket tersebut.

Pihak penerima pesan harus melakukan *reassembly* atas paket yang terfragmentasi. Setelah paket utuh diterima, hal pertama yang dilakukan adalah menentukan SA yang digunakan untuk memproteksi paket tersebut. SA yang digunakan ditentukan melalui *field* `Destination` dan *field* `Protocol` pada *header* IP serta *field* `SPI` pada *header* ESP. Bila tidak ditemukan SA yang sesuai, maka paket tersebut di-*discard*.

Kemudian seperti pada protokol AH, pengecekan `Sequence number` dilakukan. Apabila paket yang diperiksa gagal memenuhinya maka paket tersebut di-*discard*. Paket yang gagal memenuhi pengecekan `Sequence number` menunjukkan bahwa paket terproteksi tersebut telah di-*replay* oleh pihak lain.

Proses berikutnya yang dilakukan adalah pengecekan ICV. ICV yang terdapat pada *field* `Authentication data` pada *header* ESP. Setelah itu, algoritma *authenticator* yang didefinisikan SA dijalankan pada paket tersebut dari *field* `SPI` hingga *field* `Next header` dan hasilnya dibandingkan dengan ICV yang disimpan sebelumnya. Jika hasilnya sama maka paket tersebut terautentikasi.

Pemrosesan *data integrity* untuk paket yang diterima selesai sampai di sini dan akan dilanjutkan pada pemrosesan *data confidentiality* yang akan dijelaskan pada bagian selanjutnya.

## 5. *Data Confidentiality* pada IPsec

Fitur *data confidentiality* pada IPsec hanya disediakan oleh protokol ESP dengan menggunakan algoritma kriptografi simetri. Protokol AH tidak menyediakan fitur ini walaupun sama-sama menyediakan fitur *data integrity*. Aspek ini lah yang menyebabkan pembagian protokol IPsec ke dalam dua jenis (AH dan ESP). Hal ini bertujuan untuk menyediakan fleksibilitas bagi pengguna untuk dapat memilih tingkat keamanan yang dikehendaki karena tidak semua pesan bersifat rahasia tetapi integritas harus selalu dijaga. Bila pesan tidak bersifat rahasia maka pengguna dapat menggunakan protokol AH dan bila pesan harus dijamin kerahasiaanya maka pengguna dapat memilih protokol ESP.

### 5.1. Algoritma Kriptografi Simetri pada IPsec

*Confidentiality* pada IPsec disediakan melalui implementasi algoritma kriptografi simetri yang digunakan dalam enkripsi dan dekripsi paket. RFC 2451 [18)] menerangkan penggunaan algoritma Blowfish, CAST-128, 3DES, IDEA, RC5 dalam mode CBC (*Cipher Block Chaining*) dengan ESP. RFC 2451 tesebut juga menyatakan pada prinsipnya, algoritma *block cipher* yang lain juga dapat digunakan. IPsec dirancang tidak dengan mendefinisikan algorima enkripsi yang harus digunakan, tetapi hanya mengharuskan implementasi algoritma DES-CBC dan NULL untuk menjamin interoperabilitas. Beberapa algoritma selain yang telah disebut yang pernah dipakai dalam IPsec adalah AES-CBC dan RC4 [1)].

### 5.2. *Encapsulating Security Payload* (ESP)

Pada pembangkitan paket protokol ESP, pemrosesan *data confidentiality* dilakukan setelah penentuan *Security Association* (SA) dan sebelum pemrosesan *data integrity*. Sebagaimana telah

dijelaskan pada bagian sebelumnya, *encryptor* yang mendeskripsikan algoritma yang digunakan dalam pengenkripsian paket ditentukan oleh *Security Association* (SA). Seperti halnya *authenticator*, IPsec tidak mendefinisikan *encryptor* yang digunakan tetapi mengharuskan implementasi DES-CBC dengan *Initilization Vector* eksplisit [12)] untuk menjamin interoperabilitas dengan implementasi IPsec lainya. Namun demikian, DES terbukti telah dapat dipatahkan sehingga implementasi dan penggunaan algoritma enkripsi lain yang lebih kuat lebih disarankan. ESP juga mendukung algoritma enkripsi NULL [5)], yaitu fungsi enkripsi yang identik dengan fungsi identitas, NULL(m)=I(m)=m. Algoritma NULL ini disediakan untuk memberi pilihan pemakaian protokol ESP tanpa memperhatikan aspek *confidentiality*.

Setelah informasi yang diperlukan seperti algoritma enkripsi, dan kunci enkripsi ditentukan oleh *Security Association* (SA) dan nilainya telah dimasukkan dalam *field* `Security Parameter Index` (`SPI`). Maka paket yang akan diproteksi akan dienkripsi dengan *authenticator* yang telah ditentukan dengan parameter kunci yang sesuai. Jika digunakan dalam *transport mode* maka pengenkripsian akan dilakukan pada paket *layer* di atas IP (UDP atau TCP) dan jika digunakan dalam *tunnel mode* maka pengenkripsian akan dilakukan pada paket IP.

Sebagaimana telah disebut di atas, *authenticator* DES-CBC mengharuskan *Initialization Vector* (IV) secara eksplisit disimpan. Dengan demikian, bila menggunakan authenticator ini, sebuah nilai IV diambil dari 8 *octet* pertama paket yang akan dienkripsi dan nilainya dimasukkan ke dalam *field* `IV` (`Initialization Vector`). *Field* `IV` ini tidak akan disertakan dalam *field-field* yang dienkripsi karena akan digunakan sebagai salah satu parameter dalam dekripsi.

*Field* `Protected data` kemudian akan diisi dengan paket yang terproteksi. Dalam proses tunnel mode maka akan diisi dengan paket IP sedangkan dalam *transport mode* maka akan diisi dengan paket UDP atau TCP. Beberapa algoritma *authenticator* mengharuskan ukuran blok paket yang akan dienkripsi berupa kelipatan dari blok tertentu. Oleh karena itu, *field* `Pad` disediakan untuk memenuhi tujuan tersebut dan diisi dengan nilai *padding* yang tergantung pada *authenticator* yang digunakan. Ukuran *padding* yang ditambahkan dimasukkan dalam *field* `Pad length`.

Bila semua *field* pada *header* ESP telah terisi maka proses enkripsi dapat dilakukan. Enkripsi akan dilakukan terhadap *field* `Protected data` hingga *field* `Next header` dengan menggunakan algoritma yang telah ditentukan dan dengan parameter *Initialization Vector* bila diperlukan dan kunci yang juga telah ditentukan sebelumnya. Hasil dari proses enkripsi akan dimasukkan antara *field* `IV` dan *field* `Authentication data` menggantikan *field-field* yang menjadi masukan proses enkripsi.

Sampai di sini, pemrosesan *data confidentiality* dari paket ESP selesai dijalankan dan akan dilanjutkan dengan pemrosesan *data integrity* sebagaimana telah dijelaskan pada bagian sebelumnya.

Pada saat penerimaan paket, pemrosesan *data confidentiality* dilakukan setelah pemrosesan *data integrity*. Paket yang terautentikasi kemudian diserahkan pada *decryptor*. Berdasarkan informasi pada `SPI` (`Security Parameter Index`), *decryptor* mengetahui algoritma apa yang digunakan untuk mendekripsi paket beserta kuncinya. Proses dekripsi dilakukan pada potongan paket yang terletak antara *field* `IV` dan *field* `Authentication data`. Dengan parameter *Initialization Vector* yang diambil dari *field* `IV` dan kunci yang didapat dari *Security Association* (SA) berdasarkan SPI maka bagian yang terenkripsi dapat dikembalikan ke bentuknya semula.

Berdasarkan informasi pada *field* `Pad length`, paket yang terproteksi (paket IP, TCP, atau UDP) dapat dikembalikan ke bentuknya semula dengan mengurangi ukurannya sebanyak *padding* yang dilakukan. Paket yang telah didekripsi tersebut siap untuk diproses lebih lanjut dan diperlakukan sebagaimana paket biasanya.

## 6. Analisis terhadap IPsec

Beberapa ahli telah melakukan analisis terhadap IPsec. Analisis yang dilakukan merupakan usaha identifikasi kelebihan dan kelemahan IPsec (yang di antaranya menghasilkan beberapa rekomendasi untuk perbaikan).

Kelebihan IPsec:

1. IPsec dapat melindungi protokol apa pun yang berjalan di atas IP dan pada medium apa pun yang dapat digunakan IP, sehingga IPsec merupakan suatu metode umum yang dapat menyediakan keamanan komunikasi melalui jaringan komputer [11)].
2. IPsec menyediakan keamanan secara transparan, sehingga dari sisi aplikasi, *user* tidak perlu menyadari keberadaannya [2, 11)].
3. IPsec dirancang untuk memenuhi standar baru IPv6 tanpa melupakan IPv4 yang sekarang digunakan [2, 3)].
4. Perancangan IPsec tidak mengharuskan penggunaan algoritma enkripsi atau hash tertentu sehingga jika algoritma yang sering digunakan sekarang telah dipecahkan, fungsinya dapat diganti dengan algoritma lain yang lebih sulit dipecahkan [2)].

Kelemahan IPsec:

1. IPsec terlalu kompleks, penyediaan beberapa fitur tambahan dengan menambah kompleksitas yang tidak perlu [4)].
2. Beberapa dokumentasinya masih mengandung beberapa kesalahan, tidak menjelaskan beberapa penjelasan esensial, dan ambigu [4)].
3. Beberapa algoritma *default* yang digunakan dalam IPsec telah dapat dipecahkan/dianggap tidak aman (misalnya DES yang dianggap tidak aman dan MD5 yang telah mulai berhasil diserang [3)]). Algoritma penggantinya telah tersedia dan administrator sistem sendiri yang harus memastikan bahwa mereka menggunakan algoritma lain untuk mendapatkan keamanan yang lebih tinggi [3)].

Beberapa rekomendasi yang dihasilkan Ferguson dan Schneier [4)] untuk perbaikan IPsec:

1. Hilangkan *transport mode*. Dari sisi keamanan, fungsionalitas *tunnel mode* merupakan *superset* dari fungsionalitas *transport mode*.
2. Hilangkan protokol AH. Dengan tidak perlunya *transport mode*, maka protokol AH juga dapat dihilangkan karena fungsinya dan kelebihannya (*overhead bandwidth* lebih kecil) dapat digantikan dengan modifikasi minor pada protokol ESP dalam *tunnel mode*.
3. Modifikasi protokol ESP sehingga selalu menyediakan fitur autentikasi, hanya enkripsi yang opsional. Saat ini autentikasi dan enkripsi pada ESP bersifat opsional (dengan adanya algoritma NULL).
4. Modifikasi protokol ESP sehingga semua data (termasuk kunci dekripsi) yang digunakan dalam dekripsi paket terautentikasi.

Meskipun memiliki beberapa kekurangan, IPsec masih dianggap sebagai protokol keamanan yang paling baik untuk memperoleh keamanan dalam komunikasi melalui jaringan komputer bila dibandingkan dengan protokol keamanan IP yang lain seperti Microsoft PPTP dan L2TP [4)].

## 7. Kesimpulan

Kesimpulan yang bisa diambil:

1. Aspek keamanan dalam komunikasi melalui jaringan komputer menjadi semakin penting terutama karena banyaknya aktivitas pertukaran informasi rahasia melalui Internet.
2. Keamanan jaringan terbagi menjadi empat kategori umum, yaitu:
   a. *Secrecy*/*Confidentiality*
   b. *Authentication*
   c. *Nonrepudiation*
   d. *Integrity Control*
3. IPsec merupakan salah satu solusi kemanan jaringan berupa protokol kemanan yang berada di *network layer* untuk pengiriman paket IP.
4. IPsec terdiri atas dua bagian utama, yaitu:
   a. Protokol penambahan *header* pada paket IP (AH dan ESP)
   b. Protokol pembangkitan dan distribusi kunci secara otomatis (IKE)
5. IPsec menggunakan teknik-teknik kriptografi dalam menyediakan layanan keamanan

*Authentication, Data Integrity*, dan *Confidentiality*.

6. *Authentication* dan *Data Integrity* disediakan oleh protokol AH dan ESP dengan menggunakan HMAC.
7. *Confidentiality* disediakan oleh protokol ESP dengan mengunakan algoritma kriptografi simetri.
8. Walaupun menurut para ahli masih memiliki beberapa kekurangan, IPsec masih dianggap sebagai solusi terbaik dalam menyediakan keamanan dalam komunikasi melalui jaringan komputer.

## 8. Referensi


[1] Alshamsi, AbdelNasir dan Takamichi Saito, *A Technical Comparison of IPsec and SSL*, Tokyo University of Technology, 2004, http://eprint.iacr.org/2004/314.pdf, 9 Januari 2005 08:00

[2] Dahlgren, Anders dan Oskar Jönsson, *IPsec, the Future of Network Security?*, Göteborg University, 2000, http://www.handels.gu.se/epc/archive/00002483/01/dahlgrenjonsson.pdf, 9 Januari 2005 08:00

[3] Doraswamy, Naganand dan Dan Harkins, *IPsec: The New Security Standard for the Internet, Intranet, and Virtual Private Networks*, Prentice-Hall, 1999

[4] Ferguson, Neil dan Bruce Schneier, *A Cryptographic Evaluation of IPsec*. Counterpane Internet Security, Inc., 1999, http://www.schneier.com/paper-ipsec.pdf, 9 Januari 2005 08:00

[5] Glenn, Rob dan Stephen Kent, *RFC 2410: The NULL Encryption Algorithm and Its Use with IPsec,* The Internet Society: Network Working Group, 1998, http://www.ietf.org/rfc/rfc2410.txt, 9 Januari 2005 08:00

[6] Harkins, Dan dan Dave Carrel, *RFC 2409: The Internet Key Exchange (IKE)*, The Internet Society: Network Working Group, 1998, http://www.ietf.org/rfc/rfc2409.txt, 9 Januari 2005 08:00

[7] Kent, Stephen dan Randall Atkinson, *RFC 2401: Security Architecture for the Internet Protocol*, The Internet Society: Network Working Group, 1998, http://www.ietf.org/rfc/rfc2401.txt, 9 Januari 2005 08:00

[8] Kent, Stephen dan Randall Atkinson, *RFC 2402: IP Authentication Header*, The Internet Society: Network Working Group, 1998, http://www.ietf.org/rfc/rfc2402.txt, 9 Januari 2005 08:00

[9] Kent, Stephen dan Randall Atkinson, *RFC 2406: IP Encapsulating Security Payload (ESP)*, The Internet Society: Network Working Group, 1998, http://www.ietf.org/rfc/rfc2406.txt, 9 Januari 2005 08:00

[10] Krawczyk, Hugo, et al, *RFC 2104: HMAC: Keyed-Hashing for Message Authentication*, The Internet Society: Network Working Group, 1997, http://www.ietf.org/rfc/rfc2104.txt, 9 Januari 2005 08:00

[11] Lee, Jin W., *Introduction of IPsec*, Arizona State University, 2002, http://rts-lab.eas.asu.edu/document/Introduction%20of%20IPsec.pdf, 9 Januari 2005 08:00

[12] Madson, Cheryl dan Naganand Doraswamy, *RFC 2405: The ESP DES-CBC Cipher Algorithm with Explicit IV*, The Internet Society: Network Working Group, 1998, http://www.ietf.org/rfc/rfc2405.txt, 9 Januari 2005 08:00

[13] Madson, Cheryl dan Rob Glen, *RFC 2403: The Use of HMAC-MD5-96 within ESP and AH*, The Internet Society: Network Working Group, 1998, http://www.ietf.org/rfc/rfc2403.txt, 9 Januari 2005 08:00

[14] Madson, Cheryl dan Rob Glen, *RFC 2404: The Use of HMAC-SHA-1-96 within ESP and AH*, The Internet Society: Network Working Group, 1998, http://www.ietf.org/rfc/rfc2404.txt, 9 Januari 2005 08:00

[15] Nedeltchev, Plamen dan Radoslav Ratchkov, *IPsec-based VPNs and Related Algorithms*, 2002, http://www.cisco.com/warp/public/784/packet/apr02/pdfs/plamen.pdf, 9 Januari 2005 08:00

[16] Pereira, Roy dan Rob Adams, *RFC 2451: The ESP CBC-Mode Cipher Algorithms*, The Internet Society: Network Working Group, 1998, http://www.ietf.org/rfc/rfc2451.txt, 9 Januari 2005 08:00
[17] Tanenbaum, Andrew S., *Computer Networks, Fourth Edition*, Prentice-Hall, 2003
[18] Thayer, Rodney, et al, *RFC 2411: IP Security Document Roadmap*, The Internet Society: Network Working Group, 1998, http://www.ietf.org/rfc/rfc2411.txt, 9 Januari 2005 08:00