BAB 2

DASAR TEORI SISTEM KEAMANAN MOBILE WIMAX

2.1. Pengenalan WiMax

Worldwide Interoperability for Microwave Access (WiMAX) merupakan evolusi dari teknologi *Broadband Wireless Access* (BWA) sebelumnya. Bila teknologi BWA sebelumnya masih *proprietary* atau berdasarkan hak cipta, maka teknologi WiMAX bersifat standard terbuka. Dalam arti komunikasi perangkat WiMAX diantara beberapa vendor yang berbeda tetap dapat dilakukan.

Pengembangan teknologi WiMAX terjadi dalam beberapa tahap atau mengalami evolusi. Sesuai dengan standarisasinya, dikatakan bahwa teknologi WiMAX diatur dalam standard IEEE 802.16. Standard ini terbagi lagi dalam beberapa kategori yaitu IEEE 802.16a yaitu untuk standard BWA yang belum open standard atau biasa disebut dengan Pre-WiMAX. Selanjutnya standard ini dikembangkan lagi menjadi standard IEEE 802.16d untuk WiMAX area tetap atau nomadik. Sementara untuk WiMAX bergerak yang disebut Mobile WiMAX akan diatur dalam standarisasi IEEE 802.16e yang telah diratifikasi pada akhir tahun 2005.

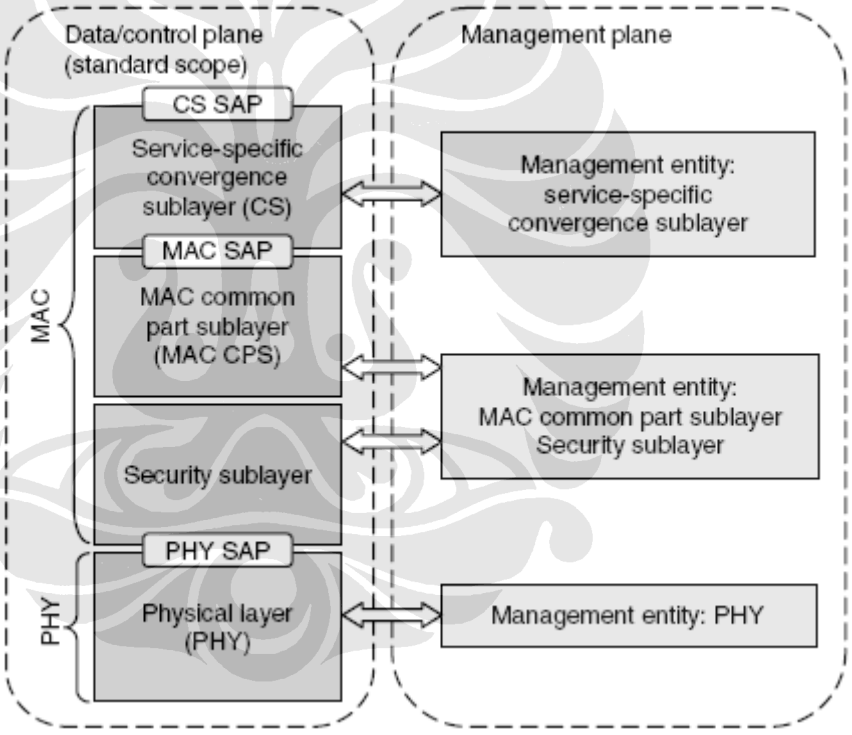
Disamping evolusi pada sisi kemampuan akses, terjadi juga evolusi pada sisi perangkat untuk pelanggan atau *Customer Premises Equipment* (CPE). Pada tahap awal, perangkat CPE WiMAX berupa perangkat untuk area tetap luar ruangan, kemudian berkembang menjadi area tetap dalam ruangan, portabel (nomadik) dan *mobile*. Perangkat untuk area tetap luar ruangan merupakan perangkat CPE terdiri dari 2 unit yaitu unit luar ruangan yang terdiri dari radio dan antena serta unit dalam ruangan yang merupakan antarmuka ke terminal pelanggan. Pada tipe area tetap dalam ruangan, perangkat CPE hanya terdiri dari satu unit perangkat dalam ruangan yang sudah terdiri dari radio, antena dan port antarmuka pengguna. Umumnya pada tipe ini, pengguna dapat menmasang sendiri perangkat CPE-nya (*self installation*).

Tahap berikutnya, perangkat CPE sudah bukan merupakan perangkat independen tetapi tergabung dalam terminal pelanggan seperti laptop dan PDA.

Pada tahap ini, CPE WiMAX portabel telah terpasang permanen pada terminal sebagaimana CPE Wi-Fi. Terakhir adalah perangkat bergerak. Keunggulan yang ditambahkan adalah kemampuan portabilitas yang lebih tinggi selain ukuran terminal yang lebih ringkas.

2.2. Lapisan Protokol WiMax

Pada WiMax/802.16 ditetapkan 2 lapisan protokol, yaitu lapisan fisik (PHY) dan lapisan *medium acces control* (MAC). Lapisan MAC mengatur masalah koneksi, Qos dan keamanan. Sedangkan lapisan PHY menangani ketersambungan sinyal dan koreksi kesalahan.



Gambar 2.1 Lapisan Protokol WiMax[1]

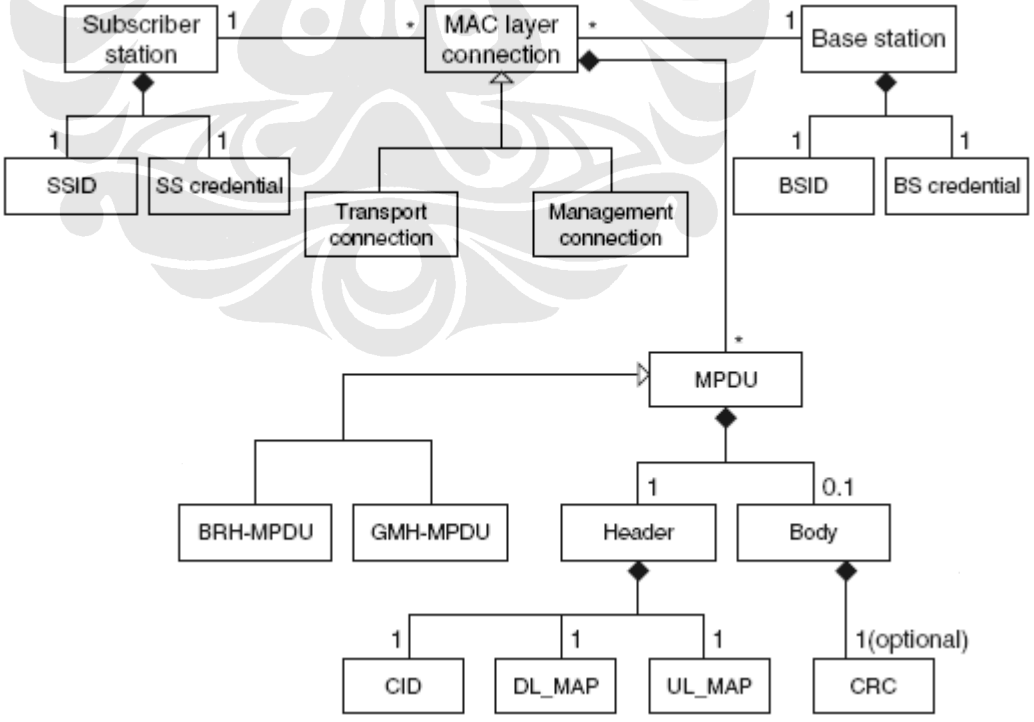
Dari lapisan MAC tersebut dapat dibagi lagi menjadi tiga sub-lapisan. Yang pertama sub-lapisan keamanan (*security sublayer*) yang berperan dalam proses autentikasi, pembentukan kunci keamanan dan enkripsi. Yang kedua adalah sub-lapisan MAC bagian umum (*MAC common part sublayer*). Dan yang ketiga adalah sub-lapisan kovergensi layanan khusus (*service-specific convergence*

sublayer). Sub Lapisan ini berhubungan langsung dengan lapisan diatasnya yang lebih tinggi.

2.3. Aspek Jaringan

Setiap stasiun pelanggan atau SS (*subscriber station*) berkomunikasi dengan stasiun utama atau BS (*base station*) melalui sambungan nirkabel. Sebelum tersambung, SS memindai daftar frekuensi-frekuensinya untuk mencari sinyal dari BS. Kemudian megobservasi trafik dari BS tersebut untuk menentukan parameter-parameter seperti perjangkaan, modulasi, koreksi kesalahan dan daya. Kemudian yang terakhir mengidentifikasi *timeslot* yang akan digunakan untuk melakukan *initial request*.

Rangkain paket-paket *initial* (*ranging request*) tersebut memperbaiki pengaturan perjangkaan dan daya, serta membentuk reservasi koneksi dengan menentukan profil celah waktu (*timeslot)* dan identitas koneksi (CID). BS memberikan bermacam CID yang berbeda kepada SS untuk tiap managemen, koneksi data dan kualitas layanan atau *Quality of Service* (QoS) yang berbeda.

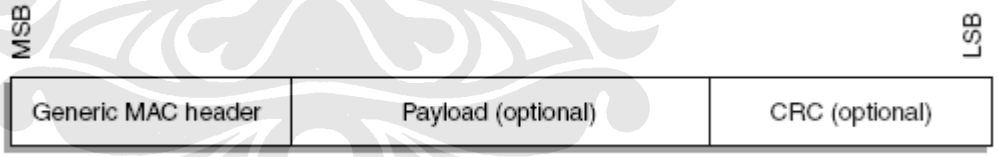


Gambar 2.2 Arsitektur Jaringan WiMax

Kominikasi antara SS dan BS dibagi menjadi kerangka-kerangka. Kerangka dari BS menuju SS disebut *downlink frames*. Sedangkan kerangka dari SS menuju BS disebut *uplink frames*. Kerangka-kerangka tersebut terdiri berisi tajuk kerangka (*frame header*) dan tubuh kerangka (*body)*. Tajuk dari tiap-tiap kerangka tersebut memiliki dua jenis pemetaan atau *map* yang menggambarkan penggunaan celah beserta penempatannya, yaitu *downlink map* (DL_MAP) dan *uplink map* (UL_MAP). Dan tiap-tiap celah tersebut merupakan bagian dari beberapa koneksi yang diidentifikasi oleh CID. [1]

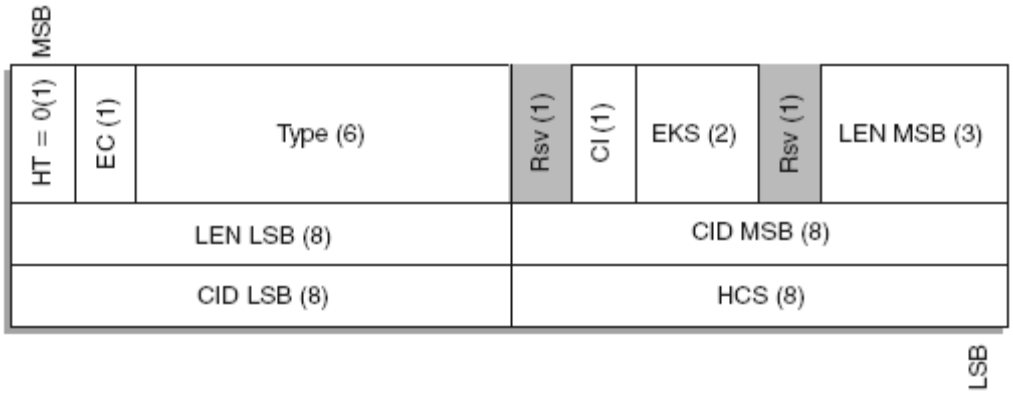
2.4. MAC Protocol Data Unit (MPDU)

Tiap-tiap *protocol data unit* (PDU) terdiri atas *generic MAC headaer* (GMH), *payload,* dan cyclic redundancy check (CRC) yang bersifat opsional. GMH menunjukkan kandungan dari payload dan dimulai dari most significant bit (MSB). Payload itu sendiri berisi nol atau lebih subheader dan MAC *service data unit* (SDU). Panjang dari payload tersebut dapat bermacam-macam. CRC bersifat opsional pada PHY layer tipe SC, namun merupakan keharusan untuk PHY layer tipe SCa, OFDM, dan OFDMA. [1]



Gambar 2.3 MAC PDU

Terdapat dua format yang ditetapkan untuk MAC *header*. Yaitu GMH yang yang digunakan untuk MAC PDU yang berisi pesan manajemen MAC atau data sub-lapisan konvergen. Berikutnya adalah *Bandwidth Request Header* (BRH) yang digunakan ketika melakukan permintaan bandwidth tambahan. Kedua jenis *header* tersebut dibedakan oleh satu bit "*header type*" (HT). Dimana berisi 0 untuk generic header dan berisi satu untuk bandwidth request header.

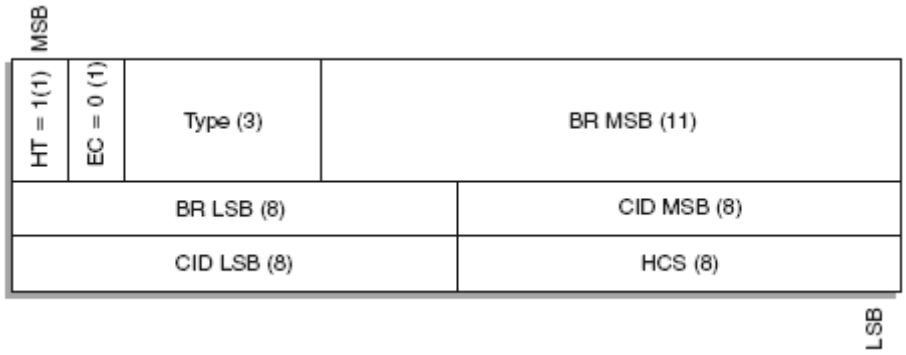


Gambar 2.4 Generic MAC Header

Pada Gambar2.4, merupakan MAC *header* jenis GMH. Hal ini ditunjukkan pada field HT yang berisi "0". Panjang dari GMH ini adalah 6 byte dan terdiri atas 12 field. Dimana isi dari masing-masing field tersebut akan dijelaskan tabel berikut :

Tabel 2.1 Field-field GMH

Name	Length (Bits)	Description
CI	1	CRC indicator 1 = CRC is included in the PDU by appending it to the payload after encryption if any 0 = No CRC is included
CID	16	Connection identifier
EC	1	Encryption control 0 = Payload is not encrypted 1 = Payload is encrypted
EKS	2	Encryption key sequence The index of the traffic encryption key (TEK) and initialization vector used to encrypt the payload. This field is only meaningful if the EC field is set to 1
HCS	8	Header check sequence An 8-bit field used to detect errors in the header
HT	1	Header type Shall be set to zero
LEN	11	Length The length in bytes of the MAC PDU including the MAC header and the CRC if present
Type	6	This field indicates the subheaders and special payload types present in the message payload



Gambar 2.5 Struktur BRH PDU

Sedangkan PDU tipe *bandwidth request* tidak memiliki *payload*, melainkan hanya terdiri atas *header* saja. Panjang dari *header* tersebut adalah sama dengan GMH yaitu 6 byte, namun hanya terdiri atas 8 field. Dimana isi dari masing-masing field tersebut akan dijelaskan tabel berikut :

Tabel 2.2 Field-field BRH

Bandwidth Request Header Fields

Name	Length (Bits)	Description
BR	19	Bandwidth request The number of bytes of uplink bandwidth requested by the subscriber station. The bandwidth request is for the CID. The request shall not include any PHY overhead
CID	16	Connection identifier
EC	1	Always set to zero
HCS	8	Header check sequence An 8-bit field used to detect errors in the header
HT	1	Header type = 1
Type	3	Indicates the type of bandwidth request header

2.5. Security Association (SA)

Security association (SA) merupakan seperangkat metode kriptografi dan asosiasi penyandian yang terdiri dari informasi tentang bagaimana penerapan suatu algoritma, bagaimana menggunakan penyandian dan lainsebagainya.

Setiap pelayanan memerlukan asosiasi keamanan. Untuk stasiun pelanggan (SS) menggunakan *traffic encryption* (TEK) *state machine* untuk setiap asosiasi keamanan. TEK akan bertanggung jawab untuk memanajemen enkripsi lalu lintas data pada setiap pelayanan. Subscriber station (SS) akan mengirimkan key request

ke base station (BS), dan base station (BS) akan mengirimkan jawaban secara random private key ke subscriber station (SS). Kunci ini dienkripsi menggunakan 3DES selama proses authorization.

Setelah dienkripsi menggunakan kunci private, maka semua data dienkripsi dengan algoritma kunci simetrik. Spesifikasi yang digunakan adalah 56-bit DES dalam mode cyclic block chaining(CBC). Ada tiga tipe security association (SA) yaitu primary, static dan dynamic. Setiap subscriber station (SS) menentukan sebuah primary security association (SA) dengan base station (BS) selama proses inisialisasi. Static security association ditentukan oleh base station (BS), sedangkan dynamic security association ditentukan dan diselesaikan oleh setiap permulaan dan akhir layanan selama proses koneksi. Setiap subscriber station (SS) mempunyai nomor yang unik dan eklusif, tetapi semua tipe static dan dynamic dapat di-share dengan multiple subscriber station. Subscriber station (SS) bertanggung jawab untuk menanyakan kepada base station (BS) untuk substansi yang baru sebelum waktunya habis pada base station(BS). Protokol PKM juga bertanggung jawab terhadap sinkronisasi antara subscriber station (SS) dan base station (BS).

2.6. Kriptografi

Kriptografi berasal dari dua kata Yunani, yaitu *Crypto* yang berarti rahasia dan *Grapho* yang berarti menulis. Secara umum kriptografi dapat diartikan sebagai ilmu dan seni penyandian yang bertujuan untuk menjaga keamanan dan kerahasiaan suatu pesan. Kriptografi pada dasarnya sudah dikenal sejak lama. Menurut catatan sejarah, kriptografi sudah digunakan oleh bangsa Mesir sejak 4000 tahun yang lalu oleh raja-raja Mesir pada saat perang untuk mengirimkan pesan rahasia kepada panglima perangnya melalui kurir-kurinya. Orang yang melakukan penyandian ini disebut *kriptografer*, sedangkan orang yang mendalami ilmu dan seni dalam membuka atau memecahkan suatu algoritma kriptografi tanpa harus mengetahui kuncinya disebut *kriptanalis*.

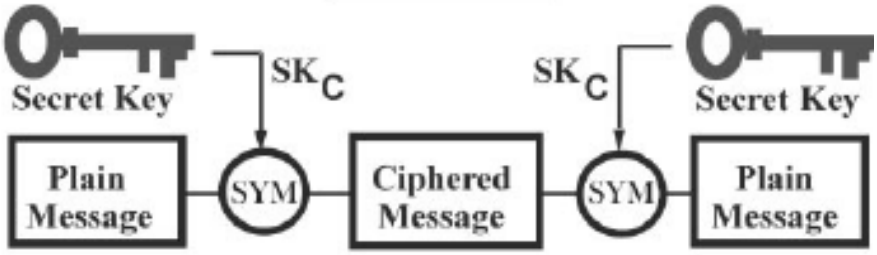
Seiring dengan perkembangan teknologi, algoritma kriptografi pun mulai berubah menuju ke arah algoritma kriptografi yang lebih rumit dan kompleks.

Kriptografi mau tidak mau harus diakui mempunyai peranan yang paling penting dalam peperangan sehingga algoritma kriptografi berkembang cukup pesat pada saat Perang Dunia I dan Perang Dunia II. Menurut catatan sejarah, terdapat beberapa algoritma kriptografi yang pernah digunakan dalam peperangan, diantaranya adalah ADFVGX yang dipakai oleh Jerman pada Perang Dunia I, *Sigaba/M-134* yang digunakan oleh Amerika Serikat pada Perang Dunia II, *Typex* oleh Inggris, dan *Purple* oleh Jepang. Selain itu Jerman juga mempunyai mesin legendaris yang dipakai untuk memecahkan sandi yang dikirim oleh pihak musuh dalam peperangan yaitu, *Enigma*.

Algoritma kriptografi yang baik tidak ditentukan oleh kerumitan dalam mengolah data atau pesan yang akan disampaikan. Yang penting, algoritma tersebut harus memenuhi 4 persyaratan berikut :

- Kerahasiaan. Pesan (*plaintext*) hanya dapat dibaca oleh pihak yang memliki kewenangan.
- Autentikasi. Pengirim pesan harus dapat diidentifikasi dengan pasti, penyusup harus dipastikan tidak bisa berpura-pura menjadi orang lain.
- Integritas. Penerima pesan harus dapat memastikan bahwa pesan yang dia terima tidak dimodifikasi ketika sedang dalam proses transmisi data.
- ***Non-Repudiation***. Pengirim pesan harus tidak bisa menyangkal pesan yang dia kirimkan.

Kriptografi pada dasarnya terdiri dari dua proses, yaitu proses enkripsi dan proses dekripsi. Proses enkripsi adalah proses penyandian pesan terbuka menjadi pesan rahasia (ciphertext). Ciphertext inilah yang nantinya akan dikirimkan melalui saluran komunikasi terbuka. Pada saat ciphertext diterima oleh penerima pesan, maka pesan rahasia tersebut diubah lagi menjadi pesan terbuka melalui proses dekripsi sehingga pesan tadi dapat dibaca kembali oleh penerima pesan. Secara umum, proses enkripsi dan dekripsi dapat digambarkan sebagai berikut :



Gambar 2.6 Gambar Proses Enkripsi dan Dekripsi[4]

Dalam sistem komputer, pesan terbuka (*plaintext*) diberi lambang M, yang merupakan singkatan dari *Message*. *Plaintext* ini dapat berupa tulisan, foto, atau video yang berbentuk data biner. *Plaintext* inilah yang nantinya akan dienkripsi menjadi pesan rahasia atau *ciphertext* yang dilambangkan dengan C (*Ciphertext*). Secara matematis, fungsi enkripsi ini dinotasikan dengan :

$$E(M) = C$$

Sedangkan fungsi dekripsi adalah proses pembalikan dari *ciphertext* menjadi *plaintext* kembali. Secara matematis dinotasikan sebagai berikut :

$$D(C) = M$$

$$D(E(M)) = M$$

2.6.1. Panjang Kunci

Keamanan dari sebuah teknik penyandian tergantung dari dua hal : algoritma penyandian dan panjang kunci (*key*). Algoritma sangat menentukan kekuatan dari sebuah teknik penyandian, tetapi panjang kunci juga tidak kalah penting dalam menentukan kekuatan sebuaBh teknik penyandian.

Sebagai contoh, apabila seorang *kriptanalis* mengetahui algoritma yang dipakai untuk melakukan teknik penyandian terhadap suatu pesan, maka *kriptanalis* tersebut harus mendapatkan kunci yang dipakai terlebih dahulu sebelum dapat melakukan dekripsi terhadap semua *ciphertext* yang dia punya. Satu-satunya cara untuk mendapatkan kunci yang dipakai adalah dengan cara mencoba semua variasi kunci yang ada. Teknik serangan ini sering dikenal dengan nama *brute force.*

Adalah mudah untuk menghitung banyaknya variasi kunci yang ada. Apabila panjang kunci adalah 8 bit, maka ada 2^8 atau 256 kemungkinan kunci yang dapat dicoba. Dari 256 percobaan ini, peluang untuk mendapatkan kunci yang benar adalah 50 persen setelah melalui setengah usaha percobaan. Bila panjang kunci 56 bit, maka ada 2^{56} kemungkinan variasi kunci. Dengan menganggap sebuah superkomputer dapat mencoba satu juta kunci per detik, maka diperkirakan sekitar 2285 tahun untuk menemukan kunci yang benar. Bila menggunakan panjang kunci 64 bit, maka dengan superkomputer yang sama akan membutuhkan 585 ribu tahun. Dengan jangka waktu yang lama ini, maka dapat dipastikan bahwa pesan yang disandikan tersebut tidak mempunyai arti lagi apabila telah berhasil dilakukan dekripsi.

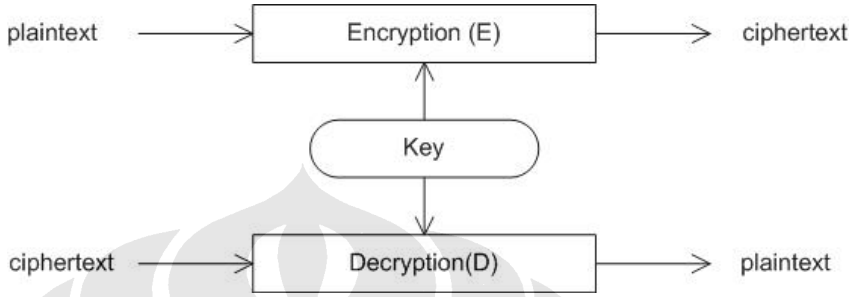
Dengan melihat situasi ini, maka kriptografi yang baik akan memilih untuk menggunakan sepanjang mungkin kunci yang akan digunakan, namun hal ini tidak dapat diterapkan begitu saja. Semakin panjang kunci, semakin lama pula waktu yang digunakan oleh komputer untuk melakukan proses enkripsi. Oleh sebab itu, panjang kunci yang akan digunakan hendaknya memperhatikan 3 hal, yaitu seberapa penting data yang akan dirahasiakan, berapa lama waktu yang dibutuhkan agar data tersebut tetap aman, dan seberapa kuat kemampuan *kriptanalis* dalam memecahkan teknik penyandian kita. Saat ini yang paling banyak dipakai adalah kunci dengan panjang 128 bit karena panjang kunci ini dianggap paling optimal untuk saat ini.

2.6.2. Kriptografi Kunci Rahasia dan Kunci Publik

Pada dasarnya terdapat dua jenis algoritma kriptografi berdasarkan kunci yang digunakan. Yang pertama adalah kriptografi dengan menggunakan secret key dan yang kedua adalah kriptografi yang menggunakan public key. Kriptografi public key menggunakan dua kunci yang berbeda dimana satu kunci digunakan untuk melakukan enkripsi dan kunci yang lain digunakan untuk melakukan dekripsi.

2.6.2.1. Kriptografi Kunci Rahasia

Kriptografi kunci rahasia atau *secret key* adalah kriptografi yang hanya melibatkan satu kunci dalam proses enkripsi dan dekripsi. Proses dekripsi dalam kriptografi *secret key* ini adalah kebalikan dari proses enkripsi.



Gambar 2.7 Kriptografi simetris

Kriptografi *secret key* seringkali disebut sebagai kriptografi konvensional atau kriptografi simetris (*Symmetric Cryptography*) dimana proses dekripsi adalah kebalikan dari proses enkripsi dan menggunakan kunci yang sama.

Kriptografi simetris dapat dibagi menjadi dua, yaitu penyandian blok dan penyandian alir. Penyandian blok bekerja pada suatu data yang terkelompok menjadi blok-blok data atau kelompok data dengan panjang data yang telah ditentukan. Pada penyandian blok, data yang masuk akan dipecah-pecah menjadi blok data yang telah ditentukan ukurannya. Penyandian alir bekerja pada suatu data bit tunggal atau terkadang dalam satu byte. Jadi format data yang mengalami proses enkripsi dan dekripsi adalah berupa aliran bit-bit data.

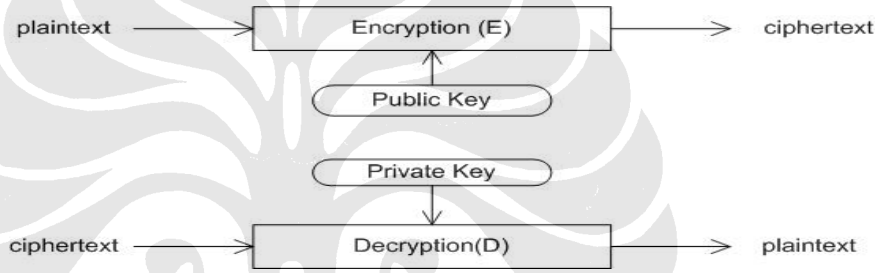
Algoritma yang ada pada saat ini kebanyakan bekerja untuk penyandian blok karena kebanyakan proses pengiriman data pada saat ini menggunakan blok-blok data yang telah ditentukan ukurannya untuk kemudian dikirim melalui saluran komunikasi.

2.6.2.2. Kriptografi Kunci Publik

Kriptografi *public key* sering disebut dengan kriptografi asimetris. Berbeda dengan kriptografi *secret key*, kunci yang digunakan pada proses enkripsi dan

proses dekripsi pada kriptografi public key ini berbeda satu sama lain. Jadi dalam kriptografi *public key*, suatu *key generator* akan menghasilkan dua kunci berbeda dimana satu kunci digunakan untuk melakukan proses enkripsi dan kunci yang lain digunakan untuk melakukan proses dekripsi.

Kunci yang digunakan untuk melakukan enkripsi akan dipublikasikan kepada umum untuk dipergunakan secara bebas. Oleh sebab itu, kunci yang digunakan untuk melakukan enkripsi disebut juga sebagai *public key*. Sedangkan kunci yang digunakan untuk melakukan dekripsi akan disimpan oleh pembuat kunci dan tidak akan dipublikasikan kepada umum. Kunci untuk melakukan dekripsi ini disebut *private key*.



Gambar 2.8 Kriptografi Asimetris

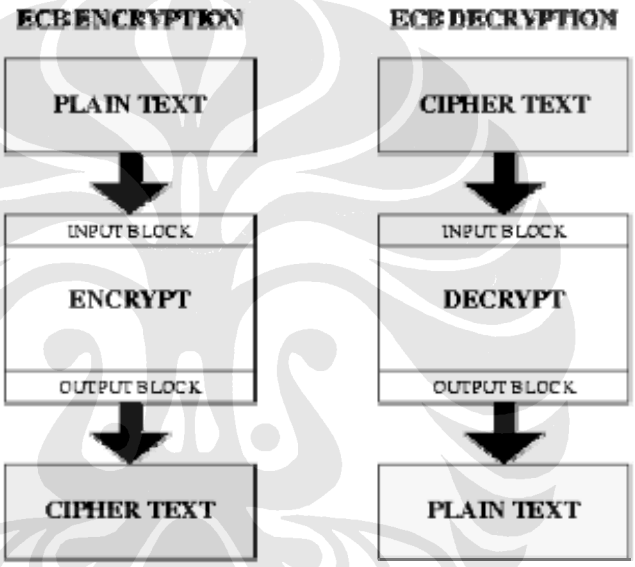
Dengan cara demikian, semua orang yang akan mengirimkan pesan kepada pembuat kunci dapat melakukan proses enkripsi terhadap pesan tersebut, sedangkan proses dekripsi hanya dapat dilakukan oleh pembuat atau pemilik kunci dekripsi. Dalam kenyataannya, kriptografi asimetris ini dipakai dalam ssh, suatu layanan untuk mengakses suatu server.

2.6.3. Penyandian Blok

Penyandian blok pada dasarnya adalah proses penyandian terhadap blok data yang jumlahnya sudah ditentukan. Untuk sistem penyandian blok terdapat empat jenis mode operasi, yaitu *Electronic Code Book (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFB), Output Feedback (OFB).*

2.6.3.1. Electronic Code Book (ECB)

Mode ECB adalah mode yang paling umum dan paling mudah untuk diimplementasikan. Cara yang digunakan adalah dengan membagi data ke dalam blok-blok data terlebih dahulu yang besarnya sudah ditentukan. Blok-blok data inilah yang disebut *plaintext* karena blok data ini belum disandikan. Proses enkripsi akan langsung mengolah *plaintext* menjadi *ciphertext* tanpa melakukan operasi tambahan. Suatu blok *plaintext* yang dienkripsi dengan menggunakan kunci yang sama akan menghasilkan *ciphertext* yang sama.



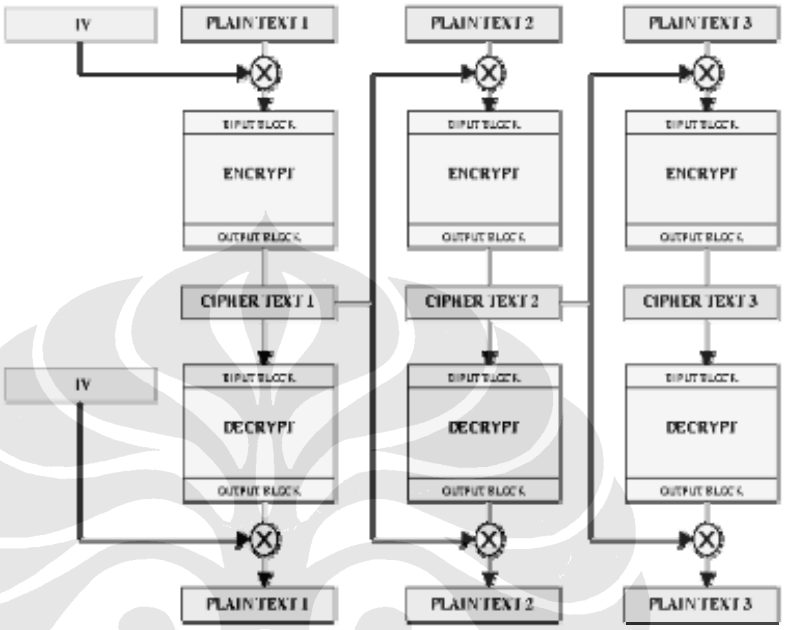
Gambar 2.9 Mode Operasi ECB

Keuntungan dari mode OBC ini adalah kemudahan dalam implementasi dan pengurangan resiko salahnya semua *plaintext* akibat kesalahan pada satu *plaintext*. Namun mode ini memiliki kelemahan pada aspek keamanannya. Dengan mengetahui pasangan *plaintext* dan *ciphertext*, seorang *kriptanalis* dapat menyusun suatu *code book* tanpa perlu mengetahui kuncinya.

2.6.3.2. Cipher Block Chaining (CBC)

Pada CBC digunakan operasi umpan balik atau dikenal dengan operasi berantai (*chaining*). Pada CBC, hasil enkripsi dari blok sebelumnya adalah

feedback untuk enkripsi dan dekripsi pada blok berikutnya. Dengan kata lain, setiap blok *ciphertext* dipakai untuk memodifikasi proses enkripsi dan dekripsi pada blok berikutnya.

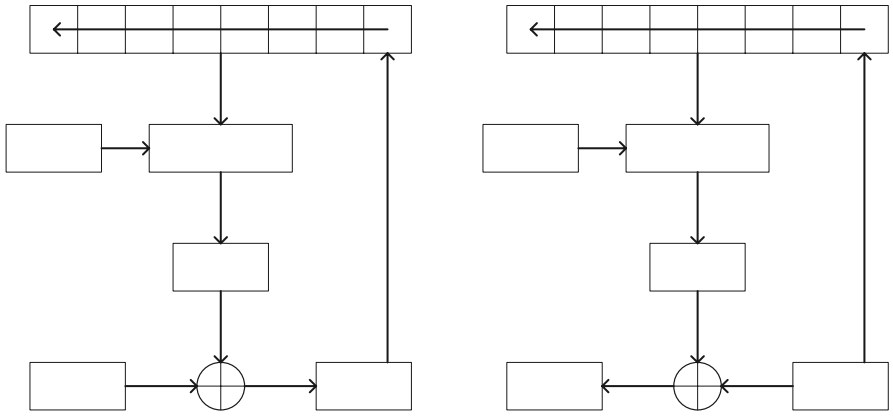


Gambar 2.10 Mode Operasi CBC

Pada CBC diperlukan data acak sebagai blok pertama. Blok data acak ini sering disebut *initialization vector* atau IV. IV digunakan hanya untuk membuat suatu pesan menjadi unik dan IV tidak mempunyai arti yang penting sehingga IV tidak perlu dirahasiakan.

2.6.3.3. Cipher Feedback (CFB)

Pada mode CBC, proses enkripsi atau dekripsi tidak dapat dilakukan sebelum blok data yang diterima lengkap terlebih dahulu. Masalah ini diatasi pada mode *Cipher Feedback* (CFB). Pada mode CFB, data dapat dienkripsi pada unit-unit yang lebih kecil atau sama dengan ukuran satu blok. Misalkan pada CFB 8 bit, maka data akan diproses tiap 8 bit.



Gambar 2.11 Mode Operasi CFB

Pada permulaan proses enkripsi, IV akan dimasukkan dalam suatu *register* geser. IV ini akan dienkripsi dengan menggunakan kunci yang sudah ada. Dari hasil enkripsi tersebut, akan diambil 8 bit paling kiri atau *Most Significant Bit* untuk di-XOR dengan 8 bit dari *plaintext*. Hasil operasi XOR inilah yang akan menjadi *ciphertext* dimana *ciphertext* ini tidak hanya dikirim untuk ditransmisikan tetapi juga dikirim sebagai *feedback* ke dalam *register* geser untuk dilakukan proses enkripsi untuk 8 bit berikutnya.

Kunci

2.6.3.4. Output Feedback (OFB)

Sama pada mode CFB, mode OFB juga memerlukan sebuah *register* geser dalam pengoperasiannya. Pertama kali, IV akan masuk ke dalam *register* geser dan dilakukan enkripsi terhadap IV tersebut. Dari hasil proses enkripsi tersebut akan diambil 8 bit paling kiri untuk dilakukan XOR dengan *plaintext* yang nantinya akan menghasilkan *ciphertext*. *Ciphertext* tidak akan diumpan balik ke dalam *register* geser, tetapi yang akan diumpan balik adalah hasil dari enkripsi IV.

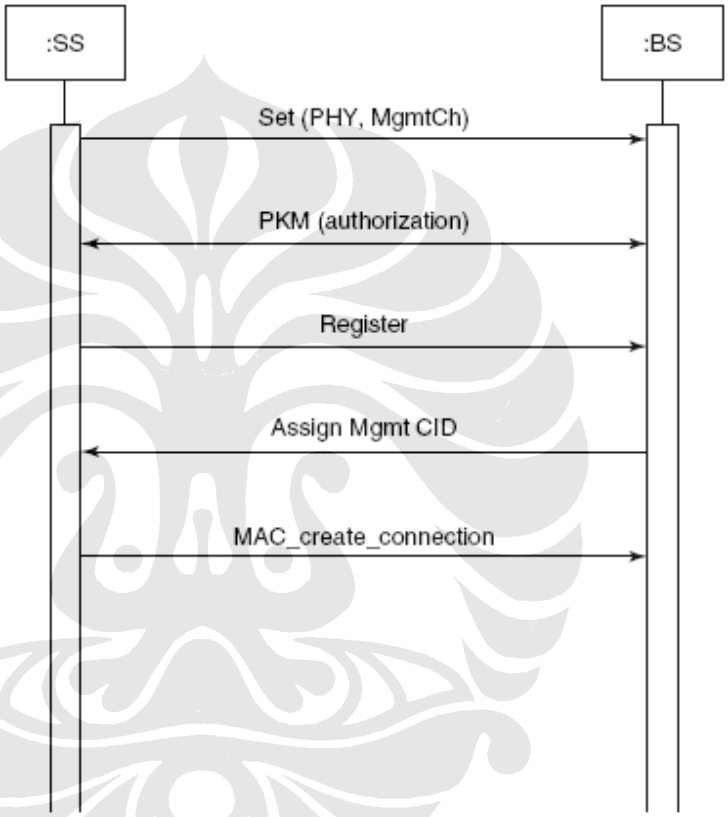
2.7. Proses Pengamanan

2.7.1. Memulai Koneksi

P(n)

Unuk memulai suatu koneksi antara SS dngan BS, dilakukan melalui beberapa tahapan. Tahapan paling awal yang harus dilakukan adalah pada layer

PHY. SS melakukan pemindaian signal untuk mengetahui keberadaan BS yang terdekat. Kemudian megobservasi trafik dari BS tersebut untuk menentukan parameter-parameter seperti timing, modulasi, error correction dan power. Kemudian yang terakhir mengidentifikasi timeslot yang akan digunakan untuk melakukan initial request. Dengan begitu SS dapat membentuk channel yang akan digunakan untuk meminta autorisasi dari BS.



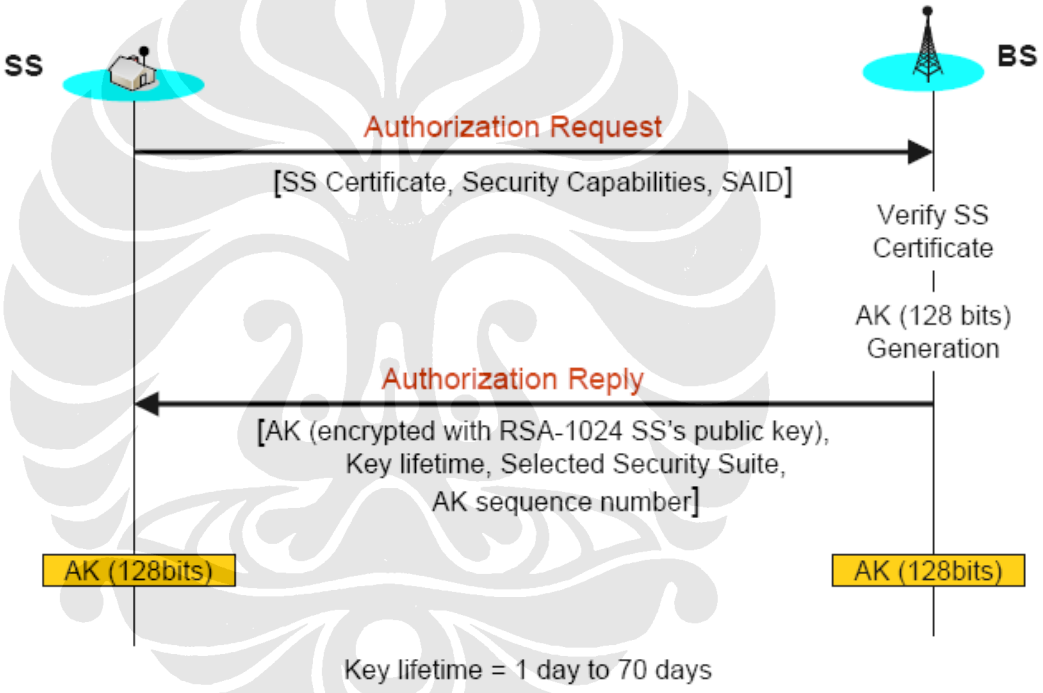
Gambar 2.12 Proses koneksi[2]

Bila telah mendapatkan alokasi channel, SS akan meminta autorsasi pada BS. SS mengirimkan permintaan tersebut dengan menyeertakan MAC address dan sertifikasi dari pabrikan pembuatnya.

Bila MAC address dan sertifikasi yang diberikan SS sesuai dan berhak mendapatkan akses, maka BS akan memberikan autorisasi. Sehingga kemudian SS dapat melakukan registrasi untuk mendapatkan management CID yang akan digunakan untuk membentuk koneksi antara SS dan BS sesuai dengan haknya.

2.7.2. Autentikasi

Untuk menentukan SS berhak membentuk koneksi atau tidak, dilakukan melalui proses auentikasi. SS memulai proses autorisasi dengan mengirimkan informasi autentikasi kepada BS yang dituju. Informasi tersebut berisi sertifikasi X.509 yang dikeluarkan oleh pabrikan yang bersangkutan atau badan otoritas lainnya. Setela mengirim informasi autentikasi tersebut, SS juga segera mengirimkan authirization request untuk meminta key autentikasi. Detail proses autentikasi tersebut digambarkan melalui Gambar 2.13 berikut :



Gambar 2.13 Proses Autentikasi[3]

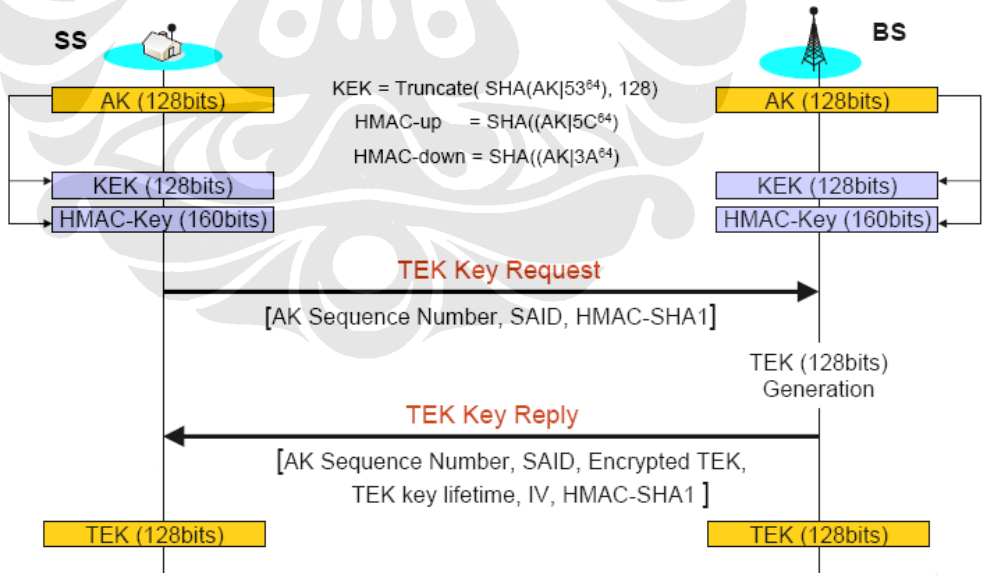
Seperti yang terilustrasi dalam Gambar 2.13, SS menginformasikan pada BS SAID-nya. BS melakukann validasi terhadap identitas SS yang melakukan permintaan. Setelah melakukan verifikasi terhadap identitas SS tersebut, BS mengaktifkan sebuah authentication key (AK) untuk SS, mengenkripsinya dengan kunci publik, kemudian mengirimkannya kembali kepada SS melalui authorization reply.

Data inforasi yang dikirimkan antara SS dan BS dalam proses autentikasi ini antaralain adalah:

- Sertifikasi X.509 yang dieluarkan oleh vendor (sebagai identitas SS yang melakukan request).
- Deskripsi dari algoritma kriptografi yang didukung oleh SS (disebut sebagai security association [SA]).
- Basic CID dari SS, yang sama dengan primary SAID.

2.7.3. Pertukaran Kunci Data

Base station (BS) bertanggung jawab dalam pemeliharaan informasi untuk semua *securityassociation (SA),* sedangkan *subscriber station* (*SS)* bertanggung jawab untuk mendukung otorisasi dengan *base station (BS)* dan memelihara otorisasi kunci yang aktif. Setelah *subscriber station*(*SS)* menyelesaikan negoisasi, maka akan mengubah otorisasi dengan *base station (BS)*. Awalnya*base station (BS)* menerima pesan dari *subscriber station* (*SS)* untuk mengaktifkan otoritas yang baru, kemudian *base station (BS)* mengirimkan jawaban atas pertanyaan *subscriber station* (*SS).*



Gambar 2.14 Proses Pertukaran Key[3]

Authorization key (AK) akan aktif sampai waktu yang ditentukan berakhir sesuai dengan bataswaktu *authorization key (AK).* Apabila *subscriber station* (*SS)* mengalami kegagalan dalam melaksanakan otorisasi sebelum waktu *authorization*

key (AK) berakhir, maka *base station (BS)* tidak dapat mengaktifkan *authorization key (AK)* untuk *subscriber station* (*SS),* dan *subscriber station* (*SS)* tidak diberi otorisasi. *Base station (BS)* akan menghapus semua *traffic encryption* (*TEK)* dengan otorisasi dari *subscriber station* (*SS). Base station (BS)* selalu menyiapkan *authorization key (AK)* ke *subscriber station* (*SS)* atas suatu permintaan. *Base station (BS)* akan mendukung dua aktivitas *authorization key (AK)* untuk setiap *client subscriber station* (*SS). Authorization key (AK)* mempunyai batas *lifetime* dan secara periodik akan di-*refresh*.

2.7.4. Privasi Data

Untuk menjaga privasi data pada sistem mobile WiMax ini digunakan metode enkripsi Adavnce Encryption Standard (AES). Proses enkripsi pada algoritma AES terdiri dari 4 jenis transformasi *bytes,* yaitu :

1. *SubBytes*, transformasi byte dimana setiap elemen pada state akan dipetakan dengan menggunakan sebuah tabel substitusi (*S-Box*).
2. *ShiftRows,* Transformasi Shiftrows pada dasarnya adalah proses pergeseran bit dimana bit paling kiri akan dipindahkan menjadi bit paling kanan (rotasi bit).
3. *Mixcolumns,* Mixolumns mengoperasikan setiap elemen yang berada dalam satu kolom pada *state*. Elemen pada kolom dikalikan dengan suatu polinomial tetap $a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$.
4. A*ddRoundKey,* Pada proses AddRoundKey, sebuah *round key* ditambahkan pada *state* dengan operasi bitwise XOR. Setiap *round key* terdiri dari *Nb word* dimana tiap *word* tersebut akan dijumlahkan dengan *word* atau kolom yang bersesuaian dari *state.*

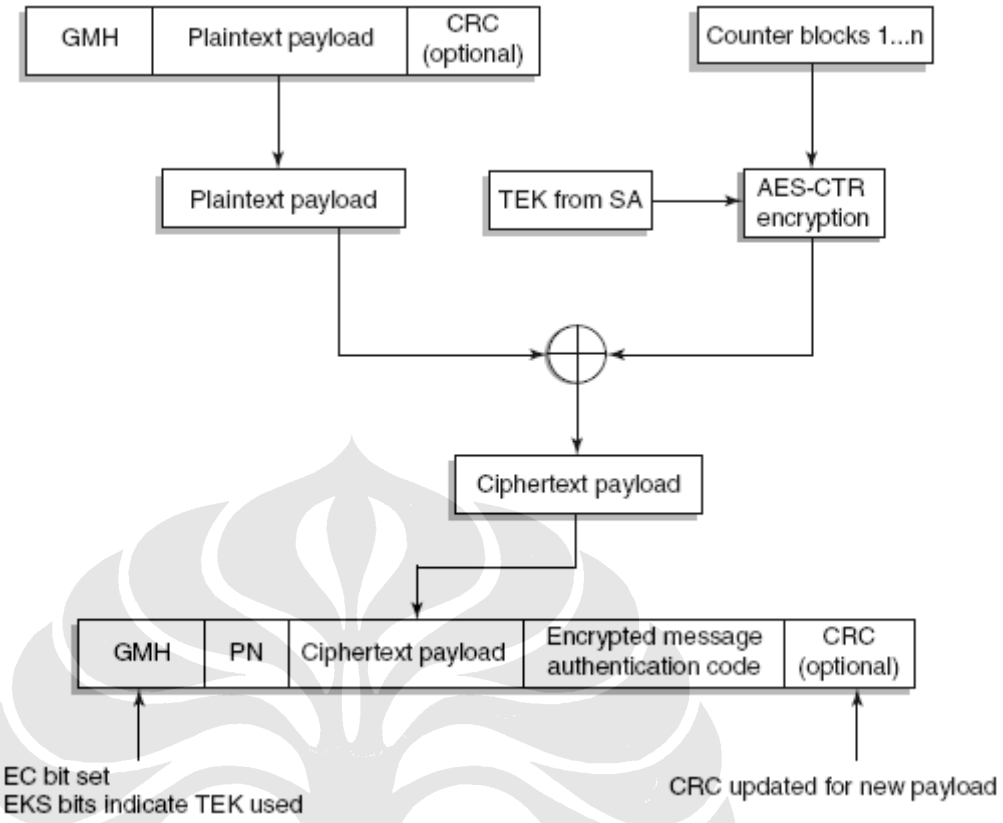
Pada awal proses enkripsi, *input* yang telah dikopikan ke dalam *state* akan mengalami transformasi byte *AddRoundKey*. Setelah itu, *state* akan mengalami transformasi *SubBytes*, *ShiftRows*, *MixColumns*, dan *AddRoundKey* secara berulang-ulang sebanyak *Nr*. Proses ini dalam algoritma AES disebut sebagai *round function*. *Round* yang terakhir agak berbeda dengan *round-round* sebelumnya dimana pada *round* terakhir, *state* tidak mengalami transformasi *MixColumns.*

Transformasi *cipher* dapat dibalikkan dan diimplementasikan dalam arah yang berlawanan untuk menghasilkan *inverse cipher* yang mudah dipahami untuk algoritma AES. Transformasi byte yang digunakan pada invers *cipher* adalah:

1. *InvShiftRows,* yaitu transformasi byte yang berkebalikan dengan transformasi *ShiftRows*. Pada transformasi *InvShiftRows,* dilakukan pergeseran bit ke kanan sedangkan pada *ShiftRows* dilakukan pergeseran bit ke kiri.
2. *InvSubBytes,* yaitu merupakan transformasi *bytes* yang berkebalikan dengan transformasi SubBytes. Pada *InvSubBytes*, tiap elemen pada *state* dipetakan dengan menggunakan tabel *inverse S-Box*
3. Pada *InvMixColumns*, kolom-kolom pada tiap *state* (*word*) akan dipandang sebagai polinom atas GF(2^8) dan mengalikan modulo $x^4 + 1$ dengan polinom tetap $a^{-1}(x)$ yang diperoleh dari :

   $$a^{-1}(x) = \{0B\}x^3 + \{0D\}x^2 + \{09\}x + \{0E\}.$$

4. *AddRoundKey* Transformasi *Inverse AddRoundKey* tidak mempunyai perbedaan dengan transformasi *AddRoundKey* karena pada transformasi ini hanya dilakukan operasi penambahan sederhana dengan menggunakan operasi bitwise XOR.[5]



Gambar 2.15 Diagram Blok Pengenkripsian Payload