**Modul 12**

1. **Tujuan**

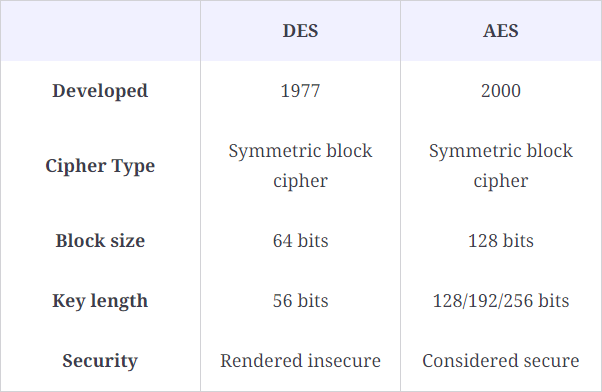
1. Mampu menjelaskan definisi dan konsep Advanced Encryption Standard (AES)

2. Mampu memahami konsep dasar Advanced Encryption Standard (AES)

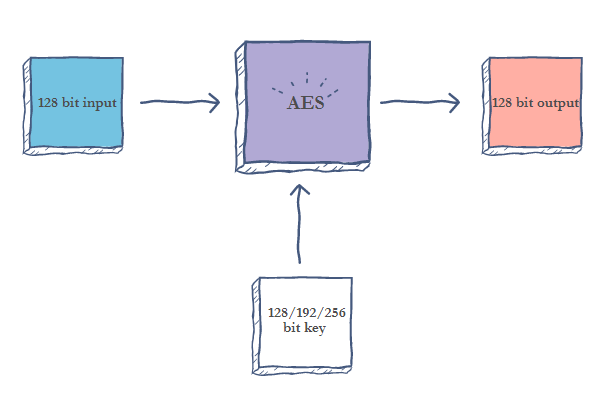
3. Mampu mengimplementasikan algoritma AES dalam mengamankan data pada sistem

1. **Dasar teori**
2. **Advanced Encryption Standard**

Pada 1990-an, DES dianggap tidak aman karena ukuran kunci 56-bitnya yang relatif kecil. Jadi, pada tahun 1997, Advanced Encryption Standard (AES) diusulkan sebagai tanggapan atas permintaan publik untuk proposal oleh National Institute of Standards and Technology (NIST). Tabel di bawah ini menunjukkan bagaimana AES dibandingkan dengan pendahulunya.



Algoritma AES (juga dikenal sebagai algoritma Rijndael) adalah algoritma cipher blok simetris yang mengambil teks biasa dalam blok 128 bit dan mengubahnya menjadi ciphertext menggunakan kunci 128, 192, dan 256 bit. Karena algoritma AES dianggap aman, ia berada dalam standar dunia.



1. **Fungsionalitas AES**

Fungsionalitas algoritma AES mungkin tampak rumit tetapi dalam arti sebenarnya, sangat mudah untuk dipahami. Langkah pertama adalah mengetahui bahwa AES memiliki 3 Block cipher, yaitu:

* AES-128

Seperti yang telah kita ketahui, proses enkripsi dan dekripsi dilakukan dengan menggunakan sebuah kunci. Sandi blok ini menggunakan kunci yang panjangnya 128 bit untuk enkripsi dan dekripsi pesan dan merupakan yang paling tidak aman dari tiga blok yang tersedia.

Meskipun AES-128 belum pernah diretas, ketahanannya terhadap serangan brute force mengkhawatirkan. Terlepas dari masalah keamanan yang terlihat di AES-128, ini sangat cepat dan dilengkapi dengan banyak efisiensi dalam enkripsi data.

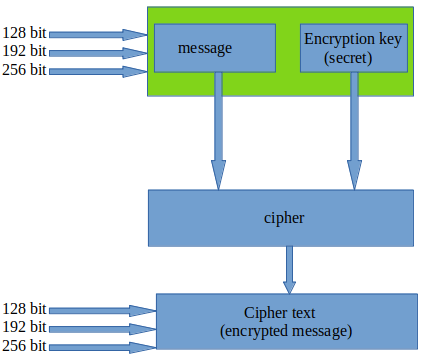
* AES-192

AES-192 menggunakan kunci yang panjangnya 192 bit untuk enkripsi dan dekripsi pesan. Ini lebih tahan terhadap serangan brute force dibandingkan dengan AES-128 karena memiliki kunci yang lebih panjang, sehingga lebih aman. Meskipun demikian, AES-192 tidak umum digunakan, orang cenderung condong ke AES-256.

* AES-256

AES-256 menggunakan kunci yang panjangnya 256 bit untuk enkripsi dan dekripsi pesan. Blok ini lebih aman jika dibandingkan dengan AES-128 dan AES-192 karena panjang kunci enkripsinya.

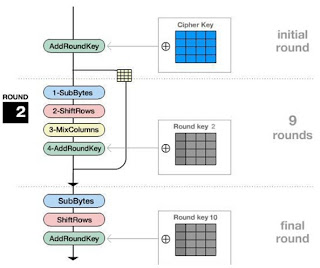
Di bawah ini adalah aliran representasi diagram sederhana tentang bagaimana AES mengenkripsi data:



Proses enkripsi memiliki 10 putaran untuk kunci panjang 128-bit, 12 putaran untuk kunci panjang 192-bit, dan 14 putaran untuk kunci panjang 256-bit.

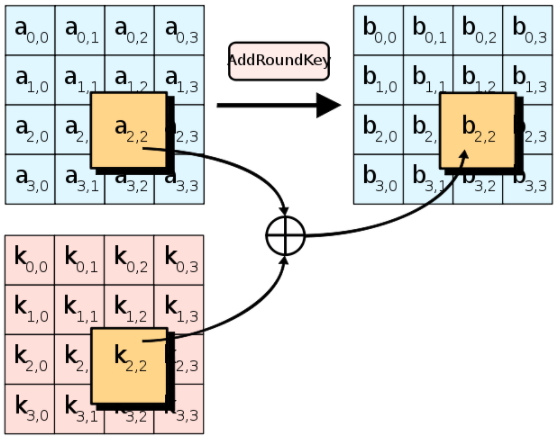
1. **Algoritma Enkripsi**

* **Alur Enkripsi Teks Menggunakan AES**



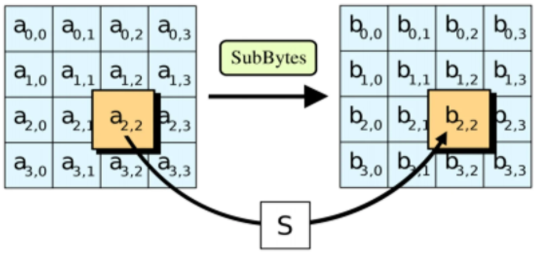
* **Step 1 Generation of round keys**

Ini terjadi melalui proses yang disebut ekspansi kunci di mana kunci rahasia asli digunakan untuk memperoleh kunci bulat dengan menggunakan algoritma jadwal kunci Rijndael. Pada langkah pertama, byte teks blok diganti berdasarkan aturan yang ditentukan oleh kotak S yang telah ditentukan sebelumnya (kependekan dari kotak substitusi).

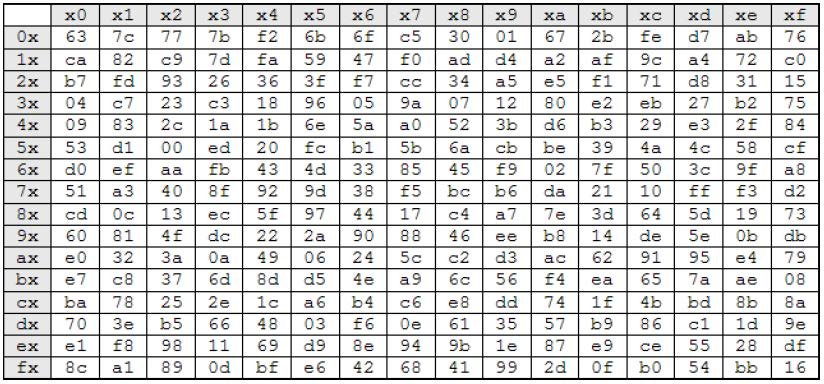


* **Step 2 Substitution of the bytes**

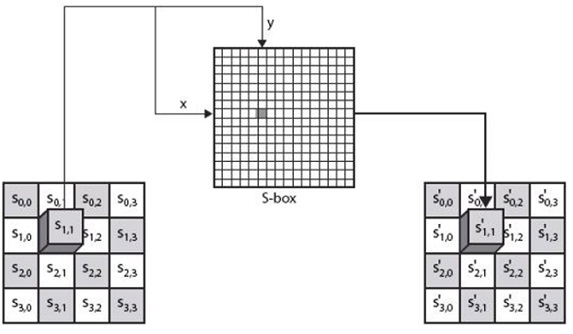
Pada langkah pertama, byte teks blok diganti berdasarkan aturan yang ditentukan oleh kotak S yang telah ditentukan sebelumnya (kependekan dari kotak substitusi).



Prinsip dari Sub Bytes adalah menukar isi matriks/tabel yang ada dengan matriks/tabel lain yang disebut dengan Rijndael S-Box. Di bawah ini adalah contoh Rijndael S-Box dan Sub-Bytes.



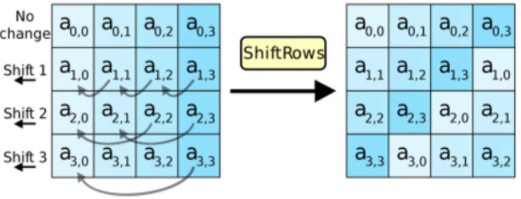
Rijndael Box



Contoh Sub-Bytes

* **Step 3 Shifting the Rows**

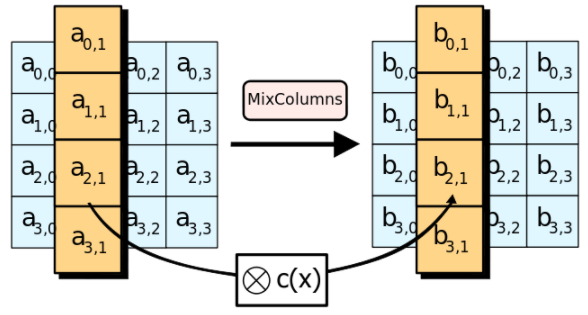
Selanjutnya adalah langkah permutasi. Pada langkah ini, semua baris kecuali yang pertama digeser satu, seperti yang ditunjukkan di bawah ini :



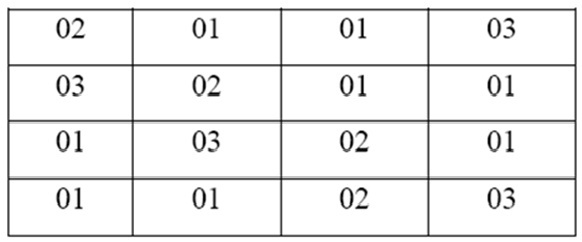
Shift Rows seperti namanya adalah sebuah proses yang melakukan shift atau pergeseran pada setiap elemen blok/tabel yang dilakukan per barisnya. Yaitu baris pertama tidak dilakukan pergeseran, baris kedua dilakukan pergeseran 1 byte, baris ketiga dilakukan pergeseran 2 byte, dan baris keempat dilakukan pergeseran 3 byte. Pergeseran tersebut terlihat dalam sebuah blok adalah sebuah pergeseran tiap elemen ke kiri tergantung berapa byte tergesernya, tiap pergeseran 1 byte berarti bergeser ke kiri sebanyak satu kali.

* **Step 4 Mix Column**

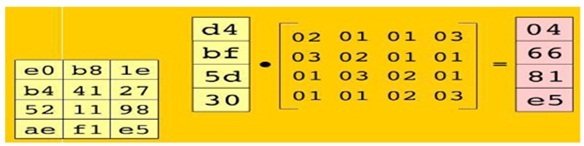
Pada langkah ketiga, Hill cipher digunakan untuk mengacak pesan lebih banyak dengan mencampur kolom blok.



Yang terjadi saat Mix Column adalah mengalikan tiap elemen dari blok chiper dengan matriks yang ditunjukkan oleh Mix Column. Tabel sudah ditentukan dan siap pakai. Pengalian dilakukan seperti perkalian matriks biasa yaitu menggunakan dot product lalu perkalian keduanya dimasukkan ke dalam sebuah blok chiper baru. Gambar ilustrasi di bawah akan menjelaskan mengenai bagaimana perkalian ini seharusnya dilakukan. Dengan begitu seluruh rangkaian proses yang terjadi pada AES telah dijelaskan dan selanjutnya adalah menerangkan mengenai penggunaan tiap-tiap proses tersebut.



Tabel Mix Column



Contoh Perhitungan Matrix

* **Step 5 Repeat Step 1 to 5**

Langkah selanjutnya pada algoritma AES adalah melakukan iterasi sesuai dengan kunci yang digunakan.

1. **Keamanan**

Kesimpulan yang didapat adalah:

* AES terbukti kebal menghadapi serangan konvensional (linear dan diferensial attack) yang menggunakan statistik untuk memecahkan sandi;
* Kesederhanaan AES memberikan keuntungan berupa kepercayaan bahwa AES tidak ditanami trapdoor;
* Namun, kesederhanaan struktur AES juga membuka kesempatan untuk mendapatkan persamaan aljabar AES yang selanjutnya akan diteliti apakah persamaan tersebut dapat dipecahkan;
* Bila persamaan AES dapat dipecahkan dengan sedikit pasangan plaintext/ciphertext, maka riwayat AES akan berakhir;
* AES didesain dengan sangat hati-hati dan baik sehingga setiap komponennya memiliki tugas yang jelas;
* AES memiliki sifat cipher yang diharapkan yaitu : tahan menghadapi analisis sandi yang diketahui, fleksibel digunakan dalam berbagai perangkat keras dan lunak, baik digunakan untuk fungsi hash karena tidak memiliki weak (semi weak) key, cocok untuk perangkat yang membutuhkan key agility yang cepat, dan cocok untuk stream cipher.

**D. Contoh Program Python**



Import algoritma AES dari library Crypto (pycryptodome). Import fungsi random untuk generate nilai yang akan digunakan sebagai kunci.



Fungsi get\_random\_bytes berguna untuk generate nilai random sebesar 16 bit kemudian disimpan di variabel key. Variabel data\_to\_encrypt menyimpan inputan dari keyboard.



Konversi data teks ke dalam bytes

**Enkripsi**



Buat objek cipher baru isinya algoritma AES. Selanjutnya panggil fungsi encrypt() dengan parameter data teks yang telah kita inputkan.



Simpan data pada variabel ciphered\_data

**Dekripsi**



Buat objek cipher baru, kemudian panggil fungsi decrypt() dengan parameter ciphertext hasil enkripsi.



Konversi plaintext hasil dekripsi dari bentuk bit ke string (teks)



Cetak hasil enkripsi dan dekripsi

**E. TUGAS**

1. Membuat program enkripsi dan dekripsi menggunakan algoritma DES.

2. Yang dikumpukan :

Kode program dan Screenshot ­­­­

3. Pengumpulan Tugas Praktikum

* **Upload file word** ke SPADA
* **Untuk kelas TI E** paling lambat tanggal 22 November 2021 Jam 23.59

**Untuk kelas TI D** paling lambat tanggal 23 November 2021 Jam 23.59

* Format penamaan file SKD\_namakelas\_nim\_nama