Judul : Penerapan Algoritma RC6 pada *Json Web Token*(JWT) dalam implementasi *Application Programming Interface* (API)

Rangkuman :

*Application Programming Interface* atau kerap disebut API adalah penghubung antar aplikasi yang berbeda baik menggunakan perangkat yang sama maupun perangkat yang berbeda. Tujuan dari API adalah untuk saling berbagi data antar aplikasi, API juga sering diterapkan dalam sistem berbasis *Client Server*, dimana *Client* meminta data kepada *Server* kemudian *Server* merespon balik dengan mengirimkan data berupa *Array* JSON yang sesuai dengan permintaan dari *Client*.

Pada proses pengiriman data melalui API dapat dilakukan oleh setiap orang melalui *Request* API ke sebuah *server. Server* diharuskan dapat mengindentifikasi apakah permintaan tersebut dari pihak yang telah diberi ijin untuk mendapatkan data yang diinginkan,, Sehingga salah satu permasalahan sistem API menggunakan JSON berada pada pengamanan data yang dikirimkan untuk mengindentifikasi pihak yang melakukan permintaan (*request*). Permasalahan ini dapat diselesaikan dengan berbagai cara salah satu nya menggunakan *auntentikasi* JWT (*Json Web Token*). JWT terdiri dari dua proses utama yaitu proses permintaan Token *autentikasi* dan proses permintaan data.

Pada proses permintaan token berawal dari sang *Clien*t memiliki kunci yang telah ditanamkan sebelumnya pada aplikasi atau akun yang terikat, kemudian *Client* meminta kepada token kepada *server* membawa kunci tersebut, *server* akan melakukan proses *autentikasi* dengan menyamakan kunci yang berada di *server* dan kunci yang dibawa oleh *Client* apabila kunci tersebut sama maka *server* akan men-generate sebuah *plaintext* yang akan disimpan dan dienkripsi menggunakan algoritma RC6. Hasil enkripsi tersebut merupakan token yang akan dikembalikan ke Client sebagai pengenal dalam proses permintaan Data.

Setelah mendapatkan token dari *serve*r, *Client* dapat meminta data yang diinginkan membawa token, *server* yang mendapat permintaan kemudian melakukan proses identifikasi untuk melihat apabila *Client* yang meminta mempunyai hak atau tidak. Token yang dibawa oleh *Client* kemudian didekripsi menjadi Plaintext untuk dicocokkan dengan token plaintext yang disimpan di server apabila terdapat kesamaan server merespon balik dengan mengirim data yang diminta, sebaliknya apabila tidak terdapat kesamaan Server tidak mengirim data melainkan array JSON kosong.

Token yang digenerate sebelumnya akan direset oleh server tiap 10/20 Menit sekali.. proses permintaan Token pun dilakukan kembali

Gambaran Alur Pengiriman Data

Apps mobile

mempunyai Key/Kunci yang tertanam dalam Apps

Apabila Apps mobile terbuka

Apps men-request token ke

Server dengan mengirim key

Tidak mengirim data

Apapun melainkan Array

Kosong

Server tidak mengirim Token

Mobile menerima data

Yang dikirim server

Mengembalikan request

berupa data berbentuk

Array JSON

Mobile apps menerima token

Dan menyimpannya

Mengembalikan request dengan mengirim Token / Ciphertext

Server menyimpan Plainteks yang nantinya bakal digunakan pada Autentikasi permintaan Data

Yes

Server menggenerate Plainteks yang nantinya di enkripsi menggunakan Algoritma RC6

No

Server menerima kunci kemudian mencocokkan

Dengan kunci server

No

Hasil dekripsi kemudian dicocokkan dengan Plainteks

Yang sebelumnya disimpan

Server menerima token dan mulai mendekripsikan menggunakan algoritma yang sama digunakan pada enkripsi yaitu RC6

Mobile menrequest data yang diinginkan kepada server membawa token