IMPLEMENTASI ONE TIME PASSWORD (OTP) MOBILE TOKEN DENGAN MENGGUNAKAN METODE ALGORITMA MD5 DAN SHA

Hendrik Lase[1)], **Mufti**[2)]
[1]Program studi, Fakultas Teknologi Informasi, Universitas Budi Luhur
[1,2]Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260
E-mail : lasehendrik1@gmail.com[1)], muftyhayat@gmail.com[2)]

Abstrak

Adanya beberapa kelemahan pada password biasa, terutama mudahnya dilakukan ancaman replay attack (pengulangan) dan masquerade (penyamaran) oleh para hackers, menjadi kendala bagi pemilik jaringan komputer atau admin untuk mendapatkan keamanan yang optimal dalam melakukan proses autentikasi. Setiap pengguna yang bertindak sebagai admin perlu terlebih dahulu diverifikasi dengan mengirimkan OTP, yang disebut dengan OTP mobile token berbasis android dengan menggunakan metode algoritma MD5 dan SHA.

Kata kunci: One Time Password, Message Digest 5, Secure Hash Algorithm,android, web

1. PENDAHULUAN

1.1. Latar Belakang

Sistem informasi memang menguntungkan dan dapat meningkatkan kinerja dari semua komponen organisasi atau perusahaan. Namun, keamanan sistem informasi yang berbasis web sangat rawan untuk disadap oleh pihak yang tidak bertanggung jawab. Perkembangan sistem informasi berbasis web dalam bidang pekerjaan saat ini pun sudah semakin pesat, dimana seluruh data dapat di akses dengan mudah oleh siapa saja. Sistem otentikasi OTP adalah salah satu cara untuk mengatasi serangan peretas pada sebuah *server*.

1.2. Rumusan Masalah

Bagaimana cara untuk meningkatkan keamanan dari proses *login* agar pada saat *username* dan *password* di sadap oleh orang yang tidak bertanggung jawab *username* masih tetap aman.

1.3. Tujuan Penulisan

Tujuan dari penelitian adalah merancang dan membangun sistem *login* aplikasi web berbasis *android*, dengan menggunakan metode *MessageDigest* 5 (MD5) dan *SecureHashAlgorithm* (SHA).

1.4. Batasan Masalah

Agar tercapai sasaran terhadap penyusunan penelitian yang dilakukan maka diadakan pembatasan terhadap masalah. Penelitian akan memanfaatkan *Smartphone Android* untuk menggantikan token dalam menjalankan aplikasi *OneTimePassword*.

a. Sistem operasi *smartphone* yang digunakan minimal adalah *Android* Versi 4.2 *JellyBean*.
b. Algoritma yang digunakan adalah *MessageDigest* 5 dan *SecureHashAlgorithm*.
c. Penelitian ini hanya berfokus pada *OneTimePassword*.
d. Penelitian ini akan memanfaatkan *smartphoneandroid* sebagai pengganti token.

1.5. Metode Penelitian

a. Metode Kepustakaan

Mencari, memahami, serta mempelajari dengan seksama dari buku, artikel, serta sumber - sumber pustaka lainnya seperti jurnal, forum diskusi, pendapat ahli, *textbook*, dan sebagainya yang berkaitan dengan *OneTimePassword*, MD5, dan SHA.

b. Metode Wawancara

Disini lah dilakukan wawancara mengenai metode aplikasi One Time Password. Hal ini dilakukan untuk mendapatkan informasi tentang pengembangan fitur fitur dari aplikasi One Time Password yang nantinya akan di-implementasiakan.

c. Metode *Prototyping*

Prototyping adalah salah satu teknik analisis data dalam pembuatan perangkat lunak dan *Prototyping* merupakan pengembangan yang cepat dan pengujian terhadap model kerja dari aplikasi baru melalui proses interaksi dan berulang – ulang.

1) Menentukan Kebutuhan
Bertemu dengan *user* dan kemudian secara melakukan konsep sistem yang akan di kembangkan.
2) Menentukan Prototype
Setelah garis bersar sudah disepakati, barulah melakukan pembuatan perangkat lunak dengan menentukan rancangan aplikasi sementara yang difokuskan pada garis besar konsep yang sudah disepakati.
3) Evaluasi Prototype
User melakukan evaluasi dari *prototype* aplikasi, apakah sudah sesuai dengan keinginan pengguna.
4) Mengkodekan Aplikasi
Dalam tahap ini, *prototyping* yang sudah disepakati diterjemahkan kedalam bahasa *pemograman* yang sesuai.
5) Menguji Aplikasi
Setelah aplikasi yang sudah di buat jadi, aplikasi akan ditesting terlebih dahulu untuk mengetahui ada atau tidaknya *bugs* pada fungsi utama aplikasi tersebut.

6) Evaluasi Aplikasi
Pengguna melakukan evaluasi terhadap aplikasi yang dikembangkan, apakah aplikasi sudah sesuai dengan apa yang di harapkan. Jika sudah maka langkah ketujuh akan dilakukan.
7) Menggunakan Prototype
Jika aplikasi yang dikembangkan sesuai dengan keinginan user, kemudian aplikasi tersebut dapat langsung diimplementasikan.

2. LANDASAN TEORI

2.1. Keamanan Komputer

Keamanan komputer merupakan satu cabang teknologi yang di kenal dengan nama keamanan informasi yang di terapkan pada komputer. Pengertian tentang keamanan komputer ini beragam – ragam. Menurut Garfinkel dan Spafford, komputer dikatakan aman jika bisa diandalkan dan perangkat lunaknya bisa berkerja sesuai dengan yang diharapkan, dimana keamanan komputer memiliki 5 tujuan anatara lain :

a. Avalability
b. Integrity
c. Control
d. Audit
e. Confidentiality

Tujuan dari keamanan komputer itu sendiri untuk berusaha melindungi data dan informasi dari orang yang tidak berada dalam ruang lingkupnya.

2.2. Serangan Keamanan Komputer

Di kutip dari (Santosa 2013) cara seseorang mendapatkan pesan dalam saluran komunikasi, penyerangan dapat dikategorikan menjadi[9]:

a. *Sniffing*
Secara harifiah berarti ingin mengetahui, tentunya dalam hal ini yang ingin diketahui adalah data (baik yang belum maupun yang sudah dienkripsi) dalam suatu kumpulan data yang telah diarsipkan.
b. *Replay Attack*
Seseorang bisa merekam pesan – pesan *handshake* dan dapat mengulang pesan – pesan yang telah direkamnya untuk menipu seseorang.
c. *Spoofing*
Penyerang, misalnya B, bisa menyamar menjadi C. semua orang dibuat percaya bahwa B adalah C. Dimana pengendus akan berusaha memanipulasi seluruh komunikasi agar seolah – olah tak ada permasalahan yang di lakukan pada saat melakukan komunikasi, padahal disitulah pengendus melakukan aksinya untuk mencuri data atau informasi.
d. *Man-in-the-middle*
Dalam serangan mitm, seorang attacker akan berada di tengah – tengah komunikasi antara dua pihak. Seluruh pembicaraan yang telah terjadi diantara mereka harus melalui attacker di posisi attacker berada di tengah. Attaker dengan leluasa melakukan penyadapan, pencegatan, pengubahan bahkan memalsukkan komunikasi.

2.3. Sistem Login

Sistem *login* (*login*, juga biasa disebut *login*, *logon*, *signon*, *signin*, *signin*) merupakan proses untuk masuk ke jaringan komputer dengan memasukkan identitas akun minimal terdiri dari username/akun pengguna dan password untuk mendapatkan hak akses (Pribadi 2014). Ketika ingin mengakses sistem, *user*akan diminta memasukkan *userid* dan *password*. Apabila *userid* dan *password* sama seperti yang tersimpan di *server*, maka *user* dapat mengakses sistem tersebut dari akunnya[8].

2.4. Password

a. Pengertian *Password*

Password atau kata sandi merupakan kumpulan dari karakter atau string yang digunakan oleh pengguna jaringan atau sebuah sistem operasi yang mendukung banyak pengguna (multiuser) untuk memverifikasi identitas dirinya kepada sistem keamanan yang dimiliki oleh jaringan atau sistem. *Password* bersifat statis atau sama, maksud disini adalah nilai atau *values* dari *password* tersebut sama dengan *password* sebelumnya hingga *user* menggantinya.

b. Otentikasi *Password*

Authentification adalah proses dalam rangka validasi user pada saat memasuki sistem, nama dan password dari user di cek melalui proses yang mengecek langsung ke daftar mereka yang diberikan hak untuk memasuki sistem tersebut. Biasanya server melakukan penyimpanan password dalam bentuk hash sehingga tidak bisa dikembalikan dalam bentuk plaintext, jadi syarat otentikasi berhasil di atas bisa diartikan sebagai hasil perhitungan hash dari password yang dikirim client harus sama dengan nilai hash yang disimpan dalam server.

2.5. One Time Password

Terdapat tiga pendekatan utama dalam proses Generate OTP, yaitu[6]:

a. OTP berbasis "*mathematicalalgorithm*"

OTP jenis ini merupakan tipe lainnya dari OTP yang menggunakan algoritma matematika kompleks seperti fungsi hash kriptografi untuk membangkitkan password baru berdasarkan password sebelumnya, dan di mulai dari kunci shared rahasia.

b. OTP berbasis "*Timesynchronization*"

OTP jenis ini berbasis sinkronisasi waktu yang berubah secara konstan pada setiap satuan interval waktu tertentu.Didalam token terdapat sebuah jam akurat yang telah disinkronisasikan dengan waktu yang terdapat pada server otentikasi dimana waktu merupakan bagian yang terpenting dari algoritma *password*, karena pembangkitan *password* baru di dasarkan pada waktu saat itu dan bukan pada *password* sebelumnya.Ukuran standar penggunaan waktu pada algoritma ini adalah 30 detik.

c. Berdasarkan "*Challengeresponse*"

Dalam metode ini, *password* baru dibangkitkan oleh suatu *challenge* dari *server*, token pengguna akan mengeluarkan sederetan angka setelah *user* memasukkan *challenge* yang telah diberikan oleh *server*.

3. ANALISA PERANCANGAN PROGRAM

3.1. Rancangan Layar

Melihat proses *login* yang dilakukan lebih mengutamankan waktu dan tidak memerlukan *challenge* sehingga *user* yang memakai nya menjadi sangat mudah. Berikut ini akan menggambarkan sistematik pemakaian aplikasi.

Gambar 3 : Alur Kerja One Time Password

Pada gambar diatas menjelaskan *user* meng-*inputusename* dan *password* pada aplikasi *Mobile* Token *Android* dan mendapatkan 6 *digit* kode OTP untuk *loginwebsite*.

a. Rancangan Layar pada *Mobile* Token

Pada rancangan layar ini *user* diminta menginput *username* dan *password* yang dia miliki untuk mendapatkan kode OTP.

Gambar 4 : Menu Login Mobile Token

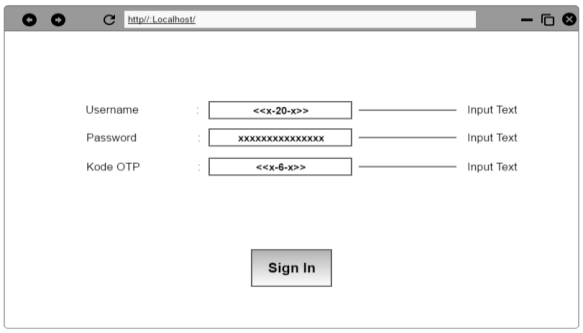
b. Rancangan Layar *OneTimePassword*

Pada rancangan layar ini menjelaskan *user* mendapatkan kode OTP hasil *hashing* dari *username*, *password*, dan waktu ketika *user* melakukan *login*. User dapat menginput kode yang telah didapatkan dari aplikasi OTP di website.

Gambar 5 : Rancangan Layar One Time Password

c. Rancangan Layar *MenuLoginWebsite*

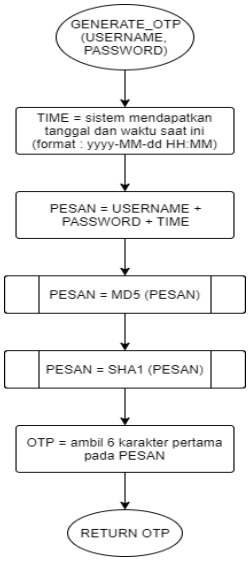
Rancangan layar ini menjelaskan *user* diminta meng-*inputusername* dan *password* yang *user* miliki dan meng-*input* kode OTP yang *user* dapat dari *Mobile* Token.



Gambar 6 : Rancangan Layar Menu Login Website

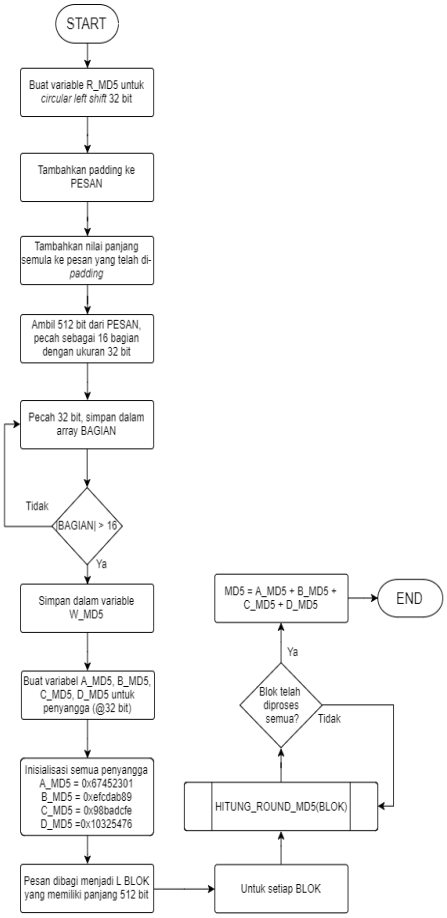
3.2. *Flowchart*

Flowchart atau bagan alur program adalah suatu bagan yang menggambarkan alur alur dari sitem kerja OTP dari awal hingga menuju akhir. Berikut ini adalah *flowchart* yang digunakan pada pembuatan aplikasi enkripsi dan kompresi *file*.



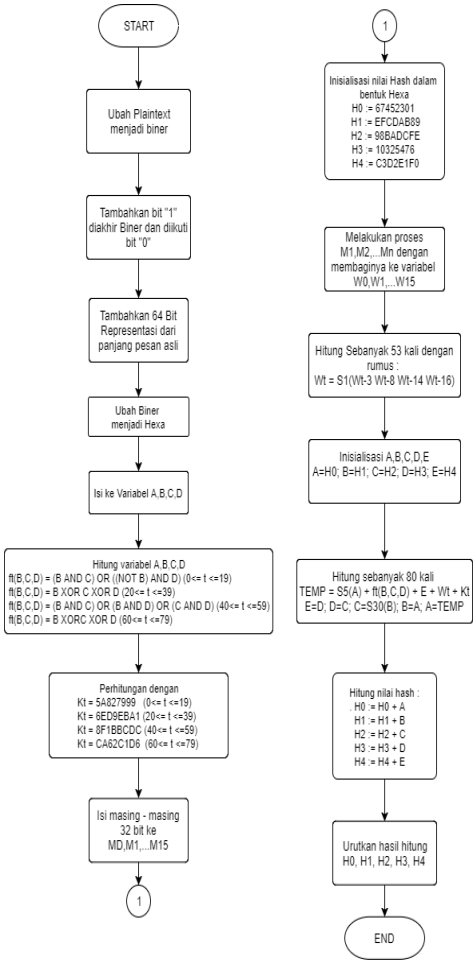
Gambar 7 : Flowchart One Time Password

Flowchart berikut ini akan menjelaskan bagaimana proses *hashing* variabel yang menampung waktu *login*, *username* dan *password* menggunakan algoritma MD5.



Gambar 8 : Flowchart MD5

Flowchart berikut ini akan menjelaskan bagaimana proses hashing variabel yang menampung waktu login, username dan password menggunakan algoritma SHA.

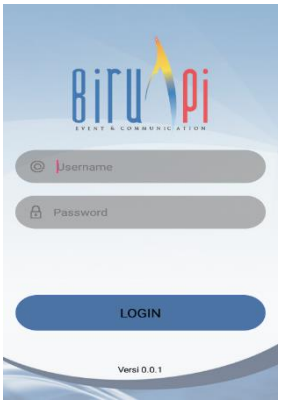


Gambar 9 : Flowchart SHA

4. HASIL DAN PEMBAHASAN

4.1. Tampilan Layar

Tampilkan pada layar *smartphone* terdiri dari beberapa bagian, yaitu tampilan awal ketika *user* meng-*input* kode *username*, *password*, dan tampilan saat kode OTP di bangkitkan.



Gambar 10 : Tampilan Menu Login Mobile Token

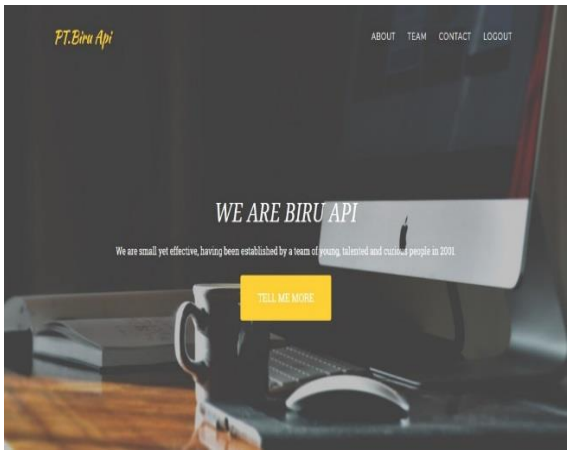
Berikut ini adalah tampilan dari layar *mobile* token otp yang dibangkitkan. pada tampilan layar berikut ini terdapat waktu mundur.

Gambar 11 : Layar Mobile Token telah Dibangkitkan

Pada tampilan layar *menulogin* web berikut ini menjelaskan bahwa terdapat 3 kolom yang tersedia. Kolom tersebut antara lain kolom *username*, *password*, dan kode OTP.

Gambar 12 : Tampilan Layar Menu Login Website

Berikut ini adalah tampilan halaman *website* setelah melakukan proses *login* halaman utama *website*.



Gambar 13 : Tampilan Layar setelah Login Website

4.2. Analisa Hasil

Proses *login* pada *website* melakukan pengecekan *username*, *password* dan kode OTP. Sistem akan akan melakukan pengecekan dan menampilkan validasi dalam bentuk *popup*. Dari proses yang sudah dilakukan dapat diketahui bahwa kode OTP menjadi salah satu bagian terpenting untuk dapat *login* ke halaman utama *website*.

Tabel 2 : Tabel Pengujian One Time Password

Username	Password	Kode OTP	Waktu	Status
hendrik	1234	F9EF27	09:20:43	Berhasil
hendrik	1234	F9EF27	09:21:02	Gagal
admin	admin	EC73B3	09:29:53	Berhasil
admin	admin	EC73B3	09:29:18	Gagal
Bobi_lase	12345	1DB859	09:45:22	Gagal
Bobi_lase	12345	8C6053	09:48:42	Berhasil

4.3. Evaluasi Program

Setelah melakukan pengetesan, ada beberapa kelebihan dan kekurangan dari aplikasi ini, antara lain :

a. Kelebihan Program

1) Dapat digunakan dimana saja dan kapan saja karena mendapatkan kode verifikasi lewat *android*.
2) Aman, karena setiap mendapatkan kode OTP harus melakukan *login* atau identitas rahasia.
3) Pembangkit OTP menggunakan *smartphoneandroid* yang dimiliki semua *user*.
4) Hemat biaya dimana penggunanya hanya menggunakan jaringan *dataseluler*.

b. Kekurangan Program

1) Hanya dapat digunakan pada *smartphoneandroid*.
2) Perbedaan waktu antara *client* dan *server* mempengaruhi keberhasilan proses pembangkit *password*.
3) *User* mendapatkan kode OTP di layar yang sama memungkinkan *hacker* dapat membobol *website*.

5. KESIMPULAN

5.1. Kesimpulan

Menurut dari hasil implementasi yang telah dilakukan, maka dapat diambil beberapa kesimpulan sebagai berikut :

a. Penerapan *OneTimePassword* menggunakan algoritma MD5 dan SHA dapat mengamankan *loginwebsite*.
b. Algoritma MD5 dan SHA dapat digunakan untuk mengimplementasikan *One Time Password* menggunakan *smartphone android* untuk sistem *login user* pada sistem informasi berbasis web agar menjadi lebih aman.
c. Dengan adanya *One Time Password* keamanan sistem *login* terbukti lebih aman dari serangan *hacker*.
d. Pengujian keamanan dari aplikasi yang telah dibuat dengan menggunakan *keylogger* untuk

mendapatkan hasil bahwa walaupun *username* dan *password* berhasil disadap oleh orang lain, tetapi orang tersebut tidak dapat menggunakan nya kembali untuk mengakses sistem *login*.

5.2. Saran

Dalam penelitian *One Time Password* ada beberapa kekurangan yang telah ditemukan. Oleh karena itu, penelitian ini dapat dikembangkan lebih lanjut agar lebih bermanfaat di kemudian hari. Saran untuk penelitian lebih lanjut yang dapat disampaikan adalah sebagai berikut:

a. Mengkombinasikan beberapa algoritma lain untuk mendapatkan kode OTP.
b. Aplikasi tidak dapat digunakan pada *smartphone* dengan sistem operasi selain *android* seperti IOS dan *WindowsPhone*.
c. Menggunakan gmail, sms, dll untuk mendapatkan kode OTP.

6. DAFTAR PUSTAKA

[1] Astuti, R.Y., 2016. "Pengamanan Akses Login Dekstop Menggunakan One Time Password Berdasarkan Time - Based One Time Password dan Advanced Encryption Standard" . Skripsi, Teknologi informasi, Universitas Budi Luhur Jakarta.

[2] Diningrum, Wili Yudha (2017) "Implementasi Mobile Token Dengan Algoritma Message Digest 5 (Md5) Pada Sistem Pengamanan Login One Time Password (Otp) (Studi Kasus Exam-Ict Labkom Universitas Budi Luhur)" Skripsi, Teknologi Informasi, Universitas Budiluhur.

[3] Doly, R (2017) "Implementasi One Time Password Mobile Token Dengan Algoritma Secure Hash Algorithm 1 (SHA) Pada Login Website Pusdakrimti Kejaksaan Agung Republik Indonesia" Skripsi. Teknologi Informasi. Universitas Budiluhur Jakarta.

[4] Munadi, R., Musliyana, Z. and Arif, T. Y. (2016) "Kombinasi Waktu Sinkronisasi dan Nilai Salt untuk Peningkatan Keamanan pada Single Sign-On", Jnteti, Vol.5, No.3, pp. 1–6.

[5] Mustofa, R. P. (2013) "Aplikasi Mobile Android One Time Password (OTP) Untuk Meningkatkan Keamanan Otentikasi". Skripsi. Teknologi Informasi. AMIKOM Yogyakarta, pp.1-15.

[6] Pribadi, A. (2014) "Perancangan Keamanan Sistem Login Aplikasi Multiuser Dengan Algoritma Message Digest 5 (Md5)", Pelita Informatika Budi Darma, 8(Desember), Vol.8, No.2, pp. 172–175.

[7] Vishwakarma, N. and Gangrade, K. (2016) "Secure Image Based One Time Password", Jurnal Ilmu Pengetahuan dan Penelitian Internasional. Vol.5, No.11, pp. 2013–2016.