Implementasi *One Time Password*(OTP) dengan Algoritma *Message Digest 5* (MD5) pada Sistem Login *E-Learning*

Wili Yudha Diningrum[#1], Mardi Hardjianto[#2]

[#] *Fakultas Teknologi Informasi, Universitas Budi Luhur*

Jl. Raya Ciledug, Petukangan Utara, Pesanggrahan, Jakarta Selatan 12260

Telp.(021) 5853753

[1]wiliyudha18@gmail.com

[2]mardi.hardjianto@budiluhur.ac.id

***Abstrak* — Sistem informasi dapat digunakan untuk menunjang kegiatan di dalam suatu organisasi. Sistem informasi menjadi sangat rawan apabila di akses oleh orang yang tidak berkepentingan sehingga dibutuhkan sistem login. Biasanya orang akan menggunakan password yang mudah ditebak oleh orang lain atau menggunakan password statis atau tidak pernah berubah. Selain itu password juga dapat diketahui oleh pihak yang tidak bertanggung jawab dengan cara *sniffing* atau penyadapan. Hal ini membuat password tidak aman dalam mengakses sistem informasi. Untuk itu, agar dapat mencegah penyusupan ke dalam sistem informasi maka dibuat sistem otentikasi berupa password unik dengan hanya sekali pakai yang disebut *One Time Password*. Oleh karena penggunaan smartphone android saat ini semakin meningkat dan sistem operasinya juga selalu mengalami perubahan, maka penelitian ini memanfaatkan smartphone android sebagai mobile token untuk mengimplementasikan one time password. *One time password* in dibangkitkan dengan algoritma Message Digest 5 (MD5) dan sedikit penambahan dari algoritma Time-Based One Time Password (TOTP) yang membangkitkan enam digit bilangan yang akan berubah setiap 60 detik. Penelitian ini mampu memberikan keamanan pada saat akses login pada sistem informasi karena kode yang dihasilkan dari one time password ini selalu berubah dan tidak dapat digunakan kembali untuk mengakses sistem bila status kode one time password tidak valid sehingga sistem tidak dapat dengan mudah dibobol oleh orang lain**

Kata Kunci : Message Digest 5, Mobile Token, Time-Based One Time Password

Abstract* — Information systems can be used to support *activities within an organization. Information system becomes very vulnerable if in access by an unauthorized person, so that required login system. Usually, people will use passwords that are easily guessed by others or use static passwords or never changed. Besides, passwords can also be known by irresponsible parties by sniffing or tapping. The impact of sniffing makes the password method unsafe in accessing the information system. Infiltration into the information system can be avoided by creating a password authentication system that can be used only once. This method is called One Time Password. The current use of Android smartphones is increasing and the operating system is also constantly changing. This research utilizes Smartphone Android as mobile token to implement one-time password. One time password is generated using the Message Digest 5 (MD5) algorithm and is combined with Time-based One Time Password (TOTP) algorithm that makes a six-digit number that will change every 60 seconds. This research can provide security at the time of login access at information system because the code generated from one-time password is always changed and can only be used once. The proposed system can not be easily broken into by others.

Keywords: Message Digest 5, Mobile Token, Time-Based One Time Password

I. PENDAHULUAN

Sistem informasi memang menguntungkan dan dapat meningkatkan kinerja dari semua komponen organisasi atau perusahaan. Namun, keamanan sistem informasi yang berbasis *web* sangat rawan untuk di sadap oleh pihak yang tidak bertanggung jawab. Banyak metode yang sering dipakai oleh *hacker* untuk dapat mengetahui *username* dan *password* dari sebuah akun (*account*). Akun di sini dapat berupa akun apa saja, seperti akun *e-mail,* akun jejaring sosial, akun *messager,* dan lain sebagainya [1].

Perkembangan sistem informasi berbasis *web* dalam bidang pendidikan saat ini pun sudah semakin pesat. Berbagai

jenis aktifitas pendidikan telah dijalankan dengan memanfaatkan teknologi ini. *E-learning* atau *electronic learning* adalah sistem pendidikan yang menggunakan aplikasi elektronik untuk mendukung belajar mengajar dengan memanfaatkan media *internet,* jaringan komputer, maupun komputer *standalone* [2]. *E-Learning* menawarkan banyak kemudahan dalam banyak hal, seperti penilaian, efisiensi ruang dan alat tulis menulis. Sistem *E-Learning* harus terhubung dengan internet yang merupakan jaringan publik dengan kata lain semua orang bisa mengaksesnya. Saat memulai menggunakan *E-learning* setiap *user* harus melakukan proses *login* dengan memasukkan *username* dan *password*. Banyak sistem *E-Learning* masih menggunakan protokol HTTP dalam proses pengiriman data *username* dan *password*. Jika *user* menggunakan *password* yang selalu sama untuk masuk ke dalam suatu sistem, hal itu dapat menyebabkan *password* tersebut menjadi rentan terhadap *sniffer* jaringan melakukan *replay attack* [3]. Untuk mengamankan *username* dan *password*, maka proses login perlu ditingkatkan keamanannya dengan *two factor authentication*(2FA). Two factor authentication mengharuskan pengguna melalui dua proses otentikasi, yaitu password dan security token yang dapat berubah dalam jangka waktu tertentu dan hanya sekali digunakan (Time-based One-Time Password).

Pada makalah ini, pengamanan username dan password ditambahkan dengan Time-based One-Time Password dimana nilai OTP ini dihasilkan menggunakan algoritma *Message Digest* 5 (MD5) dengan parameternya adalah waktu.

II. TINJAUAN STUDI

2.1. Studi Literatur

Ada beberapa penelitian yang telah dilakukan oleh beberapa peneliti sebelumnya. Perolehan nilai OTP dapat beraneka ragam. Ada yang menggunakan hash SHA [4], MD5[5], kombinasi MD5 dan SHA[6]. Ada juga yang menggunakan hash yang digabung dengan enkripsi AES[7]. Dari nilai hash yang dihasilkan, diambil enam digit yang akan digunakan sebagai nilai OTP.

2.2. Sistem *Login*

Sistem *login* merupakan proses masuk ke jaringan komputer dengan memasukkan identitas akun minimal terdiri dari *username/*akun pengguna menggunakan *password* untuk mendapatkan akses. Antara *username* dan *password* keduanya saling terkait dan tidak bisa dipisahkan, biasanya *username/*akun pengguna tidak pernah diubah karena merupakan identitas unik tetapi *password*/kata sandi dapat diubah sesuai keperluan untuk menjaga keamanan akun.

2.3. Password

Password merupakan sederet karakter bisa simbol, huruf dan angka yang memuat informasi penting untuk melakukan proses *authentifikasi*, yaitu proses sistem untuk memastikan bahwa orang yang mengakses sistem tersebut adalah orang yang sebenarnya dan bukan orang lain atau bahkan robot. Otentikasi (*Authentication*) adalah proses untuk memastikan bahwa kedua ujung koneksi dalam keadaan benar atau sama. Seperti *password* pada umumnya, syarat agar otentikasi berhasil adalah *password* yang dikirimkan *client* harus sama dengan *password* yang disimpan di *server*. Dengan alasan keamanan jarang sekali *server* menyimpan *password user* dalam bentuk *plain-text.* Biasanya *server* menyimpan *password user* dalam bentuk *hash* sehingga tidak bisa dikembalikan dalam bentuk *plain-text.* Jadi syarat otentikasi berhasil di atas bisa diartikan sebagai hasil penghitungan *hash* dari *password* yang dikirim klien harus sama dengan nilai *hash* yang disimpan dalam *server.* Ada beberapa upaya untuk mengamankan proteksi *password* [8], antara lain:

a. *Salting*

String password yang diberikan pemakai ditambah suatu *string* pendek sehingga mencapai panjang *password* tertentu.

b. *One Time Password*

Password yang dimiliki oleh pemakai diganti secara teratur, dimana seorang pemakai memiliki daftar *password* sendiri sehingga untuk *login* ia selalu menggunakan *password* berikutnya. Dengan cara ini, pemakai akan menjadi lebih direpotkan karena harus menjaga dan mengingat daftar *password* tersebut agar tidak sampai tercuri atau hilang.

c. *One Question & Long Answer*

Cara ini mengharuskan pemakai memberikan satu pertanyaan yang panjang beserta jawabannya. Pertanyaan dan jawaban tersebut dapat dipilih oleh pemakai dan sebaiknya mudah untuk diingat sehingga tidak perlu menuliskannya pada kertas.

d. *Response*

Pemakai diberikan kebebasan untuk menggunakan satu atau beberapa algoritma sekaligus.

1) *One Time Password*

One Time Password (OTP) merupakan metode otentikasi yang menggunakan *password* yang selalu berubah setelah setiap kali *login*, atau berubah setiap interval waktu tertentu. Beberapa pendekatan proses *generate* OTP:

1) *mathematical algorithm*
2) *time-synchronization*
3) *Challenge response*

2) Kriptografi

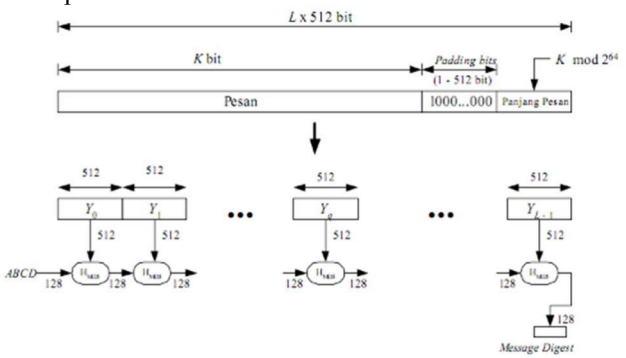
Kriptografi merupakan ilmu yang digunakan untuk menjaga kerahasiaan pesan dengan cara menyandikannya dalam bentuk yang tidak dapat dimengerti maknanya. Di dalam kriptografi kita akan sering menemukan berbagai istilah atau *terminology.* Beberapa istilah yang harus diketahui yaitu :

1) Pesan, *Plainteks*, dan *Cipherteks*
2) Pengirim dan Penerima
3) Enkripsi dan dekripsi
4) *Chipher* dan kunci

3) *Message Digest*

MD5 adalah fungsi *hash* satu-arah yang dibuat oleh Ron Rivest [9]. MD5 merupakan perbaikan dari MD4 setelah MD4 berhasil diserang oleh kriptanalis.

Algoritma MD5 menerima masukan berupa pesan dengan ukuran sembarang dan menghasilkan *message digest* yang panjangnya 128 bit. Gambaran pembuatan *message digest dengan* algoritma MD5 diperlihatkan pada Gbr 22



Gbr 22. Pembuatan Message Digest dengan algoritma MD5

Langkah-langkah pembuatan *message digest* secara garis besar adalah sebagai berikut:

a) *Padding bits*

Pesan ditambah dengan sejumlah bit pengganjal sedemikian sehingga panjang pesan (dalam satuan bit) kongruen dengan 448 modulo 512. Ini berarti panjang pesan setelah ditambahi bit-bit pengganjal adalah 64 bit kurang dari kelipatan 512. Angka 512 ini muncul karena MD5 memperoses pesan dalam blok-blok yang berukuran 512. Pesan dengan panjang 448 bit pun tetap ditambah dengan bit- bit pengganjal. Jika panjang pesan 448 bit, maka pesan tersebut ditambah dengan 512 bit menjadi 960 bit. Jadi, panjang bit-bit pengganjal adalah antara 1 sampai 512. Bit-bit pengganjal terdiri dari sebuah bit 1 diikuti dengan sisanya bit 0.

b) *Penambahan* nilai panjang pesan semula

Pesan yang telah diberi bit-bit pengganjal selanjutnya ditambah lagi dengan 64 bit yang menyatakan panjang pesan semula. Jika panjang pesan > $\mathbf{2^{64}}$ maka yang diambil adalah panjangnya dalam modulo $\mathbf{2^{64}}$. Dengan kata lain, jika panjang pesan semula adalah *K* bit, maka 64 bit yang ditambahkan menyatakan *K* modulo $\mathbf{2^{64}}$. Setelah ditambah dengan 64 bit, panjang pesan sekarang menjadi 512 bit.

c) *Inisialisasi* penyangga (*buffer*) *MD*

MD5 membutuhkan 4 buah penyangga (*buffer*) yang masing- masing panjangnya 32 bit. Total panjang penyangga adalah 4 × 32 = 128 bit. Keempat penyangga ini menampung hasil antara dan hasil akhir. Keempat penyangga ini diberi nama A, B, C, dan D. Setiap penyangga diinisialisasi dengan nilai-nilai (dalam notasi HEX) sebagai berikut:

A = 01234567
B = 89ABCDEF
C = FEDCBA98
D = 76543210

Pengolahan pesan dalam blok berukuran 512 bit

2.4. *Time-based One-Time Password*

Time-based one time password merupakan algoritma perluasan dari algoritma *one time password* yaitu *HMAC-based One Time Password* (HOTP). Algoritma TOTP menentukan *one time password* berbasis waktu. Algoritma TOTP berbeda dengan algoritma HOTP karena algoritma HOTP merupakan algoritma yang menentukan *one time password* berbasis *event*. Algoritma TOTP memberikan nilai OTP berumur pendek atau berdasarkan interval waktu tertentu untuk meningkatkan keamanan. Nilai TOTP dihitung dengan menggunakan rumus berikut :

$$TOTP(K,T) = HOTP(K,T)$$
$$where\ T = \lfloor \frac{(T1 - T0)}{X} \rfloor$$

Parameters yang digunakan dalam algoritma TOTP adalah sebagai berikut :

K = *Secret-Key*.
T = waktu-*counter*.
X = interval waktu dalam detik.
T0 = waktu ketika kita mulai menghitung waktu X.
T1 = waktu saat ini.

III. RANCANGAN *ONE TIME PASSWORD*

3.1. Penyelesaian Masalah

Penelitian ini menggunakan metode *One Time Password* berbasis sinkronisasi waktu atau *Time-Synchronization* yang akan berubah secara konstan pada interval waktu tertentu. Proses ini memerlukan sinkronisasi antara *token* milik *client* dengan *server* otentikasi. Di dalam *token* terdapat sebuah jam akurat yang telah disinkronisasi dengan waktu *server* otentikasi. Pada *One Time Password* waktu merupakan bagian terpenting dari algoritma *one time password,* karena pembangkitan *password* baru didasarkan pada waktu saat itu dan bukan waktu pada *password* sebelumnya.

Pada penelitian ini akan dibuat dua aplikasi yaitu aplikasi *login web* dan aplikasi untuk menmbangkitkan *One Time Password* pada *smartphone* berbasis Android. Proses dari aplikasi yang akan dibuat dimulai dari pengisian *username* dan *password* pada aplikasi Android. Pada aplikasi tersebut akan menghasilkan kode *generate* yang akan digunakan sebagai kode *One Time Password.* Kemudian, *user login* pada *website* dengan memasukkan *username, password,* dan kode *generate* yang sudah didapatkan dari Android lalu memilih *button login.*

Metode pembangkit kode *One Time Password* (OTP) pada penelitian ini menggunakan metode *Time-based One Time Password* (TOTP) dan algoritma *Message Digest* 5 (MD5). Sementara model yang digunakan untuk pembangkitan kode menggunakan *self-generated.* Kode OTP dibangkitkan berdasarkan waktu dan *username* pada saat OTP diminta dibangkitkan. *User* hanya perlu memasukkan kode *generate* yang ada pada *mobile token.* Kemudian *Mobile token* akan mengeluarkan enam digit kode yang berbeda-beda secara periodik ketika *mobile token* diminta untuk menghasilkan kode *self-generated*. Metode ini lebih sesuai untuk diterapkan pada

pnelitian ini karena proses *login* akan menjadi lebih mudah dan lebih cepat. Sehingga *user* tidak perlu melakukan *input* kode OTP berulang-ulang, seperti pada proses pembangkitan kode otp dengan menggunakan *challenge response*.

Metode pembangkit kode *One Time Password* (OTP) menggunakan metode *Time-based One Time Password* (TOTP) dan algoritma *Message Digest* 5 (MD5). Kode OTP dibangkitkan berdasarkan waktu dan *username* pada saat OTP diminta dibangkitkan. *User* hanya perlu memasukkan kode *generate* yang ada pada *mobile token.* Kemudian *Mobile token* akan mengeluarkan enam digit kode yang berbeda-beda secara periodik ketika *mobile token* diminta untuk menghasilkan kode *self-generated*.

3.2. Algoritma Proses One Time Password

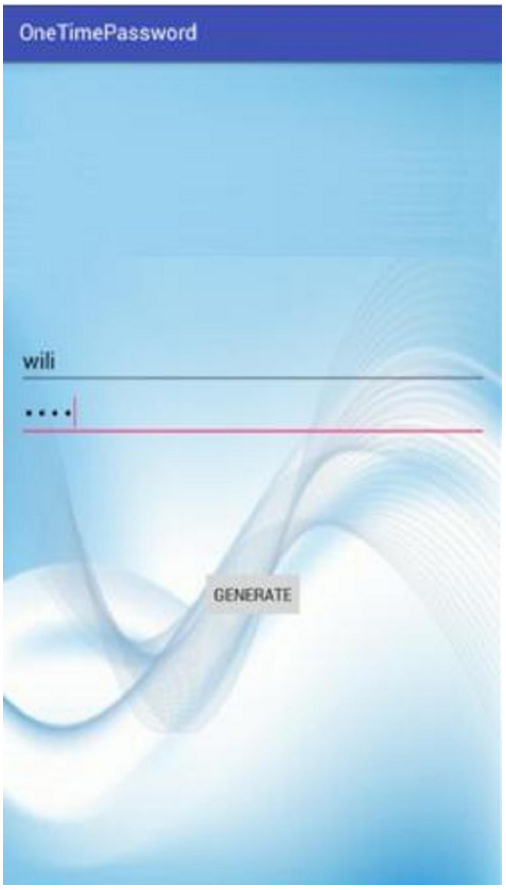
Berikut ini adalah algoritma untuk menghasilkan One-Time Password

1. Ambil *username* saat *login* masukkan *variable* U
2. Ambil waktu saat *user login* masukkan *variable* WM
3. Buat variabel r untuk operasi circular left shift 32 bit
4. For I from 0 to 63
5. K[i] ← floor(abs(sin(i+1))x2^32)
6. End for
7. Buat *variable* h0, h1, h2, h3 untuk nilai hash
8. Tambahkan nilai panjang semula ke dalam Tgl yang telah dibeeri padding bits
9. Foreach 512 bit dari U+WM
10. Pecah menjadi 32 bit sebanyak 16 masukkan ke variabel w
11. End foreach
12. Buat variabel a, b, c, d untuk inisialisasi penyangga
13. For i from 0 to 63
14. If 0 ≤ I ≤ 15 Then
15. f ← (b and c) or ((not b) and d)
16. g ← i
17. Else if 16 ≤ I ≤ 31 Then
18. f ← (d and b) or ((not d) and c)
19. g ←(5*i+1) mod 16
20. Else if 32 ≤ I ≤ 47 Then
21. f ←b xor c xor d
22. g ←(3*i+5) mod 16
23. Else if 48 ≤ i ≤ 63 Then
24. f ← c xor (b or (not d))
25. g ← (7 * i) mod 16
26. End if
27. End for
28. Temp ← d
29. d ←c
30. c ← b
31. b ← ((a+f+k(i)+w(g)leftrotater r(i)) + b
32. a←temp
33. Tambahkan nilai dengan penyangga, masukkan ke variabel hasil
34. Gabungkan nilai dari variabel hasil
35. Ambil 6 karakter pertama.

IV. Hasil Dan Pembahasan

4.1. Tampilan Layar Menu Login Mobile Token

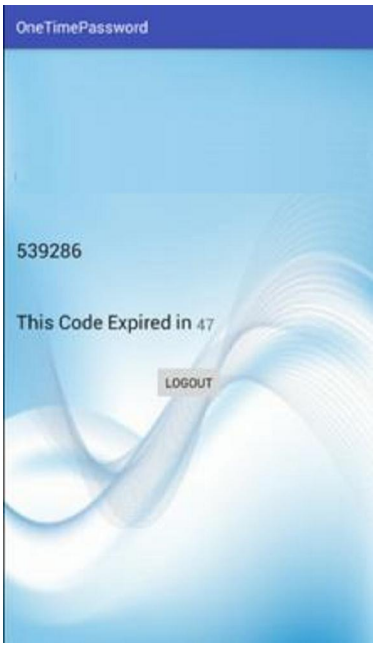
Gbr 23 menunjukkan tampilan awal ketika *user* meminta kode OTP, disini ada dua *textfield* untuk memasukkan *username* dan *password* serta terdapat tombol *generate* yang digunakan untuk membangkitkan OTP.



Gbr 23. Tampilan Layar Menu Login Mobile Token

4.2. Tampilan Layar *Mobile Token* OTP Dibangkitkan

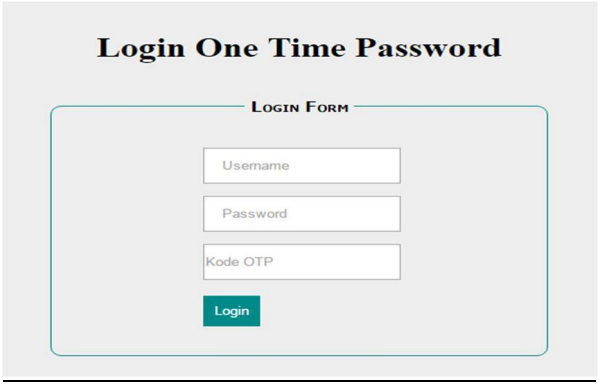
Apabila username dan password sesuai maka akan menampilkan kode OTP beserta masa aktif dari kode OTP. Pada Gbr 24 ini terdiri dari enam *digit number* yang merupakan kode OTP dan *digit* angka yang menyatakan masa aktif OTP yang bersatuan detik. Masa aktif kode OTP maksimal 60 detik. Kode OTP akan berubah-ubah dengan sendirinya setiap 60 detik. Kode OTP inilah yang akan diisikan *user* ketika akan masuk ke dalam sistem aplikasi *web*.



Gbr 24. Tampilan Layar Mobile Token Dibangkitkan

4.3. Tampilan Layar Menu Login Web

Apabila *user* memasukkan data *username, password,* dan kode OTP lalu menekan *buton login* maka sistem akan melakukan pengecekan apakah *username* dan *password* sesuai dengan data yang tersimpan dalam *databse.* Sistem juga akan membandingkan apakah kode OTP yang dimasukkan *user* sama dengan OTP yang didapatkan dari perhitungan *server* apabila hasilnya sama maka otentikasi berhasil. Menu login web ditunjukkan pada Gbr 25



Gbr 25. Tampilan Layar Menu Login Web

V. KESIMPULAN

Berdasarkan hasil implementasi dan pengujian pada bab sebelumnya, maka dapat diambil beberapa kesimpulan sebagai berikut :

1) Penambahan kode OTP dapat mengamankan akses *login* sistem dari teknik pencurian dari *hacker* selain itu penambahan *One Time Password* dapat menanggulangi kemungkinan penyalahgunaan hak akses *user* karena kode OTP yang digunakan *user* bersifat dinamis dan hanya dapat digunakan satu kali otentikasi dengan batas waktu tertentu.
2) Pengujian keamanan dari aplikasi yang telah dibuat dengan menggunakan cara *sniffing* mendapatkan hasil bahwa walaupun *password* berhasil disadap oleh orang lain tetapi orang tersebut tidak dapat menggunakannya kembali untuk mengakses sistem *login*.

DAFTAR PUSTAKA

[1] K. Imam dan E. Sediyono, "Studi Pengamanan Login Pada Sistem Informasi Akademik Menggunakan Otentifikasi One Time Password Berbasisis SMS dengan Hash MD5," *J. Sist. Inf. Bisnis*, vol. 01, hal. 7–12, 2013.

[2] A. Shaugi, "Analisa Dan Perbandingan Hasil Implementasi Algoritma Md5 Dan Sha-1 Pada Sistem Keamanan Algoritma Md5 Dan Sha-1 Pada Sistem Keamanan Autentikasi Simple-O," UNIVERSITAS INDONESIA, 2012.

[3] E. C. Simamora, "IMPLEMENTASI METODE AUTENTIKASI ONE TIME PASSWORD (OTPA) BERBASIS MOBILE TOKEN PADA APLIKASI UJIAN ONLINE(STUDI KASUS: JURUSAN MATEMATIKA FMIPA UNIKA)," Universitas Lampung, 2012.

[4] K. I. Santoso, "Dua Faktor Pengamanan Login Web Menggunakan Otentikasi One Time Password Dengan Hash SHA," *Semin. Nas. Teknol. Inf. Komun. Terap. 2013*, vol. 2013, no. November, hal. 204–210, 2013.

[5] K. I. Santoso, E. Sediyono, dan Suhartono, "Studi Pengamanan Login Pada Sistem Informasi Akademik Menggunakan Otentifikasi One Time Password Berbasis SMS dengan Hash MD5," *J. Sist. Inf. Bisnis*, vol. 01, hal. 7–12, 2013.

[6] W. Sentosa, "Implementasi Token Berbasis Waktu dengan Fungsi Hash untuk Mengotentikasi Transaksi E-Banking," 2016.

[7] R. Y. Astuti, "PENGAMANAN AKSES LOGIN DESKTOP MENGGUNAKAN ONE TIME PASSWORD BERDASARKAN TIME-BASED ONE TIME PASSWORD DAN ADVANCED ENCRYPTION STANDARD," in *Statewide Agricultural Land Use Baseline 2016*, 2016, vol. 1, hal. 1–68.

[8] J. J. Malik, *BEST TOOLS HACKING & RECOVERY PASSWORD*, 1 ed. Yogyakarta: C.V ANDI OFFSET(Penerbit ANDI), 2009.

[9] R. Munir, "Fungsi Hash Satu-Arah dan Algoritma MD5 17," hal. 0–17, 2004.