Implementasi One Time Password pada Sistem Login dengan Algoritma SHA-256 dan DES pada Aplikasi EO Blucampus Berbasis Client Server

Anggit Prayogo[1)], **Muhammad Ainur Rony**[2)]
Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur
Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260
E-mail : anggitprayogo@gmail.com [1)], ainur.rony@gmail.com [2)]

Abstrak

Otentikasi pada suatu sistem diperlukan untuk mencegah orang yang tidak memiliki kepentingan maupun otorisasi terhadap sistem tersebut. Bagian sistem yang sangat penting untuk menentukan seseorang memiliki otorisasi terhadap suatu sistem adalah login. Penggunaan password yang sama setiap kali melakukan login merupakan kelemahan yang dapat dimanfaatkan oleh peretas untuk memperoleh akses ke dalam sistem. Saat ini, aplikasi EO Blucampus masih menggunakan metode login tradisional, dengan pengunaan password yang sama setiap kali melakukan login. Hal ini tentu perlu adanya solusi dalam mengantisipasi permasalahan otentikasi sistem. Salah satu metode yang dapat digunakan adalah One Time Password. Pada penelitian ini menggunakan Algoritma SHA-256 untuk membangkitkan kode OTP dan DES untuk pengamanan data yang berjalan diatas jaringan internet. Pembankitan OTP dengan algoritma hash SHA-256 dilakukan di server, kode OTP yang telah dibangkitkan dikenakan algoritma enkripsi DES untuk kemudian dikenakan fungsi dekripsi di sisi client. Hasil dari penilitian ini menunjukan Kode OTP yang dibangkitkan dengan Algoritma hash SHA-256 tersebut unik dalam 20 kali percobaan. Dengan rata-rata waktu enkripsi dan dekripsi DES adalah 263,55 ms dan 199,25 ms.

Kata kunci: *One Time Password, SHA-256, DES, Android, Web Service*.

1. PENDAHULUAN

Perkembangan teknologi yang pesat memungkinkan setiap orang untuk dapat bergerak lebih cepat dalam memecahkan permasalahan di setiap segi kehidupan dengan kemudahan mengakses berbagai informasi. Informasi dapat diperoleh dengan mudah, murah, dan praktis melalui *smartphone*. Hal tersebut pula yang menyebabkan perkembangan aplikasi *mobile* dengan sangat pesat. Setiap pengembang aplikasi *mobile* tentu menawarkan produk dengan harapan untuk dapat mempermudah pekerjaan dan memenuhi kebutuhan pengguna. Banyak faktor yang perlu diperhatikan untuk dapat menghasilkan aplikasi *mobile* yang baik, salah satunya adalah faktor keamanan. Salah satu masalah keamanan yang sering terjadi pada sistem yang berjalan diatas jaringan internet adalah proses otentikasi pengguna oleh sistem. Otentikasi berhubungan dengan identifikasi atau pengenalan, baik secara kesatuan sistem maupun informasi itu [5].

Proses *login* merupakan bagian dari sistem yang malakukan otentikasi terhadap pengguna. Proses login yang masih menggunakan *password* yang sama setiap kali melakukan *login* ke dalam sebuah sistem. Hal ini masih kurang aman untuk melindungi *user* dari pihak yang tidak bertanggung jawab. Oleh karena itu, untuk meminimalisir terjadinya penggunaan sistem oleh orang yang tidak memiliki otorisasi perlu digunakan metode *one time password* pada sebuah sistem *login*. *one time password* adalah sebuah password yang hanya berlaku untuk sesi login tunggal atau transaksi tunggal [4].

Aplikasi *mobile EO* BluCampus merupakan aplikasi yang dapat mempermudah seorang EO untuk dapat mengelola acara pada Blucampus. Aplikasi EO Blucampus saat ini masih menggunakan metode *static password* dalam melakukan proses *login* untuk dapat mengakses halaman EO BluCampus. Pengguna hanya perlu memasukan *username* dan *password* saat melakukan *login*. Proses *login* dengan metode *static password* masih memiliki banyak kekurangan. Ada banyak cara yang digunakan oleh *hacker* untuk dapat mengetahui *username/email* dan *password* dari sebuah akun. Salah satu cara yang dapat digunakan oleh *hacker* untuk dapat mengetahui informasi seseorang adalah *sniffing*.

One Time Password diterapkan pada sistem *login* EO Blucampus. Pengguna diharuskan melalui dua tahap proses otentikasi. Pertama, pengguna diharuskan memasukan *username* dan *password*. Jika data otentik, maka pengguna diharuskan melakukan verifikasi kode OTP yang didikirimkan oleh *server* ke pengguna melalui SMS atau *email*. Jika kode *valid*, maka pengguna akan diarahkan ke halaman utama EO Blucampus. Pada penerapan OTP ini, digunakan algoritma *hash* SHA-256 untuk membangkitkan kode OTP dari data pengguna seperti nomor *handphone*, *username* dan waktu akses pengguna. Digunakan pula algoritma enrkripsi dan dekripsi pada kode OTP yang telah dibangkitkan dengan algoritma DES. Penerapan algoitma DES pada kode OTP ini bertujuan untuk mengamankan kode yang akan berjalan melalui jaringan internet [3].

Pada uraian di atas, penulis dapat merumuskan masalah yang akan dibahas dalam penilitian ini adalah Bagaimana solusi pengamanan sistem login dengan metode One Time Password (OTP) pada aplikasi Event Organizer (EO) Blucampus

menggunakan algoritma SHA-256 sebagai pembangkit OTP dan DES untuk pengamanan data ketika dikirim dari server ke client ? dan Bagaimana solusi untuk mengembalikan data yang telah dienkripsi menjadi data asli tanpa mengalami perubahan? Jika permasalahan tersebut terpacai maka akan tercapai tujuan dari penilitian ini yaitu merancang model sistem keamanan *password* dengan metode *One Time Password* (OTP) agar *Event Orginzer* (EO) lebih aman dalam mengakses aplikasi EO BluCampus.

2. METODE PENELITIAN

2.1. Metode Penelitian

Berikut ini adalah rincian tahapan dalam pembuatan aplikasi pengamanan sistem *login* dengan metode *one time password* pada aplikasi EO Blucampus yaitu:

a. Pengumpulan Data

Mengumpulkan data yang dibutuhkan dari keseluruhan elemen sistem yang akan di aplikasikan ke dalam bentuk perangkat lunak, dan mengumpulkan data mengenai Aplikasi EO BluCampus, One Time Password, serta algoritma hash SHA-256 dan Data Encyption Standard (DES).

b. Menganalisa Kebutuhan Aplikasi

Setelah memperoleh kebutuhan aplikasi kemudian dipelajari dan dianalisa mengenai fungsi-fungsi apa saja yang diperlukan untuk mengimplementasikan aplikasi ini.

c. Desain atau Perancangan Aplikasi

Di dalam tahap ini akan dilakukannya proses penulisan kode (coding) dalam bahasa pemrograman yang telah ditentukan, yakni PHP dengan framework Laravel dan menggunakan database MariaDB versi 10.1.21 serta dijalankan menggunakan web browser dan Web Server Apache. Android untuk sisi client.

d. Pengkodean

Pengkodean dilakukan untuk memudahkan dalam mengimplementasikan rancangan aplikasi kedalam algoritma hash SHA-256 dan DES dengan menggunakan bahasa pemrograman PHP degan framework Laravel dan Android. Melakukan pembangkitan kode OTP dengan algoritma SHA-256 dan dikenakan enkripsi DES dengan bahasa pemrograman PHP (sisi server) dan melakukan dekripsi pada sisi android (client).

e. Implementasi

Rancangan aplikasi yang sudah dibuat kemudian diimplementasikan berdasarkan analisis masalahnya./

f. Pengujian

Pengujian dilakukan setelah aplikasi selesai dibuat dengan melakukan beberapa pengujian program dan mencari kesalahan pada program hingga tidak ada lagi kesalahan program dan program sudah berjalan sesuai dengan yang dirancang.

2.2. Spesifikasi Basis Data

Berikut adalah rincian struktur dari tabel-tabel yang digunakan dalam *database* aplikasi ini :

Tabel 1. Spesifikasi Tabel *User*

No	Nama Field	Type	Lebar
1	Id	Int	10
2	Username	Varchar	25
3	Name	Varchar	255
4	Email	Varchar	255
5	Tanggal_lahir	Date	8
6	Alamat	Text	200
7	No_telp	Varchar	191
8	Password	Varchar	255
9	Is_permission	tinyint	4

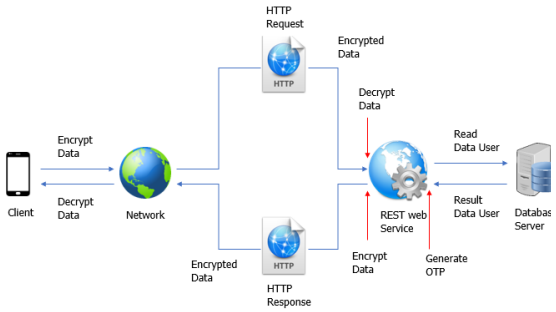
Tabel 2. Spesifikasi Tabel Acara

No	Nama Field	Type	Lebar
1	Id	Int	10
2	Id User	Int	11
3	Judul Acara	Varchar	191
4	Isi Acara	Text	500
5	Tempat Acara	Varchar	191
6	Contact Acara	Varchar	191
7	Jumlah Peserta	Int	10
8	Tanggal Acara	date	8
9	Jam Acara	Varchar	191
10	Foto Acara	Varchar	191
11	Status Acara	varchar	191

Tabel 3. Spesifikasi Tabel Pendaftaran

No	Nama Field	Type	Lebar
1	Id	Int	10
2	Id User	Int	10
3	Id Acara	Int	10
4	Status	Enum	11
5	Kode	varchar	255

2.3. Rancangan Aplikasi Program



Gambar 1. Racangan Aplikasi Program

Berdasarkan dari rancangan aplikasi diatas, user mengirimkan data berupa username dan password yang telah dienkripsi menggunakan algoritma DES dan Deffie-Helman sebagai pembangkitan kuncinya. Algoritma DES merupakan algoritma enkripsi yang paling banyak digunakan di dunia yang diadopsi oleh NIST (*National Institue of Standards and Technology*) sebagai standar pengolah informasi Federal AS. Data *plaintext* dienkrip dalam blok-blok 64 bit menjadi 64 bit data *ciphertext* menggunakan kunci 56 bit kunci internal (*internal key*) [1]. Data yang dikirimkan akan diterima oleh Web Server. Di server, data tersebut akan didekripsi oleh algoritma DES. *Web Service* akan malakukan query ke database untuk mendapatkan data user yang akan digunakan sebagai bahan untuk membangkitkan kode OTP. *Web Service* digunakan sebagai suatu fasilitas yang menyediakan layanan (dalam bentuk informasi atau data) kepada sistem lain, sehingga dapat berinteraksi dengan sistem tersebut melalui layanan-layanan yang disediakan [2]. Kode OTP dibangkitkan dengan menggunakan algoritma SHA-256. Kode OTP yang telah dibangkitkan bedasarkan data user akan dienkripsi dengan algoritma DES. Kode yang telah ternkripsi akan dikirimkan ke client sebagai response proses login. Kode OTP yang telah diterima client akan didekripsi kembali dengan algoritma DES, lalu disimpan dalam sebuah variabel untuk kemudian dikirim ke halaman verifikasi OTP menggunakan Intent.

2.4. Rancangan *Web Service*

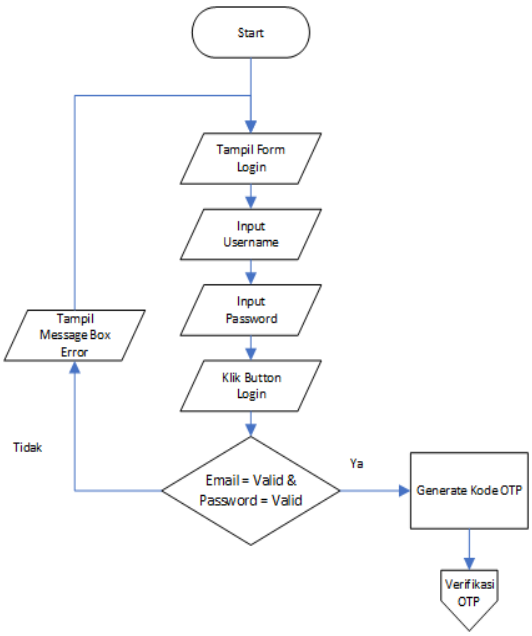
Rancangan *web service* ini berisi layanan yang akan digunakan pada pada sistem OTP Blumcapus EO, beserta penjelasan dari masing-masing layanan seperti nama layanan, nama fungsi, parameter, dan keluaran yang akan diuraikan pada Tabel 4.

Tabel 4. Tabel Rancangan Web Service

Nama Layanan	Fungsi	Path	Parameter
Registrasi untuk EO blucampus	POST	/regis_eo	Request body parameter : Username, email, name, password, alamat, telp
Login untuk EO blucampus	POST	/login	Request body parameter : Password, email
Melakukan request ulang OTP	POST	/retry_otp	Request body parameter :Password, email
Mendapatkan daftar seminar bedasarkan id user EO	Get	/events/{id}	Id_user
Mendapatkan detail acara dan user yang telah terdaftar pada acara tersebut berdasarkan id acara	Get	/events/detail/{id}	Id_acara
Melakukan absensi terhadap user yang terdaftar pada acara tertentu	Post	/events/status	Request body paramter : Kode_acara, id_acara
Mendapatkan semua event yang telah dibuat oleh semua EO	Get	/allevents	
Mendapatkan semua event yang telah diikuti oleh user tertentu berdasarkan id_user	Get	/events/myevents/{id}	Id_user
Mendapatkan detail acara event yang telah diikuti oleh user tertentu	Get	/events/myevents/{id_user}/{id_acara}	Id_user, id_acara

2.5. Flowchart dan Algoritma Program

Berikut ini merupakan rancangan *flowchart* dan algoritma pemrograman. Terdiri dari Halaman *Login* dan Halaman Verifikasi OTP yang merupakan halaman inti dari program ini. Pada Halaman *Login* terdapat inputan *username* dan *email.* Sedangkan pada Halaman Verifikasi OTP, terdapat inputan kode OTP dan tombol kirim ulang. Berikut *flowchart* dan algoritmanya :

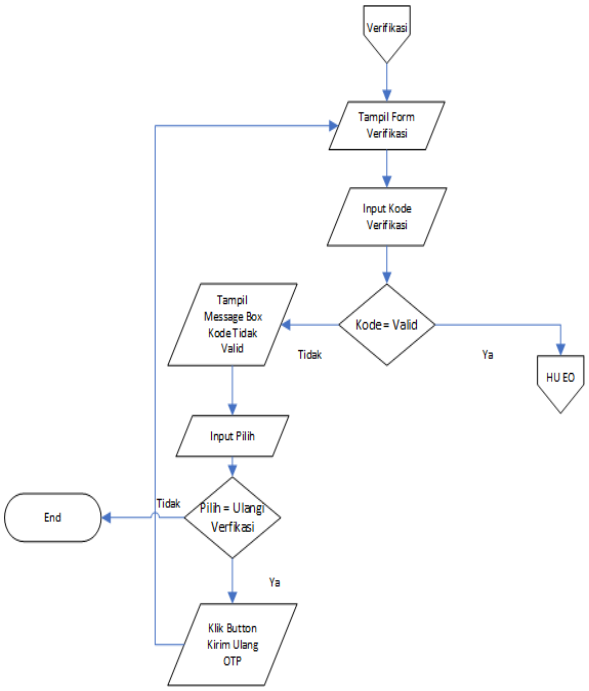


Gambar 2. Flowchart Halaman Login EO Blucampus

Algoritma proses dari *flowchart* di atas :

1.Tampil Form Login
2. Input Email
3. Input Password
4. IF Login then
5. Kirim Email dan Password Ke Server
6 IF Valid then
7. Server Generate OTP Dengan SHA-256
8. Ambil 6 Angka Awal Hasil Hash
9. Kirim Kode OTP melalui SMS dan Email ke Pengguna
10 Lakukan Enkripsi Pada Kode OTP di Server Dengan DES
11. Kirim Kode OTP ke Android Sebagai Response Login
12. Sisi Client (Android) melakukan Dekripsi Kode OTP DES
13. Simpan Kode OTP dalam Intent
14. Passing Kode OTP ke Halaman Verifikasi OTP
15. Tampil Halaman Verifikasi OTP
16. Else
17. Tampil Pesan Error
18. Kembali ke Baris 1
19 End IF
20.Else
21 Kembali ke Baris 1
22.End IF

Flowchart dan algoritma diatas menggambarkan preses *login* dengan inputan *username* dan *password*. Jika pengguna berhasil teotentikasi, maka akan arahkan ke halaman verifikasi OTP, sebaliknya jika gagal, maka akan muncul pesan *error* yang memberitahukan bahwa kombanasi *username* dan *password* salah. Pengguna diharuskan untuk memasukan *username* dan *password* jika ingin mengunlangi proses *login*. di bawah ini *flowchart* halaman verifikasi OTP.



Gambar 3. Flowchart Halama Verifikasi OTP

Algoritma proses dari *flowchart* di atas:

1. Tampil Form OTP
2. Waktu Berjalan 60 Detik
3. Simpan Kode OTP hasil Intent dalam Variabel
4. Input Kode OTP
5. IF Verifikasi then
6. IF Cek AND Waktu > 0 then
7. Tampil Halaman Utama EO
8. Else
9. Tampil Pesan Error
10. Kembali ke Baris 1
11. IF Kirim Ulang OTP then
12. Kirim Email dan Password ke Server
13. Server Generate OTP dengan SHA-256
14. Kembali Ke Baris 1
15. End IF
16. Else

17. Kembali ke Baris 1
18. End IF

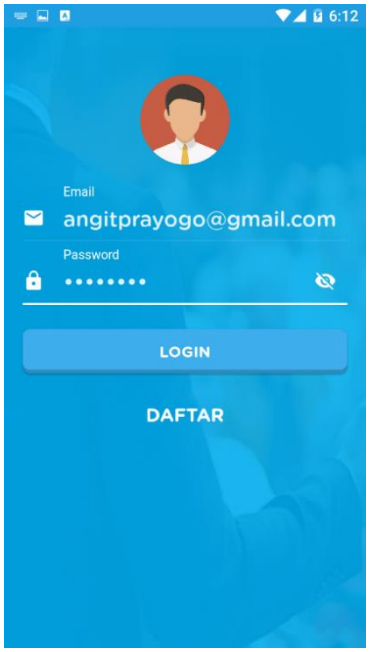
3. HASIL DAN PEMBAHASAN

3.1. Tampilan Layar

Disini akan dijelaskan mengenai tampilan-tampilan layar dari berbagai proses yang terjadi di aplikasi EO Blucampus, terutama proses *One Time Password* (OTP) diterapkan pada aplikasi ini. Berikut adalah beberapa tampilan dari aplikasi ini :

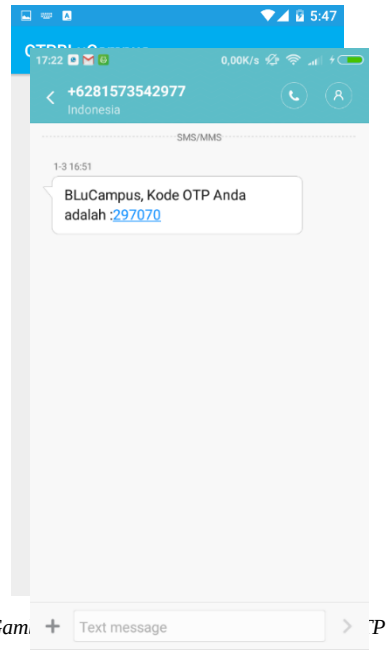
a. Tampilan Halaman Login

Pada halaman ini terdapat 2 buah inputan dan *button*. Inputan tersebut adalah *email* dan *password*. Sedangkan *button* nya adalah *login* dan daftar. Tampilan pada Halaman *Login* dapat dilihat pada gambar dibawah:



Gambar 4. Tampilan Halaman Login

b. Tampilan Halaman Verifikasi OTP



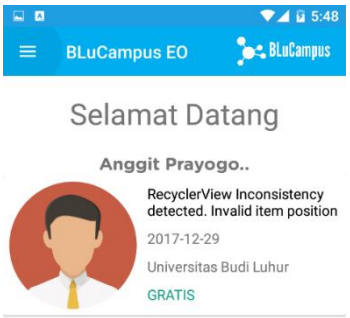
Gam *P*

c. Tampilan SMS Kode OTP

d. Tampilan Email Kode OTP

Gambar 7. Tampilan Email Kode OTP

e. Tampilan Halaman Utama EO



Gambar 8. Tampilan Halaman Utama EO

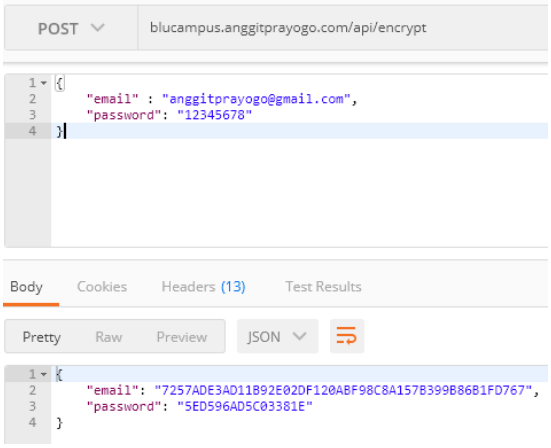
3.2. Uji Coba Program

Pada bagian ini, akan dilakukan pembuktian atau pengujian pembangkitan kode OTP, verifikasi OTP dan enkripsi serta dekripsi pada kode OTP yang akan dikirimkan ke *client* ataupun diterima dari *server*. Pengujian dilakukan untuk memenuhi dua tujuan, yaitu: apakah kode otp yang dibangkitkan unik dan apakah kode otp yang dikenakan algoritma enkripsi dapat didekripsi dengan baik.

3.3. Uji Coba Proses

a. Uji Coba Proses Enkripsi

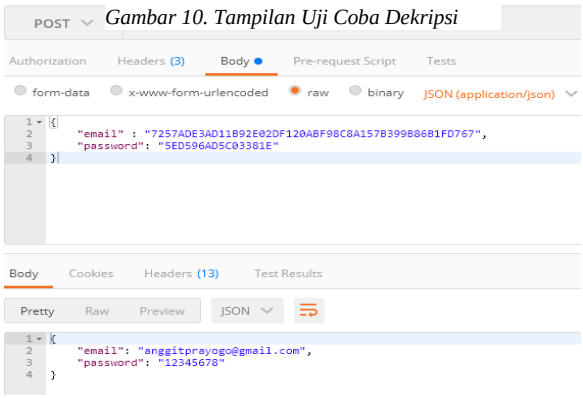
Tampilan ini berisi data user yang dikirimkan oleh user ketika login melalui aplikasi EO Blucampus dengan menggunakan tool postman.



Gambar 9. Tampilan Uii Coba Enkripsi

b. Uji Coba Proses Dekripsi

Tampilan ini berisi data hasil enrkipsi *user* yang dilakukan oleh *server* melalui aplikasi EO Blucampus dengan menggunakan *tool postman*.



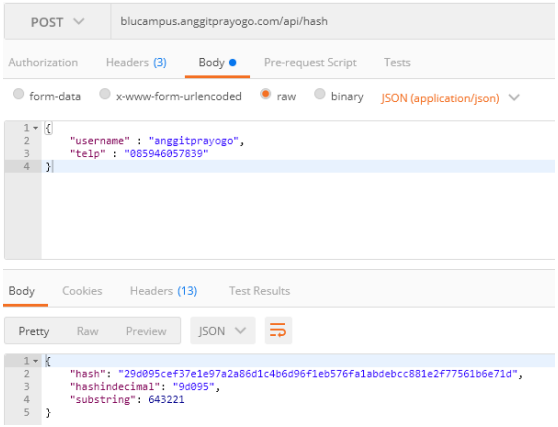
Gambar 10. Tampilan Uji Coba Dekripsi

Pada gambar di atas, dapat dibuktikan bahwa proses dekripsi terhadap data *cipher* Program Studi bisa dikembalikan seperti plainteks asli semula, yaitu "Teknik Informatika".

Berdasarkan percobaan serta analisis proses enkripsi dan dekripsi diatas, terbukti bahwa tidak ada perubahan data yang diolah dengan metode kriptigrafi yang digunakan. Dengan demikian bisa dikatakan bahwa tidak ada *error* di dalam proses enkripsi dan dekirpsi. Begitu juga dengan data dari *web service*, terbukti bahwa tidak ada perbedaan dari algoritma yang digunakan dan tidak ada *error* pada proses enkripsi dan dekripsi.

c. Uji Coba *Hash* SHA-256

Tampilan ini berisi hasil uji coba *hash* algoritma SHA-256 dengan menggunakan data diri *user*.



Gambar 11. Tampilan Uji Coba Hash SHA-256

3.4. Hasil Uji Coba Proses Pembangkitan Kode OTP

Untuk menguji keberhasilan model yang diusulkan dalam penelitian ini, penulis melakukan sampling pengguna dan waktu tertentu untuk melakukan proses login terhadap sistem. Penulis melakukan uji coba setiap pengguna masing-masing 5 kali percobaan login pada 5 pengguna yang berbeda.

Tabel 5. Tabel Hasil Uji Coba Pembangkitan Kode OTP

Email EO	Tanggal	Unix time	Kode OTP
anggitprayogo@gmail.com	2018-01-03	1515020464	423193
anggitprayogo@gmail.com	2018-01-03	1515020485	688164
anggitprayogo@gmail.com	2018-01-03	1515020488	625519
anggitprayogo@gmail.com	2018-01-03	1515020466	622755
anggitprayogo@gmail.com	2018-01-03	1515020516	832760
sokhibunwahid@gmail.com	2018-01-03	1515020460	547357
sokhibunwahid@gmail.com	2018-01-03	1515052918	130849
anggitprayogo@gmail.com	2018-01-03	1515020464	423193
sokhibunwahid@gmail.com	2018-01-03	1515020485	688164
sokhibunwahid@gmail.com	2018-01-03	1515020488	153273
sokhibunwahid@gmail.com	2018-01-03	1515020466	979734

Umarbasr1@gmail.com	2018-01-03	1515052876	215484
Umarbasr1@gmail.com	2018-01-03	1515052861	312553
Umarbasr1@gmail.com	2018-01-03	1515052903	853928
Umarbasr1@gmail.com	2018-01-03	1515052881	503413
Umarbasr1@gmail.com	2018-01-04	1515052873	930550
yantobasri@gmail.com	2018-01-04	1515052881	274920
yantobasri@gmail.com	2018-01-04	1515052907	899090
yantobasri@gmail.com	2018-01-04	1515052875	680489
yantobasri@gmail.com	2018-01-04	1515052889	348255

3.5. Evaluasi Program

Kelebihan Program:

a. Dengan adanya sistem pengamanan tambahan akan membuat aplikasi android EO Blucampus lebih aman, karena kode OTP yang dipakai selalu berubah.
b. Kode OTP yang telah digunakan tidak dapat digunakan kembali atau sekali pakai.Proses enkripsi dan dekripsi tidak memakan waktu yang lama.
c. Menggunakan algoritma SHA-256 yang bersifat *hash* (satu arah).
d. Pengguna lain tidak dapat mengakses halaman utama EO jika tidak mempunyai kode verifikasi OTP yang dikirim melalui SMS dan Email

Kekurangan Program:

a. Hanya menggunakan satu algoritma untuk membangkitkan OTP.Masih ada responden yang mengalami *error*, jadi penulis harus mencari tahu kekurangan yang masih ada dalam aplikasi ini.
b. Batas waktu EO untuk melakukan verifikasi kode OTP adalah 60 detik.

4. KESIMPULAN

Pada saat proses pembuatan dan pengujian dalam Tugas Akhir ini, berdasarkan perancangan, pembuatan, serangkaian uji coba dan analisis program dapat ditarik kesimpulan, yaitu:

a. Algoritma SHA-256 dapat digunakan untuk mengimplementasikan *One Time Password* untuk mengamankan sistem *Login*.
b. Penerapan *One Time Password* menggunakan algoritma SHA-256 dan DES dapat mengamankan proses login Android berbasis web service.
c. Algoritma DES dapat digunakan untuk mengamankan kode OTP yang telah dibangkitkan untuk dikirim ke client.
d. Hasil dari *One Time Password*, kode verifikasi hanya dikirim ke nomor handphone dan email pengguna.
e. Pengguna bisa menggunakan *password* yang sama, jika *password* diketahui oleh orang lain masih terlindungi oleh kode verifikasi.

5. DAFTAR PUSTAKA

[1] Primartha, R. (2011). Penerapan Enkripsi dan Dekripsi File menggunakan Algoritma Data Encryption Standard (DES). *Journal of Research in Computer Science and Applications*, *2*(1), hal. 13–18. https://doi.org/2301-8488

[2] Rahman, M. A., Kuswardayan, I., & Hariadi, R. (2013). Perancangan dan Implementasi RESTful Web Service untuk Game Sosial Food Merchant Saga pada Perangkat Android. *Jurnal Teknik POMITS*, *2*(1), hal. 2–5.

[3] Rohman, A. (2017). Implementasi Kriptografi dan Steganografi Dengan Metode Algoritma Des dan Metode End Of File Ajar Rohmanu, *1*(2), hal. 1–11.

[4] Santoso, K. I. (2013). Dua Faktor Pengamanan Login Web Menggunakan Otentikasi One Time Password Dengan Hash SHA. *Seminar Nasional Teknologo Informasi & Komunikasi Terapan 2013*, *2013*(November), hal. 204–210.

[5] Azham, M. N. (2016). Otentikasi Sistem Dengan Menggunakan *One Time Password* Memanfaatkan *Smartphone* Android. *Jurnal Link*, *2*(1), hal .2-7