PENINGKATAN KEAMANAN *LOGIN* WEBSITE DENGAN IMPLEMENTASI ONE TIME *PASSWORD* MENGGUNAKAN ALGORITMA SHA1 DAN MD5 BERBASIS MOBILE

Muhammad Sonny Ramadhan[1)]**, Pipin Farida Ariyani**[2)]
Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur
Jl. Ciledug Raya, Petukangan Utara, Jakarta Selatan 12260
Tlp. 021-5853753, Fax. 021-5866369
Email : ramadhansonny@gmail.com[1)], pipin.faridaariyani@budiluhur.ac.id[2)]

Abstrak

CV. Bintang Kharisma adalah perusahaan penyedia alat berat yang biasa bekerjasama dengan perusahaan-perusahaan yang bergerak di bidang pembangunan infrastruktur. Perusahaan ini menyimpan data penting seluruh pelanggan dan perusahaan yang tidak boleh diketahui oleh sembarang pihak. Sistem informasi berbasis online dapat digunakan untuk menunjang kegiatan di dalam suatu perusahaan. Sistem menjadi sangat rentan apabila dapat di akses oleh orang yang tidak berkepentingan sehingga dibutuhkan sistem login. Pada umumnya seseorang akan menggunakan password yang mudah diingat dan ditebak oleh orang lain atau menggunakan password yang tidak pernah berubah. Disisi lain password dapat dengan mudah diketahui oleh pihak lain dengan cara sniffing atau penyadapan. Hal tersebut membuat password tidak aman digunakan untuk mengakses sistem informasi. Maka dari itu, agar dapat mencegah terjadinya penyusupan kedalam sistem informasi dibuat sistem otentikasi berupa kode unik sekali pakai yang disebut One Time Password. Karena penggunaan smartphone android yang saat ini semakin marak digunakan.maka penelitian ini melakukan pemanfaatan smartphone android sebagai pengganti token untuk mengimplementasikan one time password. Untuk membangkitkan one time password digunakan algoritma Message Digest 5 (MD5) dan Secure Hash Algorithm 1 (SHA1) dan sedikit penambahan 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 mengakses login pada sistem informasi karena kode yang dihasilkan dari one time password ini selalu berubah dan tidak dapat digunakan kembali untuk mengakses sistem sehingga sistem tidak dapat dengan mudah dibobol oleh orang lain.

Kata kunci : *One Time Password,* Android, MD5, SHA1

1. PENDAHULUAN

Teknologi saat ini telah berkembang dengan sangat pesat, tentunya tingkat keamanan yang tinggi juga semakin diperlukan bersamaan dengan majunya teknologi. Sistem *login* merupakan suatu hal yang pasti ditemukan dan merupakan salah satu aspek penting di dalam dunia *internet,* yang perlu diperhatikan keamanannya. *Internet* dibangun melalui jaringan komputer yang saling terhubung, dengan begitu banyaknya *user* yang terhubung dalam jaringan, suatu data maupun informasi menjadi sangat rentan terhadap serangan dari pihak-pihak luar yang tidak berwenang.

Hal ini membuat seluruh data dan informasi dapat dengan mudah diakses oleh siapapun. Tidak hanya bisa diakses oleh orang yang berwenang atau berkepentingan, tetapi juga oleh orang lain yang ingin menggunakannya untuk kepentingan pribadi dengan cara mencuri data dan informasi dari sistem komputer tersebut (peretas). Untuk itu sistem dituntut agar dapat mengetahui pengguna atau *user*yang akan menggunakan sistem adalah pihak yang telah diberi izin atau yang berkepentingan. Pengguna haruslah mengidentifikasi dirinya ke sistem, dan sistem harus memastikan apakah identifikasi tersebut otentik atau tidak.

CV. Bintang Kharisma adalah perusahaan jasa penyedia alat berat yang biasa bekerjasama dengan perusahaan-perusahaan yang bergerak di bidang pembangunan infrastruktur. Perusahaan ini juga menyimpan data penting seluruh pelanggan dan perusahaan yang tidak boleh diketahui oleh sembarang orang supaya tidak terjadi hal-hal merugikan yang tidak diinginkan.

Selama ini *website* CV. Bintang Kharisma masih dalam tahap pengembangan yang bertujuan untuk mencegah terjadinya pencurian data yang dilakukan oleh peretas. Karena jika pencurian data terjadi dapat menyebabkan penyalahgunaan data untuk hal yang tidak diinginkan dan dapat merugikan para pelanggan dan juga perusahaan.

Sistem otentikasi *One Time Password* adalah salah satu dari berbagai macam cara untuk mengatasi serangan peretas pada sistem *login*. Algoritma yang akan digunakan pada penelitian akhir ini adalah MD5 (Message Diggest 5)dan SHA1 (Secure Hash Algorithm 1). Dipilihnya algoritma MD5 dikarenakan pemrosesan data yang cukup cepat karena hanya mengambil panjang data sebesar 128 bit. Dan dipilihnya algoritma SHA1 karena kebal terhadap *collision* (tabrakan paket data)[1].

Pada latar belakang yang telah dikemukakan, dapat diperoleh beberapa permasalahan yaitu, bagaimana meningkatkan keamanan proses *login* sistem agar pada saat *username* dan *password* berhasil diketahui oleh orang lain, *username* dan *password*, serta informasi penting pada *website* masih tetap aman. Dan bagaimana cara merancang dan menciptakan aplikasi otentikasi keamanan sistem *login* berbasis *mobile* yang mudah digunakan sekaligus dimengerti oleh pengguna.

Meningkatkan keamanan sistem *login website* menggunakan *One Time Password* dan merancang aplikasi berbasis *Android* dengan tampilan yang sederhana sehingga mudah digunakan oleh pengguna adalah tujuan dari penelitian ini. Kinerja dari One Time *Password* diimplementasikan terhadap smartphone android dengan cara meng-hashing-kan *username, password*, dan waktu menggunakan algoritma *Message Digest 5* dan *Secure Hash Algorithm 1*.

Adapun batasan masalah pada penelitian ini adalah algoritma yang digunakan untuk penelitian ini yaitu*MD5* dan *SHA1*. *Dansmartphone android* digunakan sebagai pengganti *token* dalam mengimplementasikan *one time password. Aplikasi* dikembangkan pada sistem operasi *android* versi *4.2 jelly bean* atau lebih tinggi.

2. METODE PENELITIAN

Dalam penyusunan penelitian ini penulis melakukan berbagai macam metode dengan cara mengumpulkan data atau informasi yang dibutuhkan. Metode yang dibutuhkan antara lain:

a. Studi pustaka

Yaitu mencari, memahami, dan mempelajari dengan seksama dari buku atau berbagai macam artikel yang berkaitan dengan *One Time Password*, SHA1, dan MD5. Serta menggali sumber-sumber pustaka lainnya yang juga berhubungan dengan topik ini, yang bisa juga didapat dari berbagai makalah, forum diskusi, textbook, pendapat ahli, jurnal, dan sebagainnya, baik media cetak maupun elektronik.

b. Wawancara

Yaitu upaya yang dilakukan untuk mendapatkan dan melengkapi data-data yang diperoleh dapat melalui wawancara meliputi pihak-pihak terkait di dalam aplikasi *One Time Password* yang masih dalam tahap pembuatan.

c. Prototyping

Yaitu teknik analisis data dalam pembuatan perangkat lunak atau *software*.Dan teknik*Prototyping* merupakan proses pembuatan *dummy* atau model sederhana dari *software* yang akan dibuat, sehingga memungkinkan pengguna memiliki gambaran dasar tentang program untuk bahan evaluasi serta pengujian awal.Berikut tahapan dari model Prototyping :

1) Menentukan Kebutuhan

Bertemu dengan user yang akan membutuhkan aplikasi dengan mengidentifikasi permasalahan,lalu menentuhkan kebutuhan dan mencari konsep yang sesuai dengan yang dibutuhkan.

2) Merancang *Prototype*

Setelah garis besar konsep sudah disepakati, yang selanjutnya dilakukan membuat suatu *prototype* dengan cara membuat rancangan aplikasi yang bersifat sementara yang difokuskan pada garis besar konsep yang telah disepakati antara pengguna dan pengembang.

3) Evaluasi *Prototype*

User akan melakukan evaluasi dari *prototype* aplikasi, apakah aplikasi yang dibuat sudah sesuai dengan keinginan pengguna. Jika *prototype* sudah selesai maka langkah berikutnya akan diambil, namun jika tidak sesuai maka *prototype* akan diperbaiki dengan mengulang langkah pertama, kedua, dan ketiga.

4) Mengkodekan Aplikasi

Pada tahap ini, Prototyping aplikasi yang sudah disepakati dierjemahkan ke dalam bahasa pemrograman yang sesuai.

5) Menguji Aplikasi

Setelah aplikasi sudah menjadi suatu perangkat lunak yang siap pakai, aplikasi akan dilakukan pengujian terlebih dahulu untuk mengetahui ada atau tidaknya celah (bugs) pada fungsi utama aplikasi yang sedang dibuat.

6) Evaluasi Aplikasi

Pengguna akan melakukan evaluasi terhadap program atau aplikasi yang sedang dikembangkan, apakah aplikasi sudah sesuai dengan yang di harapkan. Jika telah sesuai maka langkah berikutnya akan dilakukan, namun jika dirasatidak sesuai yang diharapkan, aplikasi akan dirancang ulang kembali dengan mengulang langkah keempat dan kelima.

7) Menggunakan *Prototype*

Dilakukan pengembangan aplikasi yang sesuai dengan hasil evaluasi serta dilakukan penambahan atau perbaikan yang diminta oleh user ditahap sebelumnya. Jika telah sesuai dengan permintaan user, maka aplikasi dapat dikembangan menjadi aplikasi nyata.

d. Pengujian

Pada tahap ini dilakukan pengujian dari aplikasi yang telah dibuat, serta mengevaluasi apabila masih terdapat kesalahan dan kekurangan.

2.1. Serangan Keamanan Komputer

Cara seseorang untuk mendapatkan pesan dalam saluran komunikasi [2], penyerangan dapat dikategorikan menjadi:

a. Sniffing

Secara harfiah sniffing memiliki arti mencium, tentunya dalam hal ini yang dicium adalah data pesan (baik yang belum maupun yang sudah dienkripsi) dalam suatu saluran komunikasi. Hal tersebut umum terjadi pada saluran publik yang rentan keamanannya. Pelaku dapat merekam pembicaraan yang terjadi antara dua *user* atau lebih.

b. Replay Attack

Hacker bisa merekam pesan-pesan handshake (persiapan komunikasi), ia sangat mungkin dapat mengulang pesan-pesan yang telah direkamnya untuk menipu salah satu pihak yang ada pada jalur komunikasi.

c. Spoofing

Penyerang, misalnya A, bisa menyamar menjadi B. Semua orang dibuat percaya bahwa A adalah B. Penyerang berusaha meyakinkan pihak-pihak lain bahwa tak ada yang salah dengan komunikasi yang sedang di lakukan, padahal komunikasi tersebut dilakukan dengan sang penipu/penyerang. PIN ke dalam *Card Acceptance Device* (CAD) yang benar-benar di buat seperti CAD asli tentu saja sang penipu bisa mendapatkan PIN pemilik smartcard. Pemilik smartcard tidak tahu bahwa telah terjadi kejahatan didalamnya.

d. Man-in-the-middle

Jika terkadang spoofing hanya menipu satu pihak, maka dalam scenario ini saat A hendak berkomunikasi dengan C, B dimata A seolah-olah adalah C, dan B dapat pula menipu C sehingga B seolah-olah adalah A. B dapat berkuasa dengan penuh atas jalur komunikasi keduanya dan bisa membuat berita fitnah atau hoax.

2.2 Metode Pengembangan Sistem

a. One Time *Password*

One Time *Password* (OTP) adalah *password* yang hanya berlaku untuk satu kali sesi *login* atau transaksi. OTP menghindari sejumlah kelemahan yang berkaitan dengan tradisional (statis) *password*. Kelemahan paling penting yang ditujukan oleh OTP berbeda dengan *password* statis, OTP tidak rentan terhadap serangan *replay* (*replay attack*). Ini berarti jika penyusup memilliki potensi berhasil merekam OTP yang sudah digunakan untuk masuk ke dalam sistem atau layanan atau untuk melakukan transaksi, penyusup tersebut tidak akan dapat menyalahgunakannya disesi berikutnya karena kode OTP tidak berlaku lagi [3].

b. Message Digest 5

Message Digest Algorithm 5 atau MD5adalah fungsi hash kriptografi. Pada umumnya metode algoritma ini digunakan untuk melakukan pemeriksaan integritas file dalam berbagai macamsituasi[4]. Dalam Penelitian ini Algoritma MD5 digunakan sebagai salah satu proses *hashing* untuk pembangkit kode *OTP*.

c. Secure Hash Algorithm 1

Secure Hash Algorithm 1 atau SHA1 menerima masukan berupa *string* dengan ukuran maksimum 264 bit. Untuk setiap *string*, SHA1 akan menghasilkan keluaran sebanyak 160bit dari *string* tersebutdan *string* keluaran itu disebut *message digest*. Panjang jarak *message digest* dapat berkisar antara 160 sampai 512 bit tergantung algoritmanya. Berdasarkan cirinya SHA1 dapat dipadukan dengan algoritma kriptografi lainnya, seperti *Digital Signature Algorithms* atau dalam generasi angka yang acak (*bits*).SHA1 dikatakan cukup aman karena proses SHA1 dihitung secara infisibel untuk mencari *string* yang sesuai untuk menghasilkan *message digest* atau dapat juga digunakan untuk mencari dua*string* berbeda yang akan menghasilkan *message digest*yang sama. Untuk SHA1 ukuran blok*string* -m bit- dapat ditentukan tergantung dari algoritmanya. Pada SHA-1 masing-masing blok *string* mempunyai 512 bit dimana dapat dilakukan dengan 16 urutan sebesar 32 bit. SHA1 digunakan untuk menghitung *message digest* pada*string* atau *file* data yang diberikan sebagai *input*. Tujuan pengisian *string* adalah untuk menghasilkan total dari *string* yang diisi menjadi perkalian dari 512 bit[5].

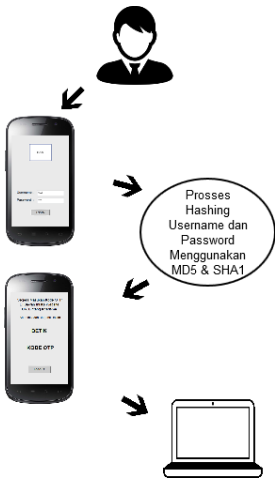
3. RANCANGAN SISTEM DAN APLIKASI

Pada penelitian ini akan dibuat dua aplikasi yaitu aplikasi *login* web dan aplikasi untuk membangkitkan One Time *Password* pada smartphone berbasis Android. Proses dari aplikasi yang akan dibuat dimulai dari pengisian *username* dan *password* yang telah terdatakepada aplikasi Android. Pada aplikasi Android akan dihasilkan

generate code yang akan digunakan untuk mengisi form sebagai kode One Time *Password* pada website. User dapat masuk (*login*) pada website dengan memasukkan *username*, *password*, dan kode generate yang sudah didapatkan dari Android lalu memilih *button login*.

3.1. Alur Proses*One Time Password*

Pada alur proses *login website* dilakukan pengecekan *username* dan *password*, apakah sudah seperti data yang terdaftar pada database atau belum. Jika sudah sesuai maka kode OTP dengan estimasi waktu 60 detik akan ditampilkan. Jika lebih dari 60 detik maka mobile akan melakukan *generate* kode *OTP* baru. Apabila semua proses tersebut sudah berjalan sistem akan melakukan validasi proses *login* berhasil atau gagal, dan menampilkan pesan dalam bentuk *message box* atau *popup*.



Gambar 1: Alur Proses One Time Password

3.2. Rancangan Layar Menu *LoginMobile*

Pada tahap pertama pengguna memasukkan *username* dan *password* pada kolom yang tersedia dengan catatan *username* dan *password* sudah terdaftar pada *database* untuk mendapatkan kode *OTP* yang akan ditampilkan pada *mobile android*.

Gambar 2: Rancangan Layar Menu LoginMobile

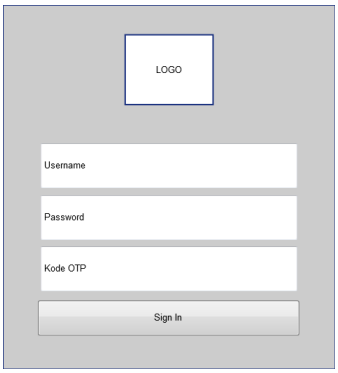
3.3. Rancangan Layar *Mobile One Time Password* Dibangkitkan

Kode *OTP* akan tampil setelah proses *login* sebelumnya dinyatakan berhasil. Kode OTP akan tampil dan berganti setiap 60 detik sekali. Kemudian kode dapat diinputkan pada *Text Field* Kode OTP yang tersedia pada halaman *loginwebsite*.

Gambar 3: Rancangan Layar MobileOne Time Password Dibangkitkan

3.4. Rancangan Layar Form *Login Website*

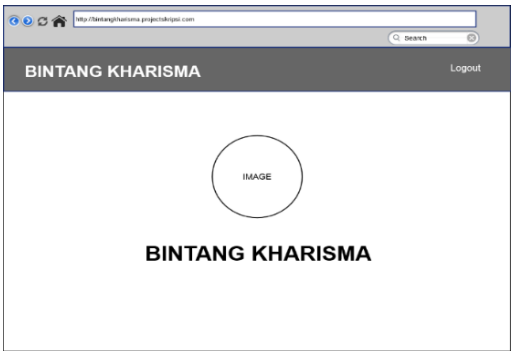
Berikut merupakan menu yang akan tampil pada website dan yang akan digunakan untuk memasukkan *Username*, *Password* dan kode OTP yang telah didapatkan dari mobile token pada proses sebelumnya. Setelah itu sistem akan melakukan pengecekan terhadap database dan kode OTP antara *client* dan *server*.Proses pengecekan kode OTP sangat berpengaruh terhadap koneksi dan waktu anata *client* dan *server*.



Gambar 4: Rancangan Layar Form Login Website

3.5. Rancangan Layar Halaman Utama *Website*

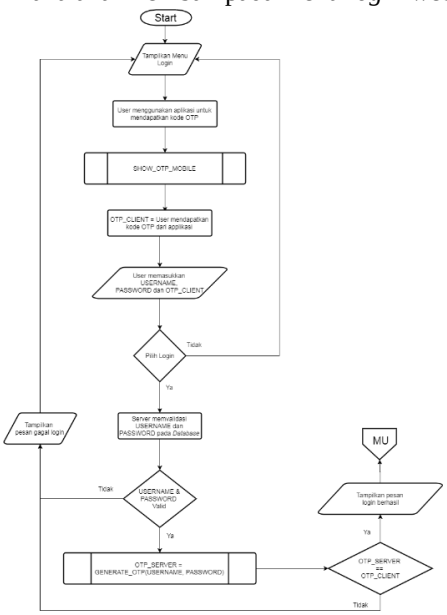
Berikut merupakan tampilan Menu Utama *Website* setelah proses *login* atau proses validasi antara *client* dan *server* yang sebelumnya berhasil dilakukan. Halaman ini tidak akan tampil jika proses *login* OTP tidak berhasil, dengan kata lain kode user ID atau kode OTP tidak diterima oleh sistem.



Gambar 5: Rancangan Layar Halaman Utama Website

3.6. *Flowchart Login Website*

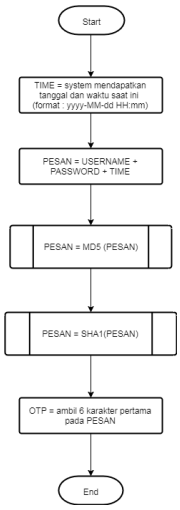
Pada Flowchart ini menggambarkan proses *Login* pada web. Untuk masuk pada halaman utama web, user harus memasukkan *username*, *password* dan kode OTP yang didapatkan dari mobile token terlebih dahulu. Setelah itu sistem akan melakukan pengecekan *username*, *password*, dan kode otp apakah sudah sesuai. Jika sesuai maka akan menampilkan menu utama web, tetapi jika tidak sesuai maka akan kembali pada menu *login* web.



Gambar 6: Flowchart Login Website

3.7. *Flowchart One Time Password*

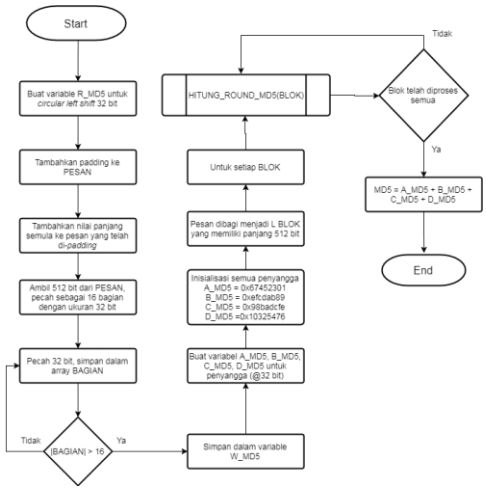
Pada flowchart berikut ini menjelaskan nilai OTP menggunakan satu variabel. Variable tersebut mengambil satuan waktu *login* yang diambil adalah jam, menit, tanggal, bulan, dan tahun. Kemudian variabel tersebut digabungkan dengan *username* dan *Password* pengguna user. Variabel ini akan di hashing dengan menggunakan algoritma Secure Hash *Algorithm 1 (SHA1)* dan *Message Diggest 5 (MD5)*. Ketika sudah mendapatkan 40 bit hasil hasing, kode akan disubstring 6 digit pertama dan disimpan sebagai variabel kode OTP. Setelah didapatkan kode OTP akan ditampilkan di menu mobile OTP pada android. Proses ini akan berlangsung setiap 60 detik sekali untuk mendapatkan kode OTP yang terbaru.



Gambar 7: Flowchart One Time Password

3.8. *Flowchart MD5*

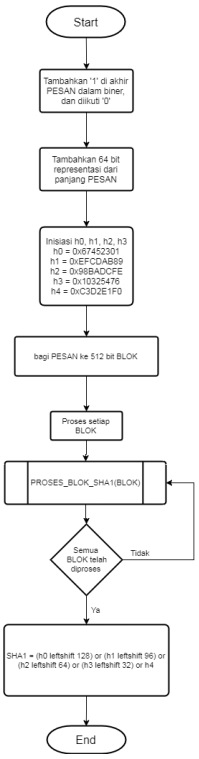
Pada flowchart berikut ini akan dijelaskan bagaimana proses hashing variabel yang menampung waktu *login*, *username* dan *password* menggunakan algoritma *Message Digest 5 (MD5)*.



Gambar 8: Flowchart MD5

3.9. *Flowchart SHA1*

Pada flowchart dibawah ini akan dijelaskan bagaimana proses hashing variabel yang menampung waktu *login*, *username* dan *password* menggunakan algoritma Secure Hash Algorithm (SHA1).



Gambar 9: Flowchart SHA1

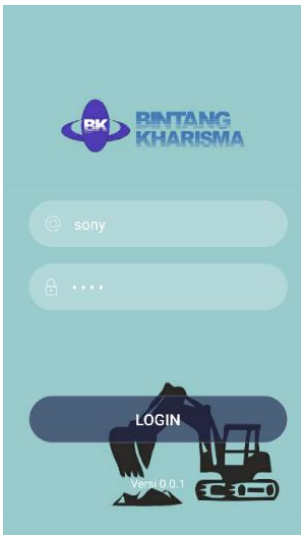
4. HASIL DAN PEMBAHASAN

4.1. Tampilan Layar Aplikasi

Tampilan layar program berguna untuk mengetahui apakah aplikasi yang telah dibuat sudah sesuai dan telah berjalan secara maksimal atau bahkan terjadi beberapa kesalahan yang tidak diinginkan. Dan tampilan layar juga berfungsi sebagai bahan untuk evaluasi peneliti terhadap program yang telah dibuat. Dengan dibuatnya tampilan layar juga memudahkan peneliti dan pembaca menganalisa keunggulan dan kekurangan program. Berikut ini adalah tampilan layar aplikasi baik pada smartphone android maupun tampilan layar pada website beserta penjelasan penggunaannya di masing-masing tampilan yang ada pada aplikasi One Time *Password* ini.

a. Tampilan Layar Menu *Login* pada *Mobile*

Tampilan aplikasi pada *smartphone android* terdiri dari beberapa bagian, salah satunya tampilan awal ketika user meng-input kode *username* dan *password*. Ketika user telah mengisi *textfield username*, *password* dan menekan tombol *Login*. Sistem akan melakukan pengecekan apakah *username* dan *password* yang diinputkan sesuai dengan data yang tersimpan pada *database*. Berikut adalah tampilan ketika *user* melakukan penginputan *username* dan *password*.



Gambar 10: Tampilan Layar Menu Login pada Mobile

b. Tampilan Layar Mobile*One Time Password* Dibangkitkan

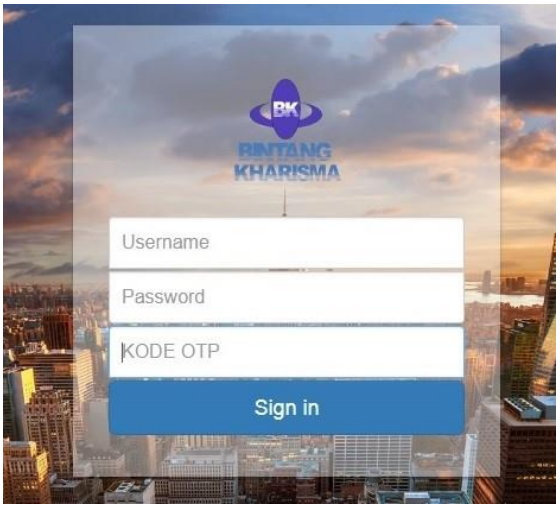
Pada gambar berikut ini adalah tampilan layar dimana ketika mobile token OTP dibangkitkan. Pada tampilan layar berikut ini juga terdapat waktu hitung mundur. Waktu mundur berfungsi mengingatkan user bahwa ketika waktu sudah habis akan muncul kode OTP baru. Pada tampilan layar ini juga terdapat kode OTP yang dapat digunakan untuk *login* pada website.

Gambar 11: Tampilan Layar Mobile One Time Password Dibangkitkan

c. Tampilan Layar *Login Website*

Pada tampilan layar menu *login* web berikut ini menjelaskan terdapat 3 kolom yang tersedia. Kolom-kolom tersebut antara lain kolom *username*, *password*, dan kode OTP. Pada kolom *username* dan *password*, user memasukkan *username* dan *password* yang user miliki yang sesuai dengan databse. Pada

kolom kode OTP, user memasukkan kode OTP yang didapat dari Mobile Token Android.



Gambar 13 : Tampilan Layar Login Website

d. Tampilan Layar *Website* Setelah Proses *Login*

Berikut ini adalah tampilan halaman *website* perusahaan yang akan muncul setelah melakukan proses *login* pada halaman utama *website*. Pada halaman ini terdapat beberapa *button link* pada bagian *header website* seperti portofolio, *about*, *contact* dan *logout*.

Gambar 14 : Tampilan Layar Website Setelah Proses Login

4.2. Evaluasi Program

Berdasarkan hasil pengujian yang telah dilakukan pada penelitian, terdapat beberapa kelebihan dan kekurangan dari aplikasi ini. Adapun kelebihan dari aplikasi ini adalah dengan adanya pengamanan tambahan pada sistem *login* akan membuat *website* menjadi lebih aman, dikarenakan kode *OTP* yang digunakan selalu berubah-ubah dalam jangka waktu 60 detik sekali. Kode *OTP* yang digunakan hanya sekali pakai atau tidak dapat digunakan kembali pada sesi berikutnya. Pembangkit kode *OTP* menggunakan *smartphone android* sebagai token yang hampir dimiliki oleh semua pengguna. Menggunakan Algoritma *MD5* dengan penambahan Algoritma *SHA1* yang bersifat *Hash* atau sekali pakai.

Sedangkan kekurangan yang terdapat pada aplikasi ini yaitu aplikasi hanya dapat digunakan atau diimplementasikan pada *smartphone* dengan sistem operasi *Android*, dan juga keberhasilan pembangkitan kode OTP dipengaruhi oleh koneksi dan perbedaan waktu antara client dan server.

4.3. Analisa Hasil

Proses *login* pada website melakukan pengecekan antara *client* dan *server* meliputi *username*, *password* dan kode OTP dan menampilkan validasi dalam bentuk *message box* atau *popup*. Dari proses yang telah dilakukan dapat disimpulkan bahwa kode OTP menjadi salah satu bagian terpenting untuk dapat melakukan *login* ke halaman utama website. Hal ini karena tersedianya kolom kode OTP pada halaman *login* website yang harus diisi dengan kode OTP yang didapat dari aplikasi Mobile Token.

Pengujian juga dilakukan dengan meng-install-kan perangkat lunak untuk sniffing yaitu keylogger. Dengan asumsi bahwa *database* telah diretas, peretas berusaha melakukan *login* kembali menggunakan *username* dan *password* yang telah didapatkan, kode OTP sudah tidak berlaku lagi karena kode OTP berubah secara otomatis setiap 60 detik. Dengan adanya masa aktif yang dimiliki kode OTP maka setelah kode OTP sudah melewati 60 detik atau kode OTP sudah pernah digunakan untuk mengakses sistem maka status dari kode OTP menjadi tidak valid. Jika kode OTP tidak valid maka user tidak dapat mengakses website.

Berikut ini dijelaskan hasil pengujian aplikasi OTP dengan menggunakan dua *username* yang berbeda secara berulang-ulang:

Tabel1 : Hasil Pengujian

Username	*Password*	Kode OTP	Waktu/Jam	Status
Sonny	1234	8D24F3	14:05:20	Berhasil
Sonny	1234	8D24F3	14:06:15	Gagal
Sonny	1234	518E85	14:06:40	Berhasil
Helmas	2234	3C98BF	14:30:10	Berhasil
Helmas	2234	D87D0E	14:31:15	Berhasil
Helmas	2234	4C3F5F	14:32:45	Berhasil

5. KESIMPULAN

5.1. Kesimpulan

Dari hasil pengujian dan implementasi yang sudah dilakukan, didapatkan beberapa kesimpulan yaitu, Algoritma *MD5* dan *SHA1* yang digunakan pada penerapan *One Time Password* dapat mengamankan sistem *login website* dengan cara memanfaatkan *Smartphone Android* sebagai pengganti token untuk mengimplementasikan *One Time Password*. Dan pengujian keamanan dari aplikasi yang telah dibuat dengan menggunakan cara

sniffing didapatkan hasil bahwa apabila *username* dan *password* berhasil diketahui atau disadap oleh orang lain, orang tersebut tidak dapat menggunakannya kembali untuk mengakses sistem *login*. Dengan demikian dengan adanya aplikasi *One Time Password* sistem *login website* terbukti menjadi lebih aman dari serangan peretas atau hacker.

5.2. Saran

Masih terdapat beberapa kekurangan pada penelitian ini, sehingga penelitian ini masih dapat dikembangkan dan diperbaharui. Saran penulis untuk penelitian atau untuk dikembangkan lebih lanjut, yang dapat dilakukan adalah menggunakan tambahan atau kombinasi algoritma lain untuk membangkitkan kode *OTP*, dan aplikasi ini diharapkan dapat dikembangkan untuk digunakan pada smartphone dengan sistem operasi lain seperti *iOS*, *Windows Phone*, dan berbagai sistem operasi lain yang mungkin saja akan dibuat pada era selanjutnya.

6. DAFTAR PUSTAKA

[1] Nugroho, E.P., 2016. SMS authentication code generated by Advance Encryption Standard (AES) 256 bits modification algorithm and One time *Password* (OTP) to activate new applicant account.International Conference on Science in Information Technology, 1(1), pp.40–44.

[2] Santoso, 2013. Dua Faktor Pengamanan *Login* Web Menggunakan Otentikasi One Time *Password* Dengan. In Semarang, pp. 204–210.

[3] Aryasa, K. & Paulus, Y.T., 2014. Implementasi Secure Hash Algorithm-1 Untuk Pengamanan Data Dalam Library Pada Pemrograman Java. , 1(1), pp.57–66.

[4] Mustofa, R.P., 2013. Aplikasi Mobile Android "One Time *Password*(OTP)" Untuk Meningkatkan Keamanan Otentikasi. , pp.1–15.

[5] Fresdian, Yama. 2015. MD5 (Message Digest Algorithm 5). Ilmu Komputer. Vol.1, No.1