IMPLEMENTASI *UNIFORM RESOURCE LOCATOR ENCRYPTION* PADA WEBSITE BERBASIS ALGORITMA BASE64 STUDI KASUS PADA PIMPINAN WILAYAH AISYIYAH JAWA TENGAH

Aldino Rahardian

Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang 50131

E-mail : rahardian72122gmail.com

ABSTRAK

Website merupakan salah satu objek yang informasinya rawan untuk diketahui oleh pihak yang tidak berwenang. Oleh karena itu diperlukan adanya pengamanan data. Penelitian ini bertujuan untuk mengamankan data karena keamanan data merupakan hal yang sangat penting dalam menjaga kerahasiaan informasi. Salah satu cara untuk mengamankan data adalah dengan sistem kriptografi yaitu dengan mengubah isi informasi (plaintext) tersebut menjadi isi informasi yang sulit dipahami melalui proses enkripsi (enchiper), dan untuk memperoleh kembali informasi asli dilakukan proses dekripsi (dechiper). Dalam hal ini, penulis menggunakan metode enkripsi Base64 yang di implementasikan pada URL (Uniform Resource Locator) website. Objek penelitian yang digunakan penulis adalah PWA Jateng. Setelah dilakukan pengujian pada penelitian ini menunjukan bahwa metode enkripsi Base64 yang diimplementasikan pada variabel URL cukup efektif untuk mencegah SQL Injection.

Kata kunci : Website, Kriptografi, Base64, Enkripsi

1. PENDAHULUAN

Masalah keamanan dan kerahasiaan merupakan salah satu aspek penting dari suatu sistem informasi. Dalam hal ini, sangat terkait dengan pentingnya informasi tersebut dikirim dan diterima oleh orang yang berkepentingan. Informasi akan tidak berguna lagi apabila di tengah jalan informasi itu disadap atau dibajak oleh orang yang tidak berwenang.

Website merupakan salah satu yang sangat rawan untuk disadap atau dibajak. Dari masa ke masa teknologi website mengalami perkembangan yang begitu pesat. Keamanan website sangat diperlukan bagi suatu organisasi ataupun perusahaan karena untuk menjaga integritas data dan informasi pada organisasi dan perusahaan tersebut. Website yang tidak menawarkan keamanan akan sangat berpotensi hilangnya integritas data. Namun, kebanyakan dari pemilik website mengabaikan *security* system pada website tersebut. Padahal banyak sekali *cracker-cracker* yang tidak bertanggung jawab yang dapat merusak dan mencari kelemahan sistem website tersebut. Ada berbagai macam serangan yang dilakukan para *hacker* dalam menyadap suatu website antara lain *SQL Injection*. *SQL Injection* adalah jenis aksi hacking pada keamanan komputer dimana seorang *hacker* bisa mendapatkan akses ke database didalam sistem. Kejadian ini dapat berisiko karena *hacker* dapat mengubah informasi ataupun data yang ada dalam website.

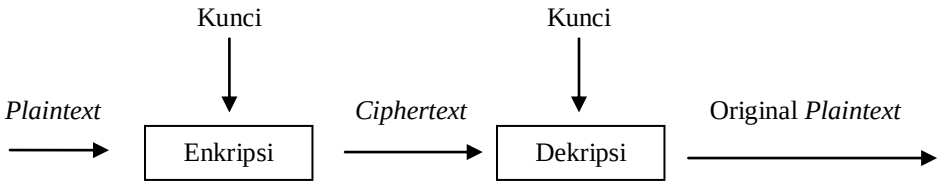
Salah satu cara untuk melakukan pengamanan website adalah dengan Ilmu Kriptografi. Kriptografi berasal dari bahasa Yunani yaitu *Criptography*. Criptography terdiri dari kata *kripto* yang artinya rahasia dan *graphia* yang artinya tulisan. Jadi dapat dikatakan Kriptografi adalah tulisan yang tersembunyi. Dalam ilmu Kriptografi mempelajari teknik enkripsi yang digunakan untuk mengamankan website. Teknik enkripsi berbasis Algoritma Base64 akan diimplementasikan pada variabel *URL* website Pimpinan Wilayah Aisyiyah Jawa Tengah.

2. TINJAUAN PUSTAKA

2.1 Kriptografi

Criptography berasal dari bahasa Yunani. Menurut bahasanya, istilah tersebut terdiri dari kata *kripto* dan *graphia*. Kripto berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). Menurut terminologinya, kriptografi adalah ilmu atau seni untuk menjaga keamanan pesan, ketika suatu pesan dikirim dari suatu tempat ke tempat lain, isi dari pesan tersebut kemungkinan dapat disadap oleh pihak lain. Untuk menjaga keamanan pesan, maka pesan tersebut dapat *discreamble*/diacak atau diubah menjadi kode yang tidak dapat dimengerti oleh orang lain. Tujuan dari sistem kriptografi adalah *Authentication, Integrity, Authority, Non-repudiation.*

Dasar matematis yang mendasari proses enkripsi dan dekripsi adalah relasi antara dua himpunan yaitu himpunan yang berisi elemen *plainteks* dan himpunan yang berisi elemen *chipteks*. Enkripsi dan dekripsi merupakan fungsi transformasi antara dua himpunan tersebut.



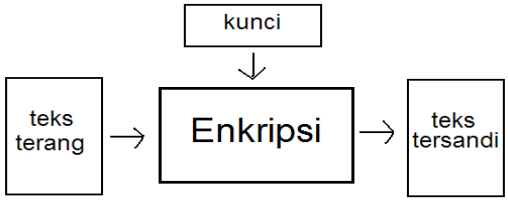
Gambar 2.1 Cryptosystem [1]

2.2 Enkripsi

Enkripsi adalah sebuah proses yang melakukan perubahan sebuah kode dari yang bisa dimengerti menjadi sebuah kode yang tidak bisa dimengerti (tidak terbaca). Enkripsi dapat diartikan sebagai kode atau chiper. Isu-isu yang terkait dengan keamanan dan kerahasiaan data adalah *privacy* (kerahasiaan), *integrity* (keutuhan), *authenticity* (keaslian), *non-repudiation* (pembuktian yang tak tersangkal). Di pertengahan tahun 1970-an, enkripsi kuat dimanfaatkan untuk pengamanan oleh sekretariat agen pemerintah Amerika Serikat pada domain publik, dan saat ini enkripsi telah digunakan pada sistem secara luas, seperti Internet e-commerce, jaringan Telepon bergerak dan ATM pada bank. Enkripsi dapat digunakan untuk tujuan keamanan. Ilmu yang mempelajari teknik enkripsi disebut kriptografi. Gambaran sederhana tentang enkripsi, misalnya mengganti huruf a dengan n, b dengan m dan seterusnya. Pembahasan enkripsi akan terfokus pada enkripsi password dan enkripsi komunikasi data.

Terdapat tiga kategori enkripsi yaitu :

1. Kunci enkripsi rahasia, dalam hal ini terdapat sebuah kunci yang digunakan untuk mengikrisi dan juga sekaligus mendeskripsikan informasi.
2. Kunci enkripsi *public*, dalam hal ini terdapat dua kunci yang digunakan, satu untuk proses enkripsi, satu lagi untuk proses deskripsi.
3. Fungsi *one-way*, dimana informasi dienkripsi untuk menciptakan "*signature*" dari informasi asli yang bisa digunakan untuk keperluan autentifikasi.



Gambar 2.2 Alur Enkripsi [7]

2.3 Algoritma Base64

Algoritma Base64 merupakan salah satu algoritma untuk Encoding dan Decoding suatu daya ke dalam format ASCII, yang didasarkan pada bilangan dasar 64 atau bisa dikatakan sebagai salah satu metode yang digunakan untuk melakukan encoding (penyadian) terhadap data binary. Umumnya digunakan pada berbagai aplikasi seperti e-mail via MME, data XML, atau untuk keperluan encoding URL. Prinsip encodingnya adalah dengan memilih kumpulan dari 64 karakter yang dapat diprint (printable), dengan demikian data dapat disimpan dan ditransfer melewati media yang didesain untuk menangani data tekstual, penggunaan lain encoding Base64 adalah untuk melakukan *obfuscation* atau pengacakan data. Skema enkripsi Base64 biasanya juga digunakan ketika diperlukan sandi terhadap data biner yang didesain untuk menangani data berbentuk teks, hal ini ditujukan untuk menjaga data selama pengiriman ke suatu server. Karakter yang dihasilkan pada transformasi Base64 ini terdiri dari A..Z, a..z dan 0..9, serta ditambahkan dengan dua karakter terakhir yang bersimbol + dan / serta satu buah karakter sama dengan (=) yang digunakan untuk penyesuaian dan menggenapkan data binary atau istilahnya disebut sebagai pengisi pas. Karakter simbol yang akan dihasilkan akan tergantung dari proses algoritma yang berjalan.

Kriptografi Base64 banyak digunakan di dunia internet sebagai media data format untuk mengirim data, ini dikarenakan hasil dari Base64 berupa *plaintext*, maka data ini akan jauh lebih mudah dikirim, dibandingkan dengan format data yang berupa binary. Dalam *Encoding_Base64* dapat dikelompokkan dan dibedakan menjadi kriteria yang tertera dan dapat dilihat di dalam table.

Tabel 2.1 Encoding Base64 (Josefsson,2003)

Value	Karakter Encoding64	Value	Karakter Encoding64	Value	Karakter Encoding64	Value	Karakter Encoding64
0	A	17	R	34	i	51	z
1	B	18	S	35	j	52	0
2	C	19	T	36	k	53	1
3	D	20	U	37	l	54	2
4	E	21	V	38	m	55	3
5	F	22	W	39	n	56	4
6	G	23	X	40	o	57	5
7	H	24	Y	41	p	58	6
8	I	25	Z	42	q	59	7
9	J	26	A	43	r	60	8
10	K	27	B	44	s	61	9
11	L	28	C	45	t	62	+
12	M	29	D	46	u	63	/
13	N	30	E	47	v	pad	=
14	O	31	f	48	w		
15	P	32	g	49	x		
16	Q	33	H	50	Y		

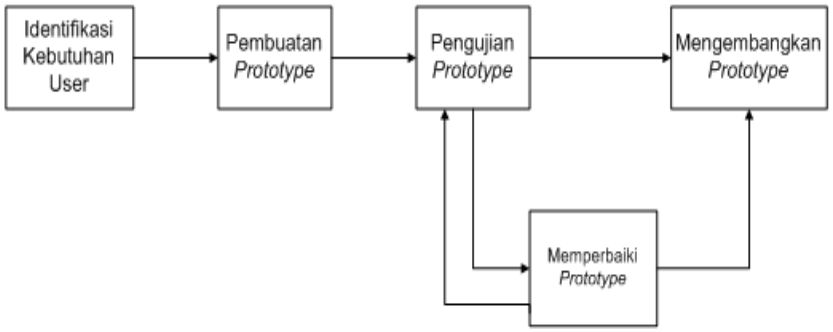
Teknik Encoding Base64 sebenarnya sederhana, jika ada satu (string) byte yang akan disandikan ke *Base64* maka caranya adalah.

1. Pecah string bytes tersebut ke per-3 bytes.
2. Gabungkan 3 bytes menjadi 24 bits. Dengan catatan 1 bytes=8 bit, sehingga 3x8=24 bits.
3. Lalu 24 bits yang disimpan di-*buffer* (disatukan) dipecah-pecah menjadi 6 bits, maka akan menghasilkan 4 pecahan.
4. Masing-masing pecahan diubah ke dalam nilai *decimal*, dimana maksimal nilai 6 bit adalah 63.
5. Terakhir, jadikan nilai-nilai decimal tersebut menjadi indeks untuk memilih karakter penyusunan dari Base64 dan maksimal adalah 63 atau indeks ke 64.

Dan seterusnya sampai akhir *string bytes* yang mau kita konversikan. Jika ternyata dalam proses *encoding* terdapat sisa pembagi, maka tambahkan sebagai penggenap sisa tersebut karakter =. Maka terkadang pada Base64 akan muncul satu atau dua karakter (=).

3. METODOLOGI

Karena halaman atau URL yang digunakan oleh aplikasi PWA Jawa Tengah sangat komplek, maka digunakan metode *prototyping*. Dengan tujuan, agar proses pengembangan sub-sistem yang diambil dari sebagian URL dalam *website* dapat dilakukan dengan lebih mudah. Pada pengembangannya, hasil *prototyping* ini dapat dilakukan pada bagian halaman atau URL yang lain.



Gambar 3.1 Metode Perancangan ``*Prototyping''*

3.1 Pengumpulan Kebutuhan

Setelah mempelajari karakteristik sistem PWA Jawa Tengah, beserta penggunaannya di lapangan, kemudian didukung pula dengan informasi yang didapatkan melalui hasil wawancara personal dengan tim programmer PWA Jawa Tengah, dalam tahap awal ini dapat dirumuskan sekurang-kurangnya 2 (dua) poin *requirement*, antara lain :

a. Dibutuhkan 1 (satu) server yang didalamnya terdapat website PWA Jawa Tengah.
b. Akses root dari website tersebut untuk mengubah konten yang ada didalamnya menggunakan media transfer file.

3.2 Pembuatan Prototype

Karena prototype yang dihasilkan harus mewakili kompleksitas dari halaman atau URL yang digunakan dalam sebuah aplikasi, dibutuhkan 2 (dua) tahapan, yaitu :

3.2.1 Design

Untuk melakukan proses *prototyping enkripsi URL* menggunakan metode Base64, dibutuhkan sebagian URL yang mewakili kompleksitas dalam sebuah website. Halaman atau URL yang diambil sebagai prototype untuk dilakukan pengujian keberhasilan enkripsi menggunakan metode Base64.

Sebagai data, URL asli akan diambil yang nantinya akan dilakukan enkripsi untuk mengamankan website PWA Jawa Tengah.

3.2.2 Coding PHP

Pada bagian coding ini menjelaskan tentang bahasa enkripsi yang berbasis metode Base64 yang diterjemahkan ke dalam bahasa pemrograman PHP. Adapun detail masing-masing function adalah sbb :

1. Function Enkripsi
   Langkah yang dilakukan :
   a) Mendeskripsikan inputan.
      function base64_encrypt ($plain_text, $password, $iv_len = 16)

   b) Menentukan panjang *plaintext* atau kata sebelum dienkripsi.
      $n = strlen($plain_text);

   c) Menentukan perulangan.
      while ($i < $n) {
      $block = substr($plain_text, $i, 16) ^ pack('H*', md5($iv));
      $enc_text .= $block;
      $iv = substr($block . $iv, 0, 512) $password;
      $i += 16;
      }
   d) Me-return value.
      return str_replace('+', '@', $hasil);

2. Function Dekripsi
   Langkah yang dilakukan :
   a) Mendeskripsikan inputan.
      function base64_decrypt ($enc_text, $password, $iv_len = 16)

   b) Menentukan panjang huruf setelah didekripsi.
      $n = strlen($enc_text);

   c) Menentukan perulangan.
      while ($i < $n) {
      $block = substr($enc_text, $i, 16);
      $plain_text .= $block ^ pack('H*', md5($iv));
      $iv = substr($block . $iv, 0, 512) ^ $password;
      $i += 16;
      }
   d) Me-return value.
      return preg_replace('/\\x13\\x00*$/', '', $plain_text);

3.3 Evaluasi Prototype

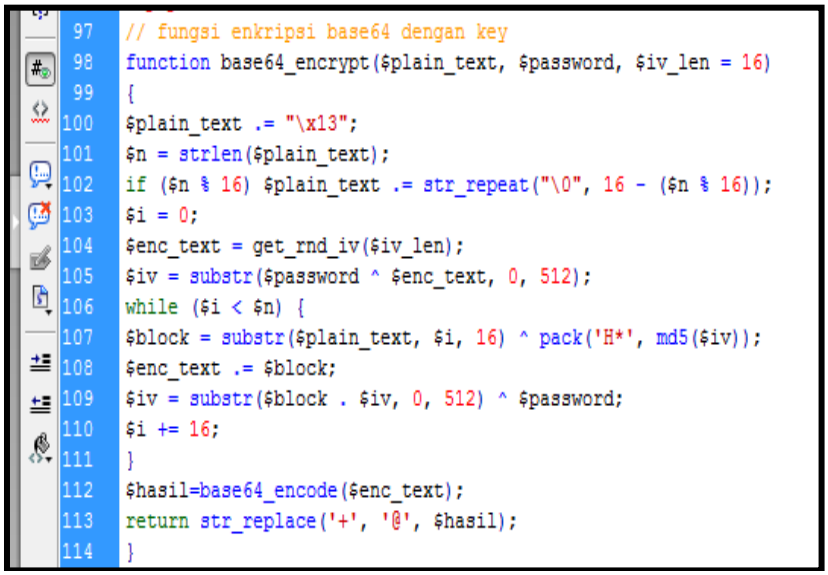
Melalui evaluasi yang dilakukan bersama dengan tim PWA Jawa Tengah, didapatkan kesimpulan bahwa *prototype* yang dihasilkan sudah memenuhi semua aspek *requirement* sistem. Yang perlu diperhatikan adalah pada sisi *coding*, *coding encrypt* diletakkan pada file "index.php", hal ini dimaksudkan agar semua alamat URL dapat terenkripsi dan terdekripsi.

4. HASIL DAN PEMBAHASAN

4.1 Implementasi Coding

Seperti yang telah dijelaskan pada bab sebelumnya, untuk mengimplementasikan rancangan coding enkripsi URL menggunakan metode Base64 perlu dilakukan dalam beberapa tahap. Yang perlu diperhatikan dalam melakukan implementasi adalah pada sisi coding. Coding encrypt diletakkan pada file index.php, hal ini dimaksudkan agar semua alamat URL dapat terenkripsi dan terdekripsi. Sebagai contoh adalah URL sebagai berikut http://pwajateng.com/news.php?hal=menu/informasi&lang=1&bid=1, bagian yang akan dienkripsi yaitu *hal=menu/informasi*. Setelah berhasil dienkripsi, maka bagian-bagian / variabel-variabel lain dapat pula dilakukan enkripsi.

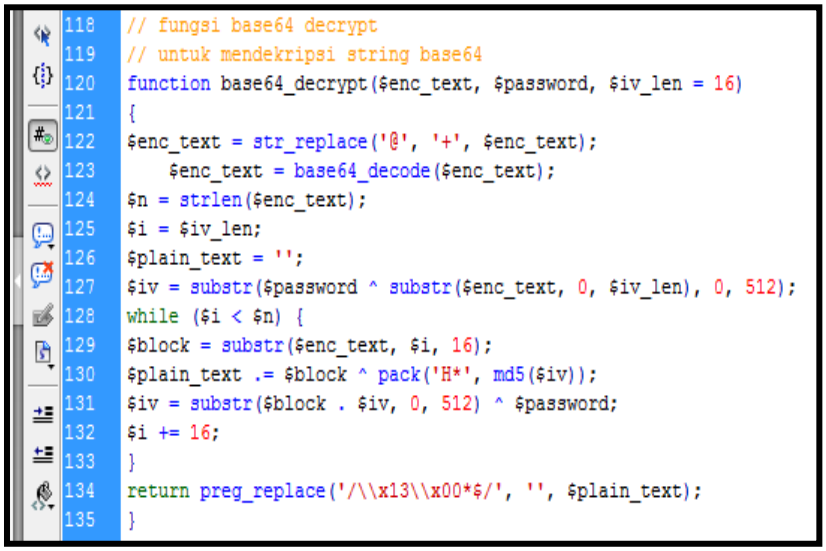
4.1.1 Implementasi Fungsi Enkripsi Base64 pada File Index Website



```
97  // fungsi enkripsi base64 dengan key
98  function base64_encrypt($plain_text, $password, $iv_len = 16)
99  {
100 $plain_text .= "\x13";
101 $n = strlen($plain_text);
102 if ($n % 16) $plain_text .= str_repeat("\0", 16 - ($n % 16));
103 $i = 0;
104 $enc_text = get_rnd_iv($iv_len);
105 $iv = substr($password ^ $enc_text, 0, 512);
106 while ($i < $n) {
107 $block = substr($plain_text, $i, 16) ^ pack('H*', md5($iv));
108 $enc_text .= $block;
109 $iv = substr($block . $iv, 0, 512) ^ $password;
110 $i += 16;
111 }
112 $hasil=base64_encode($enc_text);
113 return str_replace('+', '@', $hasil);
114 }
```

Gambar 4.1 : Coding Fungsi Enkripsi Base64

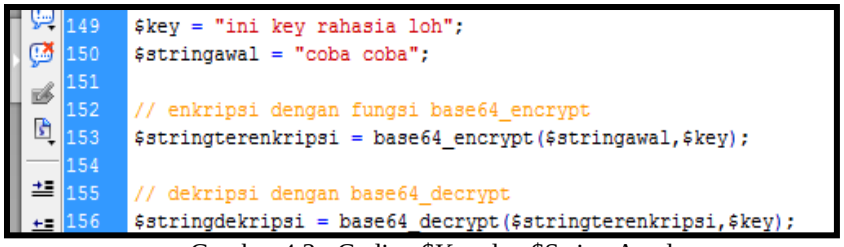
Pada gambar diatas dapat dilihat coding untuk mendefinisikan fungsi enkripsi Base64. Bagian ini berisi fungsi enkripsi mengubah *plaintext* menjadi teks yang ter-enkripsi. Untuk panjang karakter telah ditentukan yaitu 16 karakter. Selain mendefinisikan fungsi enkripsi, akan didefinsikan pula fungsi dekripsi. Untuk coding dekripsi Base64 dapat dilihat pada gambar dibawah ini :



```
118 // fungsi base64 decrypt
119 // untuk mendekripsi string base64
120 function base64_decrypt($enc_text, $password, $iv_len = 16)
121 {
122 $enc_text = str_replace('@', '+', $enc_text);
123     $enc_text = base64_decode($enc_text);
124 $n = strlen($enc_text);
125 $i = $iv_len;
126 $plain_text = '';
127 $iv = substr($password ^ substr($enc_text, 0, $iv_len), 0, 512);
128 while ($i < $n) {
129 $block = substr($enc_text, $i, 16);
130 $plain_text .= $block ^ pack('H*', md5($iv));
131 $iv = substr($block . $iv, 0, 512) ^ $password;
132 $i += 16;
133 }
134 return preg_replace('/\\x13\\x00*$/', '', $plain_text);
135 }
```

Gambar 4.2 : Coding Fungsi Dekripsi Base64

Pada gambar diatas merupakan coding fungsi dekripsi Base64. Bagian ini berisi tentang proses mendekripsikan hasil dari enkripsi plaintext sebelumnya untuk menjadi plainteks awal seperti sebelum dienkripsi. Selanjutnya adalah mendefinisikan variabel key yang disimbolkan dengan '$key', yang dapat diisi sesuai dengan keinginan penulis yang nantinya akan dikombinasikan dengan variabel string awal yang disimbolkan dengan '$stringawal' yang sudah didefinisikan. Untuk codingnya dapat dilihat pada gambar dibawah ini :

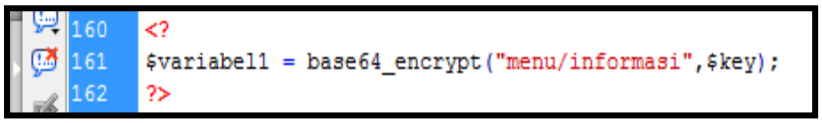


Gambar 4.3 : Coding $Key dan $String Awal

Pada gambar diatas dapat dilihat bahwa '$key' diisi dengan "ini key rahasia loh" dan '$stringawal' diisi dengan "coba-coba" karena pada dasarnya isian untuk $key dan $stringawal adalah bebas sesuai dengan keinginan. Selanjutnya adalah $stringawal akan diganti dengan "menu/informasi" dikombinasikan dengan $key dan Base64_ encrypt untuk menghasilkan $string yang terenkripsi atau dalam coding '$stringterenkripsi'. Sedangkan untuk dekripsinya adalah dengan mengkombinasikan '$stringterenkripsi' dengan $key menggunakan Base64_encrypt untuk menghasilkan $string yang terdekripsi atau dalam coding '$stringdekripsi'.

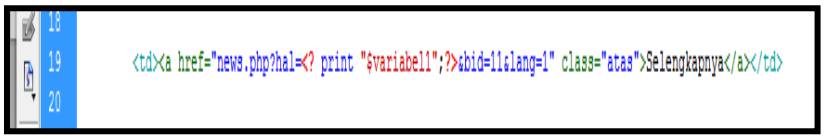
4.1.2 Mendefinisikan Menu Ke Variabel Enkripsi

Ada beberapa menu yang akan dienkripsi, menu-menu yang akan dienkripsi dilakukan enkripsi terlebih dahulu baru kemudian dilakukan dekripsi ke dalam variabel. Masing-masing menu hanya memiliki 1 variabel sehingga banyaknya variabel tergantung oleh banyaknya menu yang ada. Untuk codingnya dapat dilihat pada gambar berikut :



Gambar 4.4 : Coding Mendefinisikan Menu ke Variabel Enkripsi

Pada gambar diatas variabel diberi nama '$variabel1', '$variabel1' adalah variabel untuk menampung hasil enkripsi yang diperoleh dengan mengkombinasikan "menu/informasi" dan $key dengan menggunakan Base64_encrypt. Berikutnya adalah mengubah value dalam menu menggunakan variabel yang telah dienkripsi. Untuk coding mengubah value dalam menu tersebut dapat dilihat pada gambar dibawah ini :



Gambar 4.5 : Coding Mengubah Value ke dalam Menu Menggunakan Variabel

Pada gambar diatas terlihat bahwa value dalam menu telah diubah menggunakan variabel yang telah dienkripsi yaitu menjadi '$variabel1'. $variabel1 inilah variabel tempat menampung hasil enkripsi.

4.2 Pengujian Sub-Sistem

Pada tahap pengujian ini akan dilakukan 2 pengujian yaitu pengujian bahwa enkripsi telah berhasil dilakukan dan pengujian dengan memasukkan perintah *SQL Injection* pada URL sesudah dienkripsi dan sebelum dienkripsi.

4.2.1 Pengujian Penerapan Enkripsi pada URL Website

Pengujian merupakan tahap penting dalam membangun sebuah sub-sistem. Hasil pengujian yang didapat akan dijadikan sebagai tolak ukur dalam proses pengembangan selanjutnya. Pengujian awal yang diakukan adalah dengan memastikan bahwa fungsi enkripsi menggunakan metode Base64 dapat diterapkan sesuai dengan skema yang sebelumnya telah dibangun.
Pengujian akan dilakukan dengan menggunakan browser untuk mengakses ke website tersebut. Berikut hasil enkripsi :

Gambar 4.7 : Hasil Enkripsi Menggunakan Base64_encrypt

Pada gambar diatas memperlihatkan bahwa enkripsi pada variabel URL yaitu "hal" telah berhasil diterapkan. Dapat dilihat enkripsi pada value variabel "hal" yang merupakan isi dari $variabel1 yang telah dijelaskan pada penjelasan sebelumnya yaitu mengkombinasikan "menu/informasi" dengan $key menggunakan *Base64_encrypt*. Hasil enkripsi diatas akan selalu diperbaharui secara otomatis ketika dilakukan refresh atau reload page karena terdapat random karakter. Random karakter ini terdiri dari angka dan huruf secara acak yang selalu berubah-ubah setiap kali dijalankan.

Gambar 4.8 : Hasil Enkripsi Menggunakan Base64_encrypt Setelah Dilakukan Reload Page

Pada gambar diatas terlihat hasil enkripsi yang berbeda setelah dilakukan *refresh page* atau *reload page*. Inilah bukti bahwa hasil enkripsi terdapat *random* karakter, *random* karakter diatas nantinya akan dijalankan setiap saat beserta *cookie random* karakter yang akan selalu di update setiap kali halaman dikunjungi atau di *reload*, dan *random* karakter ini akan dijadikan sebagai sisipan terhadap request url yang di enkripsi menggunakan *Base64_encrypt*, sehingga *request* url tersebut tidak diketahui *field* dan *value* nya dan akan selalu berubah-ubah setiap saat. Hal ini membuktikan bahwa enkripsi dapat bersifat *random* setiap kali dilakukan akses ke halaman tersebut.

4.2.2 Pengujian dengan *SQL Injection* pada *URL* Sebelum di Enkripsi

Dengan menerapkan *SQL Injection* pada *URL* website sebelum dilakukan enkripsi maka dapat ditemukan *username* dan *password* dari website PWA Jawa Tengah. Tentunya dengan langkah-langkah dalam melakukan *SQL Injection*. Dengan login ke website tersebut maka data yang ada pada website menjadi tidak aman. Oleh karena itu untuk mengamankan data-data serta informasi yang ada pada website tersebut dilakukan enkripsi pada variabel URL website seperti yang sudah dilakukan pada sub-bab sebelumnya.

4.2.3 Pengujian dengan SQL Injection pada URL Sesudah di Enkripsi

Penerapan enkripsi menggunakan metode Base64 terbukti mampu mencegah terjadinya *SQL Injection* pada *URL* website. Karena penerapan *SQL Injection* pada *URL* website yang sudah terenkripsi tidak menemukan error pada tampilan website. Dengan tidak adanya pesan error maka langkah-langkah *SQL Injection* tidak dapat dilanjutkan ke tahap berikutnya.

4.3 Kesimpulan Pengujian

Dari pengujian subsistem yang dilakukan, terlihat bahwa fungsi enkripsi menggunakan metode Base64 telah berjalan dengan baik. Fungsi *random* karakter juga dapat berfungsi secara otomatis ketika dilakukan refresh atau reload page.

Pada pengujian perbandingan menggunakan serangan *SQL Injection* yang dilakukan pada variabel *URL* sebelum ter-enkripsi dan sesudah ter-enkripsi, dapat terlihat bahwa penerapan *SQL Injection* pada *URL* sebelum di enkripsi ketika ditambahkan tanda petik satu (‘) dibelakang variabel *bid* akan menampilkan pesan error, maka proses akan berlanjut ke tahap berikutnya sehingga dapat mengeksekusi database melalui *URL* dan mendapatkan akses untuk memperoleh informasi penting suatu website. Sedangkan pada *URL* yang telah di enkripsi ketika diterapkan *SQL Injection* dengan menambahkan tanda petik satu (‘) pada variabel *bid* maka halaman tidak menampilkan pesan error.

5. KESIMPULAN

Berdasarkan hasil analisa dan pengujian sistem dalam hal ini website PWA Jawa Tengah setelah dilakukan enkripsi pada variabel URL menggunakan metode Base64 dapat disimpulkan sebagai berikut :

a. Perancangan website PWA Jateng dengan penerapan enkripsi pada variabel URL menggunakan metode Base64 dapat memberikan solusi untuk mencegah terjadinya serangan SQL Injection.
b. Penerapan *SQL Injection* pada variabel URL website PWA sebelum di enkripsi dapat memperoleh *username* dan *password* untuk login ke admin.

DAFTAR PUSTAKA

[1] Fairuzabadi, Muhammad. 2010. *Implementasi Kriptografi Klasik Menggunakan Borland Delphi*. Jurnal Dinamika Informatika. (diakses tanggal 13 Juni 2013)
[2] Puspita, Oky Ristyarani (dkk). 2011. *Implementasi Pencegahan Serangan SQL Injection Menggunakan GreenSQL*. Bandung : Politeknik Telkom
[3] Nufus, Hayatun. 2009. "Pembuatan Aplikasi Kriptografi Algoritma Base64 " (diakses tanggal 11 april 2013)
[4] Adriansyah, Yusuf . "Enkripsi Sederhana dengan Base64 dan Substitusi Monoalfabetik ke Huruf Non-Latin". Bandung : ITB
[5] Febrian Wahyu, Adriana, Febry. 2012 "Penerapan Algoritma Gabungan RC4 dan Base64 Pada Sistem Keamanan E-Commerce"
[6] Hamzah, Rio.2011. Implementasi Algoritma RSA dan Blowfish Untuk Enkripsi dan Dekripsi Data Menggunakan Delphi 7. Jakarta : Universitas Islam Syarif Hidayatullah
[7] http://id.wikipedia.org/wiki/Kriptografi, diakses tanggal 27 April 2012
[8] Ariyus, D. (2009). *Keamanan Multimedia*. Yogyakarta: Andi Offset.
[9] http://id.scribd.com/doc/66899791/Pengertian-Hacker-Makalah-Lengkap