

OPTIMALISASI KEAMANAN WEBSITE MENGGUNAKAN CAPTCHA (*Completely Automated Public Turing test to tell Computerd and Human Apart*)

Imam Riadi

¹⁾Program Studi Ilmu Komputer Universitas Ahmad Dahlan Yogyakarta
Jl. Prof. Soepomo, Janturan, Yogyakarta Telp (0274)-379418
e-mail : imam_riadi@uad.ac.id

Abstrak

Keamanan sistem komputer berbasis web merupakan hal yang harus diperhatikan, karena sistem berbasis web sangat rentan terhadap tindakan kejahatan komputer. Serangan terhadap sistem tersebut banyak dilakukan dengan menggunakan robot program (bots) yang menyamar layaknya sebagai manusia. Berdasarkan masalah tersebut diperlukan metode pengujian untuk membedakan proses validasi antara robot dengan manusia. Metode tersebut sering disebut dengan reverse turing test. CAPTCHA (*Completely Automated Public Turing test to tell Computers and Human Apart*) merupakan salah satu dari metode reverse turing test. Berdasarkan hasil penelitian ini menunjukkan bahwa CAPTCHA dapat diimplementasikan dengan baik dan dapat menjadi solusi untuk memproteksi sistem komputer berbasis web dari serangan robot, sehingga sistem menjadi relatif lebih aman dan terhindar dari usaha-usaha penyusupan yang dilakukan oleh pihak yang tidak bertanggung jawab

Keyword : Reverse Turing Test, CAPTCHA, Keamanan, Website

1. PENDAHULUAN

Perkembangan teknologi informasi terutama dalam bidang jaringan komputer berkembang dengan pesat khususnya penggunaan Internet. Tidak semua pengguna Internet dapat menghargai privasi orang lain. Banyak pengguna Internet yang dengan sengaja mencari keuntungan dan memanfaatkan kelemahan sistem orang lain untuk melakukan tindakan kejahatan komputer. Serangan terhadap sebuah sistem komputer kebanyakan dilakukan melalui Internet, serangan itu terutama dilakukan untuk mendapatkan sesuatu sesuai dengan keinginan penyerang. Dalam Internet banyak program yang diotomatiskan untuk membanjiri tempat polling dan tempat pemilihan secara online. Salah satu alternatif untuk menghindari serangan yang terjadi pada Internet menggunakan metode pengujian atau tes terhadap pengguna sistem komputer. Pengujian tersebut bertujuan mengetahui bahwa suatu sistem diakses oleh pengguna atau mesin script yang di otomatiskan. Pengujian ini sering dikenal dengan metode *Reverse Turing Test*. CAPTCHA adalah salah satu aplikasi metode *Reverse Turing Test*. Penggunaan CAPCHA dapat menjadi solusi yang tepat, sehingga Internet dapat ditingkatkan keamanannya dari pihak-pihak penyerang yang tidak bertanggung jawab. Penelitian ini bertujuan untuk mengimplementasikan CAPTCHA dengan script PHP untuk proses autentikasi dan memproteksi website dari serangan script robot yang telah disiapkan oleh para penyerang.

2. TINJAUAN PUSTAKA

A. CAPTCHA

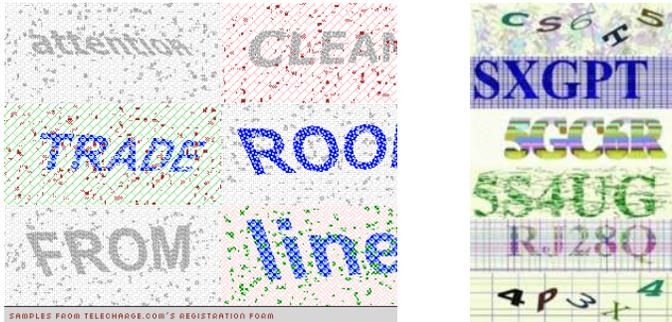
CAPTCHA (*Completely Automated Public Turing test to tell Computers and Human Apart*) pada dasarnya adalah suatu program yang sebagian besar manusia dapat melewatinya, akan tetapi komputer tidak dapat melewatinya (Ahn,2004). Aplikasi CAPTCHA banyak digunakan pada penyedia web mail contohnya Hotmail dan Yahoo. CAPTCHA dikembangkan untuk mencegah program robot atau bots yang menciptakan ratusan email account untuk mengirimkannya ke user (mori,2003). Bots ini digunakan oleh spammer untuk melakukan penyerangan terhadap sistem dengan menggunakan **HTTP POST request submission**. Program robot akan mengambil nilai variabel yang terdapat pada **HTTP POST** request tersebut dari form yang akan disubmit sebelumnya dan mengirimkannya kembali secara berulang-ulang. Penyerang dapat dengan mudah melakukan hal tersebut dengan menulis script menggunakan bahasa perl.

CAPTCHA terbagi dalam beberapa jenis, antara lain :

1) Berdasarkan Visual (*Visual Based*)

Virtual Based CAPTCHA memiliki beberapa variasi, yang paling umum digunakan saat ini adalah teks yang dimiringkan dan ditempelkan pada sebuah gambar dan pengenalan bentuk. CAPTCHA yang menggunakan teks dimiringkan yang ditempelkan pada gambar disebut *Gimpy*, *EZ-Gimpy* adalah varian dari *Gimpy*, *Pessimial print* dan *buffletext*. *Gimpy* pertama kali dikembangkan oleh Luis Von Ahn dari Carnegie Mellon University

yang mendesain versi paling sederhana dari Gimpy, disebut EZ-Gimpy (Tsui,2004). Ez-Gimpy yang sekarang ini digunakan oleh Yahoo dan suatu versi serupa digunakan oleh Hotmail. Perbedaan yang mendasar antara *gimpy* dan *EZ-gimpy* adalah *Gimpy* memiliki tiga atau lebih kata yang dimiringkan yang ditempelkan pada suatu gambar, sedangkan *EZ-Gimpy* hanya memiliki satu kata yang dimiringkan pada suatu gambar.



Gambar 1. Contoh Visual Based CAPTCHA

2) Berdasarkan suara (*Sound Based*)

Sound based CAPTCHA kebanyakan digunakan untuk membantu mereka yang tuli atau mempunyai masalah dengan pendengaran. Suatu contoh *sound based CAPTCHA* adalah bunyi yang sesuai. CAPTCHA ini digunakan pada Hotmail, Yahoo dan Altavista sebagai tambahan terhadap *CAPTCHA visual based* ketika pendaftaran sebuah email account untuk masing-masing penyedia layanan email ini. Tes ini menjalankan klip audio yang berisi rekaman suatu urutan kata atau angka-angka yang dimiringkan dan jika kata atau angka-angka yang diduga tepat maka dapat melewati tes ini. (Tsui,2004).

B. GD LIBRARY

PHP tidak terbatas hanya pada pembuatan output HTML saja, PHP memungkinkan untuk membuat dan memanipulasi gambar seperti garis kotak, lingkaran dan poligon dengan warna yang disesuaikan dengan keinginan. Pembuatan gambar tersebut PHP memanfaatkan pustaka grafik yang bernama GD yang dapat di download di <http://www.boutel.com/gd/>. Pustaka GD mendukung format gambar seperti JPEG, GIF, PNG, SWF, TIFF dan JPEG2000.

Pustaka GD memiliki sejumlah fungsi untuk operasi gambar. Fungsi-fungsi itulah yang digunakan untuk memanipulasi dan membuat gambar dalam PHP. Adapun fungsi-fungsi tersebut adalah :

- 1) Fungsi untuk menciptakan gambar
`imagecreate (int x_size, int y_size)`
Fungsi ini akan mengembalikan nilai bilangan bulat yang menyatakan identitas gambar. `x_size` menyatakan koordinat X, dan `y_size` menyatakan koordinat Y.
- 2) Fungsi untuk menentukan warna
`int imagecolorallocate (resource image, int red, int green, int blue)`
Argumen *resource image* menyatakan identitas gambar, berupa nilai kembalian yang dihasilkan oleh fungsi `imagecreate`. Dan argumen `int red`, `int green`, `int blue` menyatakan komponen warna yang menyusun suatu warna. Nilai dari ketiga argumen ini berkisar antara 0 sampai 255.
Fungsi ini menghasilkan nilai kembalian berupa bilangan bulat yang menyatakan identitas warna.
- 3) Penampilan gambar ke browser
`int imagejpeg (resource image [,string filename [, int quality]])`
`int imagepng (resource image [, string filename])`
`int imagegif (resource image [, string filename])`
Fungsi ini untuk menampilkan gambar ke web browser adalah `imagejpeg` untuk gambar JPEG, `imagepng` untuk gambar PNG, dan `imagegif` untuk gambar GIF.
- 4) Dealokasi memori
`int imagedestroy (resource image)`
Operasi gambar memerlukan pengalokasian memori yang cukup besar. Cara yang digunakan untuk membebaskan memori yang digunakan adalah dengan memanggil fungsi `imagedestroy`.
- 5) Pembuatan garis

```
int imageline (resource image, int x1, int y1, int x2, int y2, int color)
int imagedashedline (resource image, int x1, int y1, int x2, int y2, int color)
```

Garis dapat dibuat dengan menggunakan fungsi `imageline` dan `imagedashedline` untuk garis putus-putus.

6) Pembuatan lingkaran atau elips

```
int imagearc (resource image, int cx, int cy, int w, int h, int s, int e, int color)
int imageellipse (resource image, int cx, int cy, int w, int h, int s, int e, int color)
```

Lingkaran atau elips dapat dibuat dengan menggunakan fungsi `imagearc` atau pada PHP 4.0.6 dengan pustaka GD versi 2.0.2 ke atas terdapat fungsi `imageellipse`.

7) Pembuatan kotak

```
int imagerectangle (resource image, int x1, int y1, int x2, int y2, int col)
int imagefilledrectangle (resource image, int x1, int y1, int x2, int y2, int color)
```

Kotak dapat dibuat dengan menggunakan fungsi `imagerectangle` dan jika menginginkan kotak tersebut diarsir atau diberi warna pada bagian dalam kotak dapat menggunakan fungsi `imagefilledrectangle`.

8) Pembuatan poligon

```
int imagepolygon (resource image, array points, int num_points, int color)
int imagefilledpolygon (resource image, array points, int num_points, int color)
```

Poligon dapat dibuat dengan menggunakan fungsi `imagepolygon` dan jika menginginkan kotak tersebut diarsir atau diberi warna pada bagian dalam kotak dapat menggunakan fungsi `imagefilledpolygon`.

9) Penampil teks

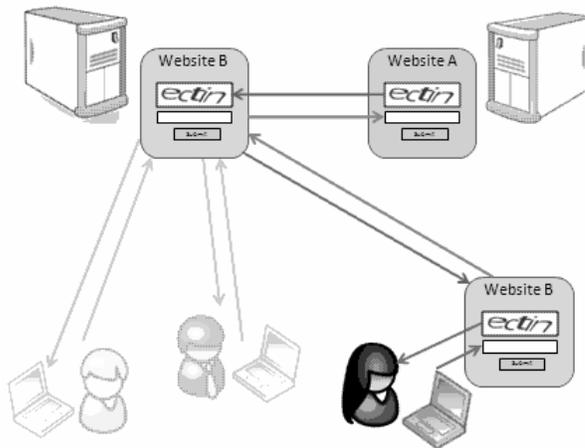
```
int imagechar (resource image, int font, int x, int y, string c, int color)
int imagecharup (resource image, int font, int x, int y, string c, int color)
int imagestring (resource image, int font, int x, int y, string s, int color)
array imagettftext (resource image, int size, int angle, int x, int y, int color,
string fontfile, string text)
array imagettfbbox (int size, int angle, string fontfile, string text)
```

Terdapat dua buah fungsi yang dapat digunakan untuk menampilkan sebuah karakter. Fungsi tersebut adalah `imagechar` dan `imagecharup`. `imagecharup` digunakan untuk menampilkan karakter yang menghadap ke atas. String atau sederetan karakter juga dapat ditampilkan menggunakan fungsi `imagestring`. PHP juga menyediakan dua buah fungsi yang digunakan untuk menampilkan True Type Font yaitu `imagettftext` dan `imagettfbbox`.

C. REVERSE TURING TEST

Reverse Turing Test merupakan suatu metode pengujian untuk membedakan antara manusia dan program komputer yang diotomatiskan (robot). Tes ini dapat dengan mudah dilewati oleh user manusia, tetapi sangat sulit untuk program komputer yang diotomatiskan. Alternatif lain yang digunakan untuk melakukan tes tersebut adalah dengan menggunakan CAPTCHA. Tes tersebut harus memiliki kriteria sebagai berikut :

- 1) Tes dibuat secara otomatis
- 2) Tes sangat mudah digunakan oleh user manusia, yaitu tes yang dihasilkan harus mudah dipecahkan oleh *user* manusia.
- 3) Tes sangat sulit untuk digunakan oleh program komputer
- 4) Kecil kemungkinan untuk menebak jawaban dengan tepat.



Gambar 2. Cara kerja *reverse turing test*

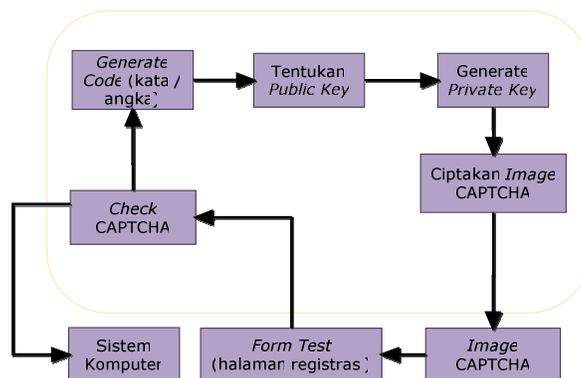
3. METODE PENELITIAN

Metode yang digunakan dalam penelitian ini menggunakan studi literatur dan eksperimen. Sedangkan kebutuhan perangkat lunak dan perangkat keras yang digunakan antara lain : seperangkat komputer dengan beberapa software pendukung antara lain : Apache sebagai *web server*, PHP 4.2.0 sebagai *interpreter*, *Library GD 2.0*, Opera dan Mozilla Firefox sebagai *browser* dan *text editor*.

4. HASIL DAN PEMBAHASAN

A. CAPTCHA sebagai alternatif Reverse Turing Test

CAPTCHA dapat digunakan sebagai alternatif reverse turing test untuk melindungi suatu website dari para penyerang yang tidak berhak. Berikut ini penjelasan cara kerja CAPTCHA.



Gambar 3. Skema cara kerja CAPTCHA

- 1) Di *generate* rangkaian kode yang berupa huruf dan angka atau sebuah kata, dan kemudian akan dihasilkan *image CAPTCHA*. Sesuai dengan algoritma *public key cryptography* akan digenerate *private key* yang hanya dapat berpasangan dengan rangkaian kode yang dihasilkan tersebut (*public key*).
- 2) *Private key* akan dirender sebagai *hidden field* di *form*, dan harus ada *input box* yang akan menerima masukan dari user berupa *public key* atau *image CAPTCHA* yang ditampilkan.
- 3) *Private key* dan *public key* yang akan *submit* oleh user akan dibandingkan dengan algoritma *public key cryptography*, jika ada kecocokan, berarti captcha sah dan jika tidak ada kecocokan, berarti ada indikasi yang melakukan *submit* bukan manusia.

B. Implementasi CAPTCHA

Pada bagian ini akan dijelaskan cara pembuatan CAPTCHA dengan menggunakan pemrograman PHP. PHP mendukung Object Oriented Programming, maka dalam pembuatan modul CAPTCHA ini digunakan OOP pada PHP. Sebelum pembuatan modul ini perlu dilakukan beberapa tahapan antara lain instalasi dan konfigurasi pustaka GD, PHP dan Apache sebagai web server untuk mengimplementasikan modul tersebut.

1) Instalasi Pustaka GD

Pustaka GD adalah pustaka yang menyediakan fungsi-fungsi untuk memanipulasi gambar pada web. Pustaka GD memiliki sejumlah fungsi untuk operasi gambar. PHP menggunakan pustaka GD untuk keperluan manipulasi gambar. Sebelum menggunakannya maka perlu diinstall terlebih dahulu. Pustaka GD yang digunakan dalam penelitian ini adalah versi 2.0.0.

Cara yang dilakukan untuk instalasi pustaka GD adalah dengan source code yang di download di <http://www.boutel.com/gd/>. Adapun caranya adalah Mengekstrak file kode sumber atau source code dari pustaka GD

```
# gunzip < gd-2.0.0.tar.gz | tar xvf -
```

Proses ekstraksi file ini akan secara otomatis membuat direktori gd-2.0.0 dan file source code berada pada direktori tersebut. Setelah proses ekstraksi selesai dilakukan maka perlu mengkonfigurasi dan kompilasi terhadap kode sumber tersebut sebelum diinstalasi

```
# cd gd-2.0.0
# ./configure --with-prefix=/usr/local/gd2 --with-jpeg-dir=/usr/local --with-png-dir=/usr --with-xpm-dir=/etc/X1186
# make
# make install
```

2) Instalasi PHP dan Apache Web Server

PHP dan Apache web server biasanya telah disertakan secara default pada CD instalasi Linux. Namun jika tidak menyertakan paket ini pada saat instalasi Linux, dapat dilakukan instalasi paket tersebut secara manual, baik dengan menggunakan paket RPM pada distribusi Linux RedHat dan turunannya

3) Pembuatan Modul CAPTCHA

Modul CAPTCHA yang akan dibuat adalah *EZ-Gimpy*. *EZ-Gimpy* adalah *visual based CAPTCHA* yang terdiri dari image dan sebuah kata teks yang digabungkan didalamnya.

Modul tersebut terdiri dari beberapa file yang saling terkait. File tersebut diantaranya adalah *file script* PHP yang berekstensi php, *File True Type Font* yang berekstensi ttf, dan file pustaka untuk kata-kata yang akan ditampilkan.

a) Pembuatan file konfigurasi **config.inc.php**

File ini berisi konfigurasi untuk modul CAPTCHA, Konfigurasi tersebut digunakan untuk memudahkan dalam penggunaan modul tersebut. Berikut ini potongan script file config.inc.php

```
// Physical directory untuk modul CAPTCHA
$config ['DOC_ROOT'] = "c:/www/captcha";
$config ['FONT_DIR'] = $config['DOC_ROOT']. "/font";

//TrueType Font
$config ['FONT'] = "comic.ttf,arial.ttf,lsansdi.ttf";

//Image Type (png, jpeg)
$config ['IMAGE_TYPE'] = "png";

//Pesan untuk input box pada form
$config ['MESSAGE_TEXT'] = "Masukkan kata yang tertera pada gambar";
```

b) Pembuatan file utama **captcha.class.php**

File ini berupa class utama dari modul CAPTCHA tersebut. File ini berisi *script-script* yang digunakan dalam pembentukan image CAPTCHA.

captcha
-PublicKey -word -image Type -image -image Text -font -width -height
+captcha(in width, in height) -get Word() -getFilename(in public) -generatePrivateKey(in public) -drawText () -drawImage () -destroy() -makeCaptcha () +heckCaptcha(in public, in private) +display()

Gambar 4. Diagram class CAPTCHA

Berikut ini potongan *script PHP* untuk mendeklarasikan class CAPTCHA dan pendeklarasian variabel pendukung.

c)

```
class captcha (  
    // property  
    var $publicKey;  
    var $word;  
    var $imageType;  
    var $image;  
    var $imageText;  
    var $width;  
    var $height;  
    var $font;
```

File ini adalah file pendukung yang berisi kata-kata yang akan ditampilkan sebagai *public key* pada *image CAPTCHA*. Kata-kata yang terdapat dalam file tersebut nantinya akan diambil secara random. File tersebut adalah sebuah *file text ASCII*. Kata-kata yang terdapat dalam file tersebut dengan mudah dapat ditambah atau dikurangi sesuai dengan kebutuhan. Setiap kata akan dipisahkan oleh baris, artinya setiap kata harus ditulis dalam baris yang berbeda.

```
test  
coba  
system  
computer  
security
```

4) Pengujian CAPTCHA

Pada tahapan ini akan dilakukan pengujian untuk mengetahui modul tersebut sudah berjalan dengan baik atau belum. Pengujian ini dilakukan oleh peneliti sehingga hasil outputnya sesuai dengan yang diharapkan. Langkah-langkah yang dilakukan dalam pengujian modul CAPTCHA adalah dengan melihat output image yang dihasilkan, kata yang ditampilkan serta distorsi yang dihasilkan. Hasil pengujian ini menunjukkan bahwa modul CAPTCHA sudah berjalan dan hasil outputnya sudah sesuai dengan harapan.

5. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan dapat disimpulkan bahwa :

- a) Telah dihasilkan sebuah modul CAPTCHA dengan menggunakan script PHP yang dapat memproteksi website dari serangan robot,
- b) berdasarkan hasil pengujian membuktikan bahwa modul CAPTCHA dapat bekerja dengan baik sesuai dengan harapan.

6. DAFTAR PUSTAKA

Captcha Project, <http://www.captcha.net>

Ahn, L.V., Blum, M., dan Langford, J., 2004, "Telling Humans and Computers Apart Automatically", http://www.cs.cmu.edu/~biglou/captcha_cacm.pdf

Chew, M., Baird, H. S., 2003, "Baffletext : A human Interactive Proof" <http://cs.berkeley.edu/~mmc/papers/drr02.pdf>.

Elmatsani, H. M., 2004, "Verification Word", <http://program-ruti.org/veriword/>

Indrajit, R.E., Prastowo, B.N., dan Yuliyardi, Rofiq, 2002, "Memahami Sekuriti Linux", Elexmedia komputindo, Jakarta.

Kadir, Abdul, 2003, "Dasar Pemrograman Web Dinamis Menggunakan PHP", Andi Offset, Yogyakarta

Mori, G., Malik, J., 2003, "Recognizing Objects in Adversarial Clutter : Breaking a Visual CAPTCHA", http://www.cs.berkeley.edu/~mori/research/papers/mori_cvpr03.pdf

Nugroho A, 2004, "Implementasi CAPTCHA sebagai Pendukung Proses Registrasi Online" Tugas Akhir, Universitas Ahmad Dahlan, Yogyakarta.

Nogajski, H., 2004 "PHP-Class hn_captcha Version 1.2",

<http://hn273.users.phpclasses.org/browse/package/1569.html>

O'Brien, Matt, 2001, "Computer or Human ? New program can tell".

http://www.captcha.net/news/computer_human.html.

Pinkas, B., Sander, T., 2004, "Securing password against dictionary attacks" www.pinkas.net/papers/pwdweb.pdf

Roy, G.M, 2004, "Session Captcha v.1.0", http://www.gavinroy.com/downloads/captcha_v1.tar.gz

Suprapti, Iswanti, 2003, "Studi Sistem Keamanan Data dengan Metode Public Key Cryptography", Tugas Akhir, Institut Teknologi Bandung.

Tsui, Annete, 2004, "CAPTCHA",

<http://www.cs.sjsu.edu/faculty/stamp/CS265/projects/SPR04/section2/papers/Tsui.doc>

<http://www.kecoak-elektronik.net/log/2007/11/15/breaking-the-captcha/>

<http://www.boutell.com/gd/manual2.0.28.html>