

# Watermark pada Game

Ahmad Fauzan/13510004  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia  
13510004@std.stei.itb.ac.id

**Abstract**— Pembajakan terhadap software semakin banyak. Pembajakan terbesar ada pada Game. Salah satu solusinya adalah menambahkan *signature* yang menandakan hak cipta. Pada media citra ada penerapan *signature* yang disebut *watermark*. Pada media game akan digunakan hal yang sama. Sama halnya seperti *watermark* pada citra, *watermark* pada aplikasi pengembangan permainan sebenarnya bisa setidaknya mengurangi tingkat pembajakan. Makalah akan lebih menjelaskan detail teknik, dampak *watermark* pada pengembangan game dari sisi performa dan bisnis, dan kemungkinan penyelesaian kasus pembajakan yang terjadi dengan menggunakan *watermark* pada game.

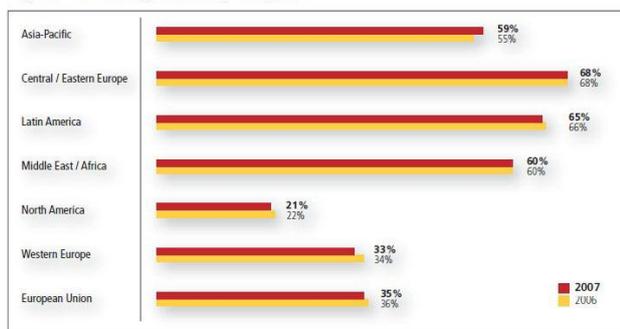
**Index Terms**—kriptografi, watermark, game

## I. PENDAHULUAN

Selain pembajakan terhadap perangkat lunak, pembajakan game juga telah merambah ke segala penjuru negara. Pembajakan ini sangat merugikan para publisher dan pengembang game itu sendiri. Pembajakan itu sendiri dikisar menyebabkan kerugian yang tidak sedikit. Menurut data tahun 2006-2007, Eropa merupakan daerah yang tingkat pembajakan perangkat lunak paling tinggi. Disusul oleh Amerika latin dan Asia Tengah.

Figure 1. Piracy Rate by Region

Figure 1. Piracy Rate by Region

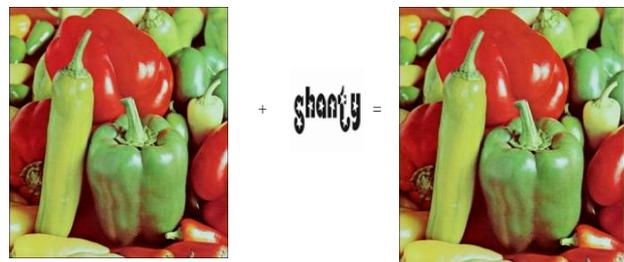


Berita terbaru datang dari Amerika Utara, Asosiasi Entertain Kanada melansir bahwa Industri Game Komputer di U.S dan Kanada kehilangan mencapai \$3.5 Billion setiap tahun karena pembajakan game. Kerugian itu merupakan 1/5 dari total keuntungan yang didapat

(Agustus 2012).

Jika kita melirik pada media lain, seperti citra, maka ada beberapa teknik yang bisa digunakan untuk menjaga hak cipta dari sebuah dokumen digital. *Digital Watermarking* merupakan sebuah teknik penyisipan informasi yang menyatakan kepemilikan data multimedia. *Watermark* dapat berupa teks, logo, audio, data biner, atau barisan-barisan bilangan riil. Tujuan utama dari *watermarking* itu sendiri adalah memberikan perlindungan hak cipta.

Figure 2. Watermarking pada Citra



Jika pada media digital lain bisa dilakukan teknik *watermarking*, apakah game sebagai media interaktif juga bisa dilakukan teknik *watermarking* untuk menandai hak cipta sebuah game ?

Berita terbaru tentang *Game Watermarking* diduga telah diimplementasikan oleh salah satu pengembang game terkenal, yaitu Blizzard. Salah seorang pengguna melaporkan terjadi keanehan saat dia memodifikasi citra screenshot. Apakah mungkin *watermarking* pada Citra bisa diimplementasikan pada Game.?

## II. TEKNIK WATERMARKING PADA CITRA

*Watermark* sebenarnya terbagi 2 menurut penglihatan. Pertama, *watermark visible* yang bisa dilihat tanpa bantuan alat apapun. *Watermark* yang diberikan dapat dilihat dengan mudah.

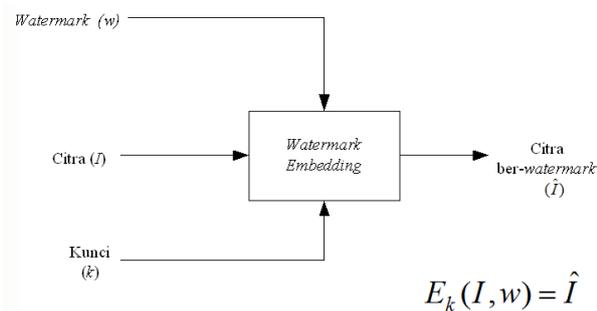
Kedua, *watermark invisible* yang tidak bisa dilihat secara kasat mata. *Watermark* ini merupakan aplikasi dari steganografi. Steganografi adalah sebuah teknik menyisipkan sebuah pesan ke dalam sebuah media lain.

Pada *invisible watermark* ada beberapa syarat yang harus dipenuhi untuk menjamin sebuah citra itu aman.

1. *imperceptible*  
Citra hasil *watermarking* tidak dapat dibedakan dengan citra yang asli.
2. *robustnes*  
Citra hasil *watermarking* jika dimodifikasi dan/atau ditambah, tidak dapat merupah watermark yang telah terembed di dalamnya.
3. *secure*  
*Watermark* tidak dapat terdeteksi oleh orang yang tidak berhak.

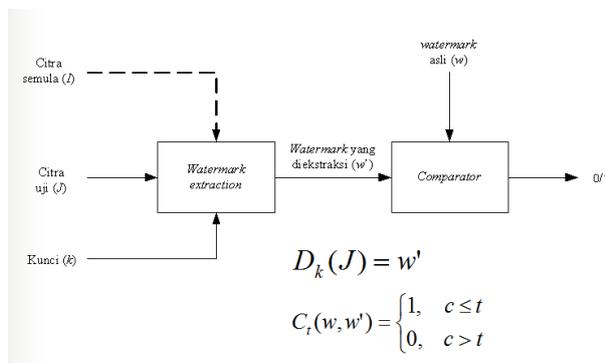
Ada 2 proses penting dalam *watermarking* yaitu, penyisipan dan ekstraksi. Penyisipan adalah proses dimana watermark dimasukkan ke dalam citra tertentu. Sedang ekstaksi adalah pengambilan atau pengecekan sebuah *watermark* pada citra.

**Figure 3. Proses Penyisipan Watermark**



Proses penyisipan *watermark* terdapat beberapa teknik. Teknik tersebut hampir mirip dengan steganografi. Bisa menggunakan 1 atau 2 bit LSB dari warna. Untuk mendapatkan proteksi tambahan. Proses penyisipan juga bisa menggunakan fungsi enkripsi agar watermark tidak mudah dibaca dengan mengedit file citra.

**Figure 4. Proses Ekstraksi Watermark**



Proses ekstraksi harus menggunakan algoritma dan teknik yang sesuai dengan proses penyisipan. Jika teknik yang digunakan berbeda maka kemungkinan besar tidak diperoleh watermark yang diinginkan. Teknik yang dapat

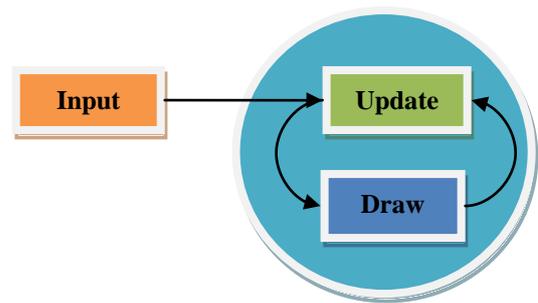
digunakan sama seperti teknik penyisipan yaitu memodifikasi 1 atau 2 bit LSB dari warna.

#### IV. PROSES/SIKLUS PADA GAME

Game merupakan gabungan dari beberapa media. Media-media tersebut antara lain gambar, video, suara dan masukan.

Siklus sederhana pada game terdiri dari *update*, *draw* dan *input*. *Update* merupakan komponen saat game sedang mengupdate logika dari game. *Draw* adalah komponen yang bertugas menggambar semua objek ke layar. Sedang, *input* adalah sebuah masukan pengguna yang berpengaruh terhadap siklus update game.

**Figure 5. Siklus Sederhana Game**



1. *Input*  
Input komponen yang menerima masukan dari pemain. Input ini dapat berupa tombol, posisi atau yang lainnya yang berhubungan dengan game yang sedang dimainkan.
2. *MainLoop*  
Loop utama yang bertugas mengulang setiap logic dan fungsi gambar. MainLoop terdiri dari 2 buah fungsi penting.
  - a. *Update*  
Fungsi yang digunakan untuk mengupdate logic dari game. Update dapat berupa perpindahan posisi, pergerakan. Update lah yang menerima data dari inputan pengguna.
  - b. *Draw*  
Fungsi ini digunakan untuk menampilkan objek ke layar. Di dalam fungsi ini lah proses penggambaran pixel di lakukan.

Berikut pseudocode dari siklus game ini.

```
while (true) {
    Input();
    Update();
    Draw();
}
```

### V. TEKNIK WATERMARKING PADA GAME

Pengimplementasian watermarking pada game tidaklah serumit yang dibayangkan dari segi teori. Namun, mungkin juga tidak sesulit pengimplementasiannya. Sebenarnya ada dua bagian yang bisa disisipkan watermark pada game. Bagian tersebut antara lain , gambar dan suara. Namun, dalam makalah ini hanya membahas watermark pada citra game.

Berdasarkan jenis *watermark* ada dua teknik untuk watermarking pada game, antara lain *visible watermark* dan *invisible watermark*.

#### 1. Visible Watermark

Teknik ini kurang cocok untuk game. Hal ini disebabkan oleh game itu sendiri. Game bersifat interaktif dengan pengguna. Citra/Gambar merupakan komponen penting yang menghubungkan game dengan pengguna. Jika *watermark* terlihat pada game, maka kemungkinan besar pengguna/pemain jelas akan terganggu dengan adanya *watermark* tersebut.

Namun, jika dilihat dari segi pengimplementasian, *watermark* ini sangat mudah untuk diimplementasikan dan tidak mengubah siklus dari game itu sendiri.

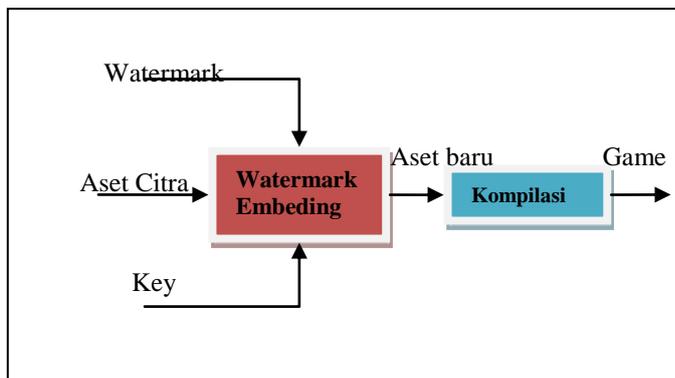
#### 2. Invisible Watermark

Teknik ini sangat cocok untuk game. *Watermark* yang tersembunyi jelas tidak akan mengganggu pemain yang memainkan game. Dari segi pengimplementasiannya ada dua cara yang dapat digunakan menurut teknik penyisipan.

##### a) Teknik proses penyisipan di luar proses game.

Ide dari teknik ini adalah proses penyisipannya. Proses penyisipan *watermark* tidak dilakukan di dalam game. Melainkan dilakukan di luar game. Penyisipan dilakukan kepada aset-aset citra sebelum

Figure 6. Teknik penyisipan di luar proses Game



Pada implementasinya, teknik ini tidak mengubah sama sekali siklus game. Sehingga tidak berpengaruh sama sekali terhadap performa game.

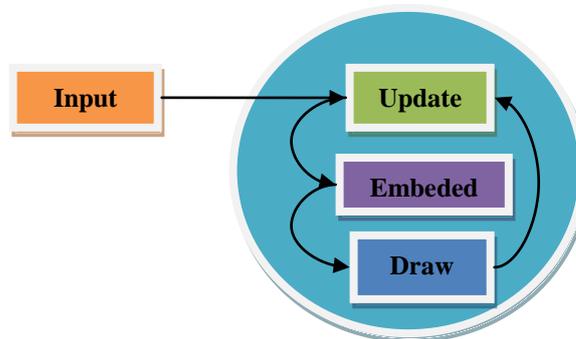
Jika game memiliki layar dimana tidak menggunakan aset citra maka layar tersebut tidak memiliki watermark karena watermark hanya ada di aset citra.

Pengembangan game 3D tidak disarankan menggunakan teknik ini. Jika teknik ini diimplementasikan pada game 3D maka ekstraksi watermark nantinya akan tidak sesuai harapan. Dikarenakan aset citra pada game 3D merupakan tekstur.

##### b) Teknik penyisipan di dalam proses game

Ide dari teknik ini adalah siklus dalam game diselipkan proses penyisipan *watermark*. Dengan kata lain watermark merupakan salah satu bagian dari aset citra game. Jadi pengembang tidak perlu membuat program terpisah untuk memberi *watermark* pada game mereka.

Figure 7. Siklus game setelah pemasukan proses penyisipan watermark



Pada siklus game bertambah satu fungsi diantara fungsi update dan draw (figure 7). Fungsi ini digunakan untuk memberi watermark pada buffer layar sebelum ditampilkan ke layar. Jika sebelumnya fungsi update kemudian diteruskan ke fungsi draw, maka pada siklus yang baru fungsi update dilanjutkan ke fungsi *embedded* lalu dilanjutkan ke fungsi draw. Harga dari fungsi *embedded* ini ditentukan oleh metode yang

digunakan untuk melakukan *watermarking* dan *buffer* (layar).

```
while(true) {
    Input();
    Embeded();
    Update();
    Draw();
}
```

Karena proses penyisipan berada dalam siklus game, maka kemungkinan teknik ini akan berpengaruh pada performa game. Pengaruh ini juga dipengaruhi oleh ukuran *pixel* layar. Semakin besar ukuran layar semakin mahal juga proses penyisipannya.

Pada teknik ini, tidak ada layar yang tidak disisipkan oleh *watermark*.

Untuk kedua teknik diatas, teknik ekstraksi tetap sama yaitu harus terpisah dengan proses game. Bagian yang diekstraksi adalah screenshot dari game. Screenshot bisa diambil dari layar mana pun.

Metode penyisipan watermark yang akan dibahas mirip dengan metode penyisipan dan ekstraksi pada citra. Metode yang akan dibahas disini adalah metode LSB dan metode Spread Spectrum.

1. Metode LSB

LSB (*Less Significant Bit*) merupakan bit-bit terakhir di dalam sebuah byte. Perubahan LSB pada warna tidak menunjukkan perubahan yang signifikan bahkan tidak bisa diketahui langsung dengan mata. Hal ini lah yang dimanfaatkan untuk menyembunyikan sebuah pesan atau *watermark*.

11010101  
                  |  
                  LSB

Perubahan pada warna tidak terlalu signifikan jika mengubah LSB nya. Warna sendiri mempunyai 3 segment masing-masing 1 byte. Segment tersebut merepresentasikan 3 warna penting, yaitu merah, hijau dan biru (RGB). Jadi dalam satu warna bisa menyimpan 3 bit.

Pada game, metode paling sederhana yang dapat digunakan adalah menyimpan sebuah gambar yang dijadikan sebagai watermark secara berurutan di dalam setiap pixel layar. Dapat juga digunakan dalam satu pixel terdapat satu pixel citra watermark. Jika dalam satu *pixel* dapat menampung 3 bit, maka representasi warna dalam citra watermark harus direpresentasikan ke dalam 3 bit (R = 1 bit, G = 1 bit, B = 1 bit). Dengan kata

lain, setiap segmen memiliki hanya 2 opsi, tampilkan atau tidak ditampilkan. Jumlah warna yang bisa digunakan adalah  $2^n = 2^3 = 8$  warna. Warna-warna yang diizinkan untuk implementasi ini adalah :

Representasi Warna	Nama Warna
000	Hitam
001	Biru
010	Hijau
011	Biru cerah
100	Merah
101	Merah jambu
110	Kuning
111	Putih

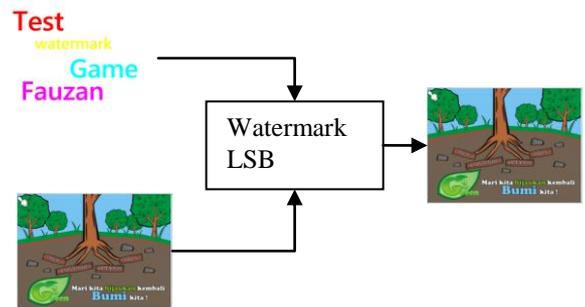
Kelebihan dari metode ini adalah :

- a. Mudah diimplementasikan.
- b. Dapat digunakan pada proses penyisipan yang embeded pada game atau yang tidak.
- c. Harga tidak terlalu mahal untuk proses.

Kelemahan dari metode ini adalah

- a. Tidak aman karena polanya yang berurutan.
- b. Robustnes : Artinya tahan terhadap perubahan.
- c. Mudah dihapus dengan mengganti bit LSB.

Figure 8. Metode dengan LSB



2. Metode Spread Spectrum

Metode *spread spectrum* merupakan metode yang ditawarkan pertama kali oleh Cox dalam sebuah makalah. Idennya adalah menyebar watermark ke dalam sebuah citra. Metode ini dapat dilakukan dalam 2 ranah, yaitu ranah spasial dan ranah transform.

Pada ranah spasial, metode ini menyisipkan watermark langsung pada nilai byte pixel. Sedangkan pada ranah transform, metode ini menyisipkan watermark pada koefisien transformasi dari citra. Penyisipan dalam ranah transformasi lebih robust dibandingkan dalam ranah spasial.

Pada ranah frekuensi, sebelum watermark disisipkan Citra ditransformasi ke dalam ranah frekuensi dengan DCT (*Discrete Cosine Transform*). Setelah penyisipan, ranah frekuensi dikembalikan ke ranah spasial (citra) dengan IDCT (*Inverse Discrete Cosine Transform*).

Figure 9. Proses penyisipan Spread Spectrum

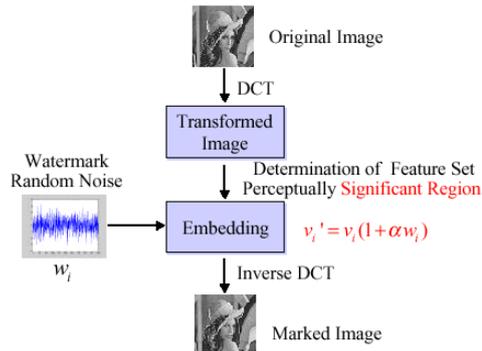
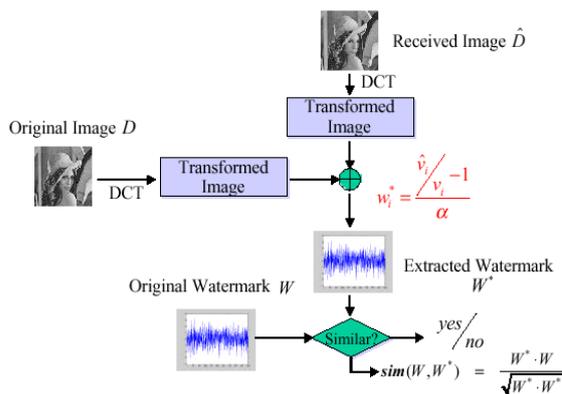


Figure 10. Proses Ekstraksi Spread Spectrum



Watermark pada metode ini merupakan bilangan riil acak (pseudo-noise) yang memiliki distribusi normal :

$$p(w) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{w^2}{2\sigma^2}\right)$$

Kelebihan dari metode ini :

- Hasil lebih *robust*.
- Lebih aman, tidak gampang dideteksi adanya watermark.
- Cocok untuk teknik penyisipan watermark pada asset game.

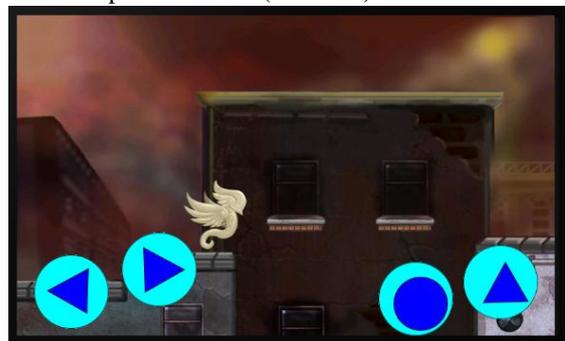
Kelemahan dari metode ini adalah tidak cocok untuk teknik penyisipan yang embed ke game karena biaya proses sangat mahal dan menggunakan banyak waktu. Selain itu proses ekstraksi yang memerlukan citra orisinil yang sulit didapat dari game karena game merupakan media yang dinamis. Namun, penggunaan teknik pada teknik penyisipan di luar proses game memungkinkan adanya bagian yang tidak ter-watermark. Hal ini harus dipikirkan ketika

mengembangkan game dengan menggunakan metode watermark ini.

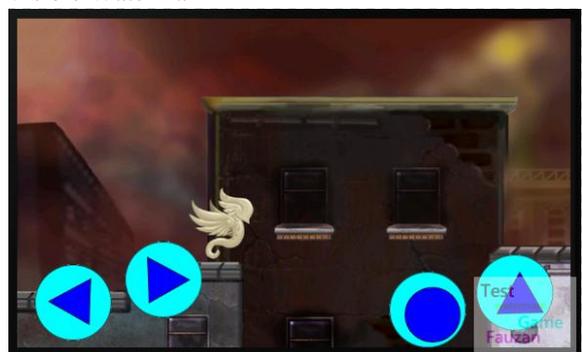
Jika dilihat dari metode dan teknik telah dibahas di atas. Metode dan teknik penyisipan adalah metode secure spread spectrum dengan teknik penyisipan di luar game (aset citra yang diberi watermark).

## VI. PENGUJIAN DAN ANALISIS

Game tanpa watermark (FPS : 30)



Visible Watermark



Invisible Watermark

Pengujian dilakukan pada metode LSB

1. Teknik dengan embed fungsi terintegrasi (rata-rata FPS : 25).



Hasil Dekripsi :

# Test watermark Game Fauzan

## 2. Teknik watermarking aset (FPS : 30)



Hasil ekstraksi watermark



Huruf "T" menggambarkan setiap objek dari game dimana object tersebut merupakan hasil dari watermark setiap aset dalam game.

## VII. DAMPAK PADA GAME

Berikut dampak-dampak pengimplementasian watermark pada game.

1. Performa bisa menurun jika menggunakan proses *watermark* yang di-embed pada game. Sedang jika menggunakan proses watermarking pada asset, sama sekali tidak mempengaruhi performa dari game.
2. Dapat memberikan tanda hak cipta pada game sehingga dapat mengakui hak milik. Apalagi untuk game-game yang tidak cukup berkelas dan kurang banyak dikenal orang.
3. Jika terjadi pencurian aset game dapat dengan mudah mengusut perkara ini.

Ada beberapa hal yang perlu dipertimbangkan dalam melakukan watermarking pada game.

1. Performa game.

2. Peluang bisnis dan peluang terjadinya pembajakan.
3. Peluang terjadinya pengambilan aset atau pembajakan secara intelektual.

## VIII. PENYELESAIAN KASUS PEMBAJAKAN

Dalam menyelesaikan kasus pembajakan. Langkah-langkah seperti apa yang dapat digunakan untuk mengatasinya dengan memanfaatkan watermark pada game.

Jika pada watermark citra, watermark digunakan untuk beberapa kasus diantaranya pengidentifikasian pemilik, bukti kepemilikan, melacak transaksi, dan content authentication.

Kasus dalam watermark citra juga bisa terjadi pada game. Namun, pada game kasus-kasus di atas jarang terjadi.

1. Identifikasi kepemilikan  
Watermark membantu mengidentifikasi siapa pengembang asli dari game tersebut. Hal ini kemungkinan terjadi pada game-game yang cukup tidak terkenal dan game indie.
2. Bukti kepemilikan  
Memberikan bukti tentang kepemilikan sebuah game atau aset game yang diakui kepemilikan oleh orang lain.
3. Melacak transaksi  
Hal ini mungkin dilakukan jika pengembang memasarkan game nya di dua atau lebih tempat terpisah. Setiap tempat memiliki watermark berbeda.
4. Content Authentication  
Hal ini dilakukan ketika ada pembajak yang men-decompile dan memodifikasi game (seperti menambah aset, memodifikasi background dan sebagainya).

## IX. KESIMPULAN

1. Pengimplementasian *watermarking* pada game sangat mungkin untuk dilakukan.
2. Metode yang digunakan masih terbatas jika ingin menggunakan fungsi penyisipan terintegrasi di dalam game. Fungsi LSB yang sederhana untuk digunakan masih mahal jika digunakan pada game.
3. Teknik penyisipan pada aset merupakan solusi terbaik saat ini karena tidak mengurangi performa game.
4. Butuh implementasi metode penyisipan lain yang lebih robust. Implementasi *Spread Spectrum* sulit untuk dilakukan pada game. Namun, mungkin untuk dilakukan pada aset.

## REFERENSI

[http://www.tweakguides.com/Piracy\\_4.html](http://www.tweakguides.com/Piracy_4.html). Tanggal akses 24 Maret 2013  
Joseph de Weck and Marie Mawad, "Free Online Games Sink Pirates to Unlock Emerging-Market Growth," Bloomberg BusinessWeek, August 27, 2012.  
<http://us.battle.net/wow/en/forum/topic/6521203906>. Tanggal akses 20 Maret 2013  
Slide kuliah Rinaldi Munir IF3058 Kriptografi

## PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 29 April 2010



Ahmad Fauzan  
13510004