

PENYAMARAN PESAN PADA SCRIPT CASCADING STYLE SHEET DENGAN METODE “DON PETERDON”

Dody Dharma – NIM : 13505061

*Program Studi Teknik Informatika, Institut Teknologi Bandung
Jl. Ganesha 10, Bandung
E-mail : dody.dharma@yahoo.com*

Abstrak

Cascading Style Sheet merupakan sebuah bahasa yang dimanfaatkan untuk mendefinisikan aspek antarmuka (*rendering*) dari suatu *formatted document* seperti HTML dan dokumen XML. Sebuah Dokumen CSS berisi selektor-selektor yang merepresentasikan pola-pola yang bersesuaian dengan elemen-elemen yang ada pada suatu dokumen HTML serta merupakan salah satu teknologi yang dapat dimanfaatkan untuk menyeleksi *nodes* pada dokumen XML.

CSS berkorelasi dengan *formatted document* melalui selektor-selektor. Selektor-selektor pada CSS mampu mengendalikan berbagai aspek *rendering* dari setiap elemen pada dokumen HTML. Salah satunya adalah warna elemen. Pada CSS versi termutakhir, CSS3, selain *selector* terdapat properti baru CSS yang disebut *Pseudo-element*. Dua *Pseudo-element* yang dibahas pada makalah ini adalah “**::selection**” dan “**::-moz-selection**”. Kombinasi selektor-selektor *class* serta dua elemen ini, dapat dimanfaatkan untuk memanipulasi warna suatu karakter beserta *background*-nya sedemikian sehingga setiap karakter teks pada dokumen HTML dapat direpresentasikan sebagai 1 pixel gambar.

Dari karakteristik diatas dirancanglah suatu mekanisme steganografi bernama “DON PETERDON”. Langkah penyamaran suatu pesan dilakukan sebagai berikut: pesan rahasia disimpan dalam bentuk gambar, kemudian dipersiapkan suatu teks panjang (paragraf) yang akan dijadikan sebagai tempat penyamaran pesan (*maskText*), kemudian oleh fungsi *DonPeterdonImgToTeks()* men-generate suatu dokumen HTML yang berisi *maskText* serta sebuah dokumen CSS. Setiap karakter pada *maskText* merepresentasikan setiap *pixel* dari gambar yang mengandung pesan rahasia. Warna dari setiap karakter *maskText* dikendalikan oleh dokumen CSS melalui *class selector*. *Timing* untuk menampilkan kembali pesan rahasia dikendalikan oleh *pseudo-element* “**::selection**” dan “**::-moz-selection**”. Dokumen HTML dan CSS tersebut disimpan pada file terpisah, ibarat sebuah lubang kunci dan kunci yang *match*. Jika Dokumen HTML dan CSS ini telah ter-link (melalui *tag* HTML misalnya: `<link rel="stylesheet" type="text/css" href="document.css">`), maka pesan rahasia dapat dilihat kembali dengan membuka Dokumen HTML tersebut pada browser yang mendukung *rendering* CSS3 dengan cara melakukan operasi seleksi teks pada dokumen HTML tersebut. Bahkan tingkat keamanan dalam proses menampilkan pesan dapat ditingkatkan dengan memanfaatkan *javascript* atau bahasa *server side script* sesuai kreativitas para pengguna metode “DONPETERDON” ini.

Kata kunci: Don Peterdon, *pseudo-element*, *class-selector*, *rendering*, *maskText*, *selector*, **::selection** dan **::-moz-selection**.

1. Pendahuluan

Pertukaran informasi di internet telah menjadi bagian yang krusial dalam perkembangan teknologi informasi di seluruh dunia. Namun, media internet bukanlah media yang aman untuk pertukaran informasi. Seseorang bisa saja menyadap aliran data pengiriman email penting atau bahkan mengganggu transaksi

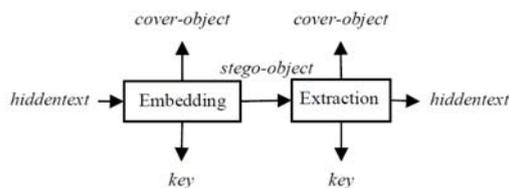
online. Oleh karena itu, digunakanlah teknik kriptografi. Sesuai perkembangan jaman, teknik kriptografi klasik berubah menjadi teknik kriptografi modern yang beroperasi pada satuan bit atau byte. Namun dengan kriptografi, walaupun makna suatu pesan menjadi hilang atau kacau, keberadaan akan pesan rahasia tersebut masih dapat diidentifikasi karena adanya keganjilan pada

pesan yang dikirimkan. Hal tersebut membuat penyadap informasi menyadari adanya pesan rahasia yang disembunyikan dan mengundang kriptanalis untuk melakukan kriptanalisis.

Kontras dengan penyamaran pesan atau steganografi, steganografi merupakan salah satu metode yang dapat dipergunakan dalam mengamankan akses terhadap suatu pesan. Penyembunyian atau penyamaran pesan ini dibuat sedemikian rupa sehingga pihak lain tidak mengetahui bahwa ada 'pesan lain' didalam pesan yang dikirimkan. Hanya pihak penerima yang sah saja yang dapat mengetahui 'pesan lain' tersebut.

Berbeda dengan kriptografi, dimana karakter pesan diubah/diacak menjadi bentuk lain yang tidak bermakna, dalam steganografi pesannya itu sendiri tetap dipertahankan hanya dalam penyampaiannya dikaburkan/disembunyikan dengan berbagai cara.

Pesan yang disampaikan secara kriptografi menjadi mencurigakan karena ke-'tidak bermakna'-annya tersebut. Sedangkan pesan dalam steganografi, terlihat seperti pesan biasa sehingga kecil kemungkinan untuk dicurigai. Langkah-langkah steganografi secara umum adalah sebagai berikut:



Gambar 1. Langkah Steganografi secara umum

Salah satu wadah yang dapat dipergunakan untuk menyamaran suatu pesan adalah *formatted document* (HTML atau XML) dengan memanfaatkan *script Cascading Style Sheet* (CSS) sebagai manipulator.

Cascading Style Sheet merupakan sebuah bahasa yang dimanfaatkan untuk mendefinisikan aspek antarmuka (*rendering layout* dan *style*) dari suatu *formatted document* seperti HTML dan dokumen XML. Sebuah Dokumen CSS berisi selektor-selektor yang merepresentasikan pola-pola yang bersesuaian dengan elemen-elemen yang ada pada suatu dokumen HTML serta merupakan salah satu teknologi yang dapat dimanfaatkan untuk menyeleksi *nodes* pada dokumen XML.

Selain dapat ditanamkan secara langsung pada file HTML atau XML, Cascading Style Sheet biasanya juga dapat berbentuk sebagai file yang berdiri sendiri dengan ekstensi ".css" yang berkorelasi melalui suatu *string tag connector* dengan file HTML atau XML tertentu.

Konvensi pada *script* CSS didefinisikan oleh suatu lembaga yang dikenal sebagai W3C (*World Wide Web Concorcium*) yaitu suatu organisasi yang didedikasikan untuk merancang spesifikasi pada *web*. Dalam perkembangannya CSS telah melalui beberapa fase pengembangan. Kini *script* CSS telah memasuki versi ke-3 yang dikenal sebagai CSS3.

CSS berkorelasi dengan *formatted document* melalui selektor-selektor. Selektor-selektor pada CSS mampu mengendalikan berbagai aspek *rendering* dari setiap elemen pada dokumen HTML. Pada CSS versi termutakhir, CSS3, selain *selector* terdapat properti baru CSS yang disebut *Pseudo-element* atau *Pseudo-class*.

Definisi dari *pseudo element* berdasarkan dokumen spesifikasi *Selector Level 3* oleh W3 (<http://www.w3.org/TR/2009/WD-css3-selectors-20090310>) adalah sebagai berikut:

The pseudo-class concept is introduced to permit selection based on information that lies outside of the document tree or that cannot be expressed using the other simple selectors.

Dua *Pseudo-element* yang dibahas pada makalah ini adalah "**::selection**" dan "**::-moz-selection**". Kombinasi selektor-selektor *class* serta dua elemen ini, dapat dimanfaatkan untuk memanipulasi warna suatu karakter beserta *background*-nya sedemikian sehingga setiap karakter teks pada dokumen HTML dapat direpresentasikan sebagai 1 pixel gambar.

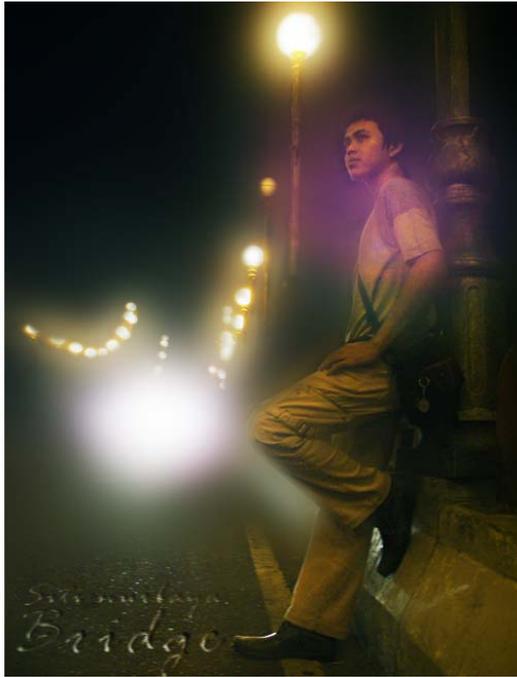
2. Teknik Steganografi Don Peterdon

Ide dari Teknik steganografi Don Peterdon sebenarnya sederhana mudah dipahami namun sangat menarik. Steganografi Don Peterdon mempergunakan karakteristik *raster image*, notasi akses CSS *class selector* dan karakteristik *pseudo-class* dari CSS3. Aspek dan karakteristik yang menjadi landasan steganografi Don Peterdon akan dijelaskan sebagai berikut.

2.1 Karakteristik Elemen-elemen

2.1.1 Raster Image

Raster Image merupakan gambar yang tersusun atas kotak-kotak yang disebut *pixel*. Setiap pixel pada gambar dapat menyimpan satu jenis warna. Sehingga dapat dikatakan sebuah gambar adalah kumpulan kotak-kotak yang berwarna-warni membentuk pola tertentu. Hal ini dapat diamati dengan melakukan perbesaran terhadap suatu *raster image*.



Gambar 2. Sebuah Raster Image



Gambar 3. raster image pada Gambar.2 diperbesar 10 kali pada bagian wajah

Terdapat berbagai format *file raster image*. Diantara Format yang cukup populer adalah JPEG, PNG, GIF, dan BITMAP. Yang membedakan format-format file ini satu sama lain adalah metode kompresi serta kedalaman warna. Kompresi dan kedalaman warna ini menentukan kualitas dan ukuran suatu gambar. Sebagai contoh, format JPEG memiliki kedalaman warna hingga 32 bit. Artinya suatu

gambar JPEG dapat mengandung hingga 2^{32} ($2^{32} = 4.294.967.296$) jenis warna. Sehingga kualitas gambar yang dapat ditampilkan oleh format gambar JPEG adalah setara dengan kualitas foto.

Pada makalah ini, penulis membatasi masalah dengan merancang Algoritma steganografi yang dapat menghasilkan gambar dengan kedalaman warna **RGB (256 x 256 x 256 warna)**.

2.1.2 CSS Class Selector

Pada CSS konvensi CSS terdapat tiga jenis Selector yaitu ID, *Class*, dan *tag*. Peran ketiga jenis selektor ini dijelaskan sebagai berikut:

1. ID : berfungsi sebagai selektor terhadap salah satu elemen dari *tree* pada suatu *formatted document*. Pada sebuah *formatted document* ID harus bersifat unik. Artinya *selector* berjenis ID tidak boleh dipergunakan lebih dari **satu kali**.

```
<body>
<div ice:repeatinggroup="*">
  <div id="header" ice:repeating="true">
    <div id="animation-box">
```

Gambar 4. id="header" pada ilustrasi diatas hanya boleh muncul satu kali pada suatu *formatted document*.

2. Class : Class berperan sebagai *selector* terhadap berbagai elemen dari *tree* pada suatu *formatted document*. Pada sebuah *formatted document*. *Class* tidak bersifat unik. Artinya *Class*s dapat dipergunakan pada berbagai objek sekaligus. Misalnya: beberapa paragraf jika mempergunakan *class* yang sama, maka semua paragraf tersebut akan memiliki *style* yang sama pula.

```
<li>
  <div class="container-image-news">
    <div class="image-slide">
    <div class="caption">
      <h3 class="caption-title"><a href="slide.html" class="caption-link">
      <p class="caption-excerpt">Acara Arkavidia telah dibuka
      rak-Arakan Massa himpunan Mahasiswa Teknik Informatika ITB di c
```

Gambar 5 Contoh penggunaan Class

3. Tag : merupakan *selector* yang secara langsung mendefinisikan *style* berdasarkan jenis tag: misalnya pada html untuk mendefinisikan badan halaman dipergunakan tag

```
<body></body>
```

Maka pada file CSS cukup dipergunakan notasi selektor dengan ilustrasi sebagai berikut

```
<style type="text/css"
media="screen">
<!--
body {
background-color: #808080;
margin-left: 0px;
margin-top: 0px;
margin-right: 0px;
margin-bottom: 0px;
background-attachment: scroll;
background-repeat: no-repeat;
background-position: center 0px;
background-image:
url(images/background-side.jpg);
}
```

Pada makalah ini jenis *selector* yang dipergunakan adalah *class*. Karena kelak pada steganografi i *class* akan dipergunakan untuk mendefinisikan warna dari berbagai karakter secara sekaligus, sehingga dibutuhkan *selector* yang dapat dipergunakan lebih dari satu kali.

2.1.3 Pseudo-class

Berdasarkan W3C pseudo class didefinisikan sebagai berikut:

The pseudo-class concept is introduced to permit selection based on information that lies outside of the document tree or that cannot be expressed using the other simple selectors.

A pseudo-class always consists of a "colon" (:) followed by the name of the pseudo-class and optionally by a value between parentheses.

Pseudo-classes are allowed in all sequences of simple selectors contained in a selector. Pseudo-classes are allowed anywhere in sequences of simple selectors, after the leading type selector or universal selector (possibly omitted). Pseudo-class names are case-insensitive. Some pseudo-classes are mutually exclusive, while others can be applied simultaneously to the same element. Pseudo-classes may be dynamic, in the sense that an element may acquire or lose a pseudo-class while a user interacts with the document.

Pada Makalah ini *::pseudo-class* yang dimanfaatkan adalah "*::selection*" dan *::-moz-selection*.

2.2 Langkah Merepresentasikan karakter alfanumerik sebagai pixel gambar.

Untuk merepresentasikan setiap karakter alfanumerik sebagai pixel, penulis mempergunakan tag `` beserta

class selector yang dipergunakan untuk mengendalikan warna *font* beserta warna *background* dari *font* tersebut. Agar terlihat sebagai sebuah kota, maka warna *font* dan *background*-nya dibuat homogen. Sebagai ilustrasi, penulis hendak merepresentasikan kata "XML" sebagai elemen-elemen pixel yang terdiri dari 3 pixel, yaitu pixel dari karakter "X", karakter "M", dan karakter "L", maka penulis mendefinisikan ketiga karakter tersebut pada dokumen html sebagai berikut:

```
<span class='color-2'>X</span>
<span class='color-1'>M</span>
<span class='color-0'>L.</span>
```

Kemudian pada dokumen CSS, penulis mendefinisikan *class* color-1, color-2, dan color-3 dengan kombinasi nilai-nilai RGB padesebagai berikut:

```
.color-0::selection {
color: rgb(252, 253, 251);
background: rgb(252, 253, 251)
}
.color-0::-moz-selection {
color: rgb(252, 253, 251);
background: rgb(252, 253, 251)
}
.color-1::selection {
color: rgb(252, 246, 249);
background: rgb(252, 246, 249)
}
.color-1::-moz-selection {
color: rgb(252, 246, 249);
background: rgb(252, 246, 249)
}
.color-2::selection {
color: rgb(243, 244, 246);
background: rgb(243, 244, 246)
}
.color-2::-moz-selection {
color: rgb(243, 244, 246);
background: rgb(243, 244, 246)
}
```

Dengan teknik sederhana ini, maka ketika seorang user melakukan *selection* dengan mouse cursor pada text "XML" pada

dokumen HTML nya maka text yang semula tampil dalam bentuk seperti berikut:

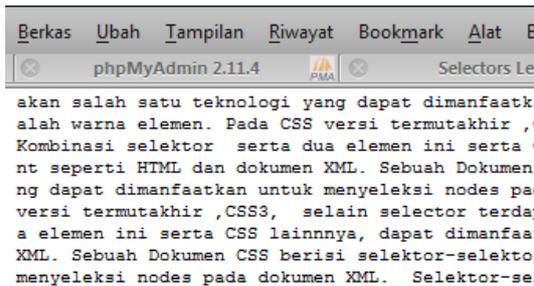
XML
Menjadi



Dengan teknik ini, maka penulis dapat merepresentasikan pixel-pixel pada gambar dengan mempergunakan setiap karakter.

Berikut adalah ilustrasi bagaimana penulis menyembunyikan fotonya sendiri dalam abstrak makalah ini, penulis membuat sebuah dokumen HTML berisi bastrack dengan mendefinisikan setiap karakter pada dokumen HTML tersebut didefinisikan:

```
<span class='color-0'>g merepresentasikan pola-po</span><span class='color-1'>la</span><span class='color-0'> yang bersesuaian dengan elemen- </span><span class='color-1'>ele</span><span class='color-0'>men yang ada pada suatu dokumen HTML</span><span class='color-1'>L </span><span class='color-0'>serta merupakan salah satu teknologi yang dapat dimanfaatkan untuk menyeleksi node</span>.....
```



Gambar 6 Ilustrasi Abstrak Makalah pada HTML



Gambar 7. Ilustrai makalah pada tampilan browser Mozilla Firefox



Gambar 8 Ilustrasi Gambar yang muncul ketika penulis melakukan seleksi terhadap text dari Abstrak dalam dokumen HTML.

2.3 Algoritma Don Peterdon

Adalah tidak mangkus dan sangkil jika penulis mendefinisikan setiap karakter pada dokumen dengan mengetikkan tag-tag HTML secara satu persatu, karena hal ini akan sangat menyita waktu. Untuk itu penulis merancang sebuah fungsi dengan mempergunakan script PHP yang menerima parameter berupa: *mask_text*, *direktori_gambar*, *kedalaman_warna*, *_output*, dan *lebar_paragraf* sebagai representasi lebar gambar. Berikut algoritma Don Peterdon secara keseluruhan:

1. *Resize* gambar dengan *constraint width* sesuai nilai parameter *lebar_paragraf*
2. Baca pixel gambar satu persatu dari awal sampai akhir.
3. Setiap kali pemacaan pixel gambar, akuisisi informasi warna pixel tersebut dalam representasi kombinasi nilai-nilai Red, Green, dan Blue (RGB).
4. Selama iterasi pembacaan pixel, *generate* tag `` "karakter yang dibaca" `` Dan definis class pada CSS dengan pseudo-class `::moz-selection` dan `::selection`.
5. Generate File HTML dengan tag linker terhadap CSS class.
6. Generate file CSS.

3. Hasil dan Pembahasan

Agar Image pada gambar itu tersebut tidak dapat dilihat oleh yang tidak berhak maka, pihak yang terkait harus memisahkan file HTML dan CSS nya pada tempat yang terpisah. Selain itu pihak yang berkepentingan juga dapat mengubah-ngubah pendefinisian "href" dari linker CSS pada file HTML agar pihak yang tidak berhak tidak mengetahui style kunci yang dapat menginterpretasi warna dari dokumen tersebut.

Jika pihak yang memiliki authority, hendak melihat pesan tersembunyi pada dokumen HTML tersebut, maka ia cukup mengkonfigurasi linker style dari dokumen HTML tersebut ke alama yang sesuai terhadap letak file CSS kunci yang dapat

menginterpretasi warna pada text dokumen tersebut tersebut.

Algoritma yang penulis rancang pada makalah hanyalah prototipe fondasi dari metode steganografi Don Peterdon. Karena untuk mencapai tingkat keamanan yang lebih baik, maka algoritma ini harus dikolaborasikan dengan algoritma enkripsi lain. Serta metode-metode kreatif dari pihak yang hendak memanfaatkan algoritma ini.

Selain itu karena dokumen yang dihasilkan berbentuk HTML serta mempergunakan mempergunakan jenis script CSS3 , maka kesuksesan hasil proses steganografi ini masih tergantung terhadap Browser yang dipergunakan. Browser yang dimanfaatkan harus mendukung CSS3. Saat ini yang penulis ketahui, browser yang mendukung CSS3 diantaranya adalah Mozilla Firefox, Safari, dan Opera. Intenet Explorer (IE) belum mendukung interpretasi CSS3, sehingga metode ini tidak akan berhasil pada IE.

Agar integritas Don Peterdon ini dapat ditingkatkan, Penulis berencana mengembangkan mekanisme yang lebih baik dalam upaya meningkatkan tingkat sekuritas dari Mekanisme Don Peterdon ini.

4. Simpulan

Berikut adalah simpulan yang dapat ditarik dari pemaparan pada makalan ini:

1. Algoritma Don Peterdon ini cukup menarik untuk dimanfaatkan sebagai salah alternatif mekanisme penyembunyian pesan berbentuk gambar dalam teks. Bahkan Algoritma Don Peterdon juga dapat dimanfaatkan untuk tujuan-tujuan lain. Misalnya menyembunyikan *text* pada suatu *image*, dengan meng-*invert* pendefinisian `::moz-selection` dan `::selection` pada file script CSS dari dokumen HTML yang berperan sebagai *mask_text*.
2. Untuk peningkatan keamanan, maka Algortma ini dapat dikolaborasikan dengan berbagai algortima enkripsi pesan yang lain.
3. Metode steganografi ini bersifat browser dependent. Karena interpretasi CSS3 masih belum sepenuhnya didukung oleh semua jenis browse.

5. Daftar Pustaka

- [1] Duckett, Jon. (2005). Accessible XHTML and CSS Web Sites – Problem Design Solution. Wiley Publishing, Inc
- [2] <http://www.w3.org/TR/2009/WD-css3-selectors-20090310>.