

Laporan Tugas Akhir Kuliah
EC 7010 Keamanan Sistem Lanjut
Dosen: Dr. Ir. Budi Rahardjo

SISTEM KEAMANAN MACROMEDIA FLASH MX

Oleh :

NASIR SURUALI
NIM : 23203122



PROGRAM PASCA SARJANA
DIDANG KHUSUS TEKNOLOGI INFORMASI-DIKEMNJR
INSTITUT TEKNOLOGI BANDUNG
2004

Daftar Isi

halaman

Daftar Isi	2
Abstrak.....	3
I. Pendahuluan.....	4
1.1 Latar Belakang.....	4
1.2 Tujuan.....	4
1.3 Lingkup Masalah.....	5
II. Sistem Keamanan Macromedia Flash MX.....	5
2.1 Pengertian.....	5
2.2 Mengapa Mengamankan Macromedia Flash MX.....	5
2.3 Bagaimana Cara Macromedia Flash MX melindungi Privacy User.....	6
2.4 SWF dan Ostriches	6
2.5 Encryption data dengan SSL.....	7
2.6 Crypto Bulevard (Keamanan untuk satu arah).....	8
2.7 Encryption data searah dengan menggunakan MD5.....	9
2.8 Keamanan Tansfer data dari Macromedia Flash Projector.....	9
2.9 Keamanan dari Format Teknologi Terbuka (Open format technology).....	10
2.10 Pencegahan Pengamanan data dalam Macromedia Flash movie.....	10
III. Hubungan Keamanan terhadap Virus atau Trojan.....	11
3.1 Playback Macromedia Flash melalui projector.....	11
3.1.1 Macromedia Flash Projector Carrier dari Virus.....	12
3.1.2 Penyamaran file Malicious sebagai file Macromedia Flash.....	12
3.1.3 Isi Macromedia Flash dari Code Malicious.....	12
3.2 Macromedia Flash Attachments E-mail.....	13
IV. Macromedia Flash Player 6 Sandbox.....	13
4.1 Apa itu sandbox.....	13
4.2 Authenticasi Domain-based.....	14
4.3 Bagaimana mengimplementasikan Sandbox.....	15
4.3.1 Mengakses I/O file.....	15
4.3.2 Komunikasi Cross-movie.....	17
V. Komunikasi Flash-JavaScript.....	19
5.1 Liveconnect API.....	20
5.2 Control ActiveX API.....	21
VI. Hubungan Macromedia Flash dengan Keamanan Lainnya.....	23
VII. Penutup.....	23
Literatur.....	24

SISTEM KEAMANAN MACROMEDIA FLASH MX

Oleh :
Nasir Suruali
E-mail : nasir_si2003@yahoo.com

Abstrak

Dalam dunia animasi Web, teknologi Macromedia Flash MX kini seolah meraja. Keunggulan-keunggulan teknologi ini telah membuat kemajuan yang sangat berarti dalam perkembangan dunia multimedia melalui internet. Dengan teknologi ini, kini kita tidak hanya melihat sebuah situs lebih menarik, dinamis, tetapi juga kecepatan.

Seiring dengan perkembangannya, maka keamanan penggunaan Macromedia Flash harus dijaga karena tidak hanya digunakan untuk membuat animasi di Web, bahkan dengan dukungan Action Script yang makin canggih, Flash banyak pula digunakan untuk membuat cd presentasi, cd interaktif dan games atau permainan.

Dukungan terhadap aplikasi ini pun kian marak, tidak hanya dukungan dari internet, yaitu browser tetapi juga berasal dari perusahaan-perusahaan di luar Macromedia untuk membuat program-program pelengkap yang biasa disebut “third party” yang dapat digunakan secara bersama-sama atau terpisah dengan program Macromedia Flash MX.

Dalam tulisan ini, penekanan isi terarah pada pengamanan Macromedia Flash terhadap privacy user, keamanan satu arah, keamanan transfer data, pencegahan keamanan data serta hubungan keamanan terhadap virus atau trojan.

Kata kunci : Macromedia Flash MX, Keamanan, Internet, Action Script, Privacy, Virus atau Trojan.

I. PENDAHULUAN

1.1. Latar Belakang

Macromedia flash MX merupakan sebuah program aplikasi standar *authoring tool* professional yang digunakan untuk membuat animasi dan *bitmap* yang sangat menarik untuk keperluan pembangunan situs web yang interaktif dan dinamis. Selain itu aplikasi ini juga dapat digunakan untuk membuat animasi logo, *movie*, *game*, pembuatan *navigasi* pada *situs web*, tombol animasi, *banner*, *menu interaktif*, *interaktif form isian*, *e-card*, *screen saver* dan pembuatan aplikasi-aplikasi web lainnya.

Movie-movie Flash MX memiliki ukuran file yang kecil dan dapat ditampilkan dengan ukuran layar yang dapat disesuaikan dengan keinginan. Aplikasi Flash MX merupakan sebuah standar aplikasi industri perancangan animasi web dengan peningkatan pengaturan dan perluasan kemampuan integrasi yang lebih baik.

Banyak fitur-fitur baru dalam Flash MX yang dapat meningkatkan kreativitas dalam pembuatan isi media yang kaya dengan memanfaatkan kemampuan aplikasi tersebut secara maksimal. Fitur-fitur baru ini membantu kita lebih memusatkan perhatian pada desain yang dibuat secara cepat, bukannya memusatkan pada cara kerja dan penggunaan aplikasi tersebut. Flash MX juga dapat digunakan untuk mengembangkan secara cepat aplikasi-aplikasi web yang kaya dengan pembuatan *script* tingkat lanjut. Di dalam aplikasinya juga tersedia sebuah alat untuk *men-debug script*. Dengan menggunakan *Code hint* untuk mempermudah dan mempercepat pembuatan dan pengembangan isi *ActionScript* secara otomatis.

Makin banyak orang yang tertarik dengan aplikasi Macromedia Flash MX sehingga menimbulkan suatu pertanyaan tentang keamanan pada Macromedia Flash. Dimana aplikasi Macromedia Flash mempunyai tingkat keamanan yang didesain dengan menggunakan aplikasi dasar HTML.

1.2 Tujuan

Tujuan dari penulisan tugas ini adalah untuk mempelajari pengembangan dan implementasi dari keamanan Macromedia Flash MX.

1.3 Lingkup Masalah

Keamanan Macromedia Flash MX adalah untuk melakukan dan memastikan bahwa Macromedia Flash adalah suatu keamanan dan *privacy*. Penulisan tugas ini hanya membahas tentang keamanan dan *privacy* dari isi Macromedia Flash yang meliputi sejumlah isu keamanan bila dihubungkan dengan isi *player* dari sistem file yang dimulai dari sumber yang tidak diketahui atau tidak dapat dipercaya kebenarannya.

II. Sistem Keamanan Macromedia Flash MX

2.1 Pengertian

Macromedia Flash Player adalah suatu perangkat lunak *bundled* dari Macromedia *client* yang banyak digunakan dalam sejarah internet dengan *Internet Explorer*, *AOL*, *Netscape Navigator*, *Opera*, dan *Windows XP*. Macromedia Flash Player juga tersedia dengan pertumbuhan sejumlah internet dengan sarana penghubung seperti *wireless handhelds*, *iTV* dan *game consoles*.

Macromedia Flash MX juga merupakan salah satu solusi teknologi *client* yang ada dimana Macromedia Flash Player dengan Macromedia Flash MX masih dalam tahap pengembangan yang telah dioptimasi *server-side conectivitasnya* serta mengizinkan *developer* untuk memenuhi aplikasinya di internet. Ini membutuhkan suatu pengalaman, konsistensi dan familiar agar dikenal oleh umum dan memaksimalkan penggunaannya oleh pelanggan secara online.

2.2 Mengapa mengamankan Macromedia Flash MX

Macromedia Flash Player mempunyai *list cek* yang luas, Pembatasan dan *featurenya* untuk memastikan bahwa isi dari Macromedia Flash terjamin keamanannya, hal ini meliputi :

- kemampuan untuk menggunakan *encryption* dari *browser* seperti SSL, dan semua komunikasi antara Macromedia Flash movie dan *server* ke *encrypt*,
- sistem keamanan dapat membatasi perpindahan (*transver*) informasi secara luas yang mungkin berisiko bagi keamanan atau *privacy*,
- macromedia flash player tidak mengizinkan untuk membaca isi data web dari drive kecuali untuk *SharedObjects* yang telah diciptakan domainnya,

- macromedia flash player, disknya tidak dapat menulis data kecuali data yang telah di *encapsulasi* dalam *SharedObjects*,
- macromedia flash player tidak mengizinkan membaca isi data web dari suatu *server* yang tidak sama domainnya kecuali jika isinya diizinkan untuk diakses,
- macromedia flash player tidak mengizinkan untuk menempatkan isi web lebih dari 100K dalam disk dari satu domain,
- macromedia flash player memungkinkan pemakai (*user*) untuk menghilangkan domain tempat penyimpanan data informasi,
- macromedia flash player tidak mengizinkan data untuk dikirim melalui kamera atau mikropon kecuali jika *user* memberi ijin bagi seorang *user* domain.

2.3 Bagaimana cara Macromedia Flash MX melindungi *privacy user*

Macromedia Flash memahami pentingnya *privacy user* dan telah berpengalaman dalam memastikan bahwa isi Macromedia Flash tidak mengganggu *privacy*, hal ini dapat terpenuhi dengan cara sebagai berikut :

- para user harus menyetujui akses ke web kamera dan mikropon,
- suatu domain tidak dapat mengakses data yang disimpan dari domain lain, ini adalah bagian dari pekerjaan *web browser*,
- Macromedia Flash tidak mempunyai akses ke informasi yang lebih pribadi, kecuali jika disajikan sendiri oleh *user*.

2.4 SWF dan *ostriches*

Untuk memahami keamanan Macromedia Flash dapat dilihat dari beberapa sudut pandang, berdasarkan beberapa sumber referensi bahwa tidak ada perbedaan menyolok antara HTML dan *JavaScript* dimana didalamnya terdapat banyak *tools* yang dapat diambil dari SWF termasuk *ActionScript*. Sehingga kode data dapat terjamin keamanannya.

Oleh sebab itu, semua kebutuhan data yang terdapat dalam SWF dapat diambil kembali melalui *server*. Keuntungan menggunakan metode yang sama dengan menggunakan aplikasi web yang standar adalah akan menjamin dan mengamankan penyimpanan dan perpindahan data.

2.5 Encryption data dengan SSL

Metoda standar untuk mengamankan koneksi data dalam web adalah dengan menempatkan jenis *web server* khusus di belakang *HTTP sever*. *HTTP server* ini adalah standar untuk menjamin dan mengamnkan dari sistem komunikasi *encrypting* dengan SSL.

Untuk mengirim data dari Macromedia Flash ke *server*, Macromedia Flash tidak secara langsung ke *server* tetapi melalui *browser*. Ini terjadi karena SWF terlebih dahulu mengontak *HTTP server* bahwa data akan di *encrypted* sebelum dikirim. Ini merupakan suatu pencegahan awal terhadap aplikasi potensial Macromedia Flash yang rahasia dan menempatkannya ke *HTTP server*. Cara ini tidak hanya mengamankan pengirim data ke *server* tetapi juga mengunci *icon* pada *browser*, hal ini menandakan bahwa aplikasi kita terjamin keamanannya.

Bermain di Macromedia Flash movie, mempunyai keamanan *browser* dalam halaman HTML hal ini dapat dilihat dari *browsersnya*. Dimana meliputi keamanan Macromedia Flash movie yang berada dalam *browser*, seperti keamanan komunikasi antara Macromedia Flash dengan *server* setelah *movie* dijalankan di *browser*. Khususnya komunikasi data antara *browser* dan *server* sangat rahasia bila *diinterupsi* pihak ketiga. Solusinya, pada HTML dengan meng-*encrypt* komunikasi antara *client* dan *server* dalam membuat data yang dapat di ambil oleh pihak ketiga, walaupun tidak dipakai dan di baca. Ini dapat dilakukan dengan menggunakan SSL yang dimungkinkan oleh *browser* dan *server*.

Sejak Macromedia Flash movie dijalankan dalam *browser* untuk semua komunikasi *server*, dapat diambil manfaatnya dari *support browser built-in SSL*. Hal ini dapat juga dilakukan dengan meng-*encrypsi* komunikasi antara *byte actual* dari Macromedia Flash movie yang terdapat dalam *browser*.

Jadi, bermain di Macromedia Flash movie dalam SSL melalui HTTP memungkinkan *browser* terkoneksi dengan *server*, juga dapat memastikan *encrypted* untuk menjamin atau mengamankan komunikasi antara Macromedia Flash Player dan *server*.

Ada satu perkecualian pada Macromedia Flash dalam menggunakan *persistent sockets* melalui *ActionScript XMLSocket objec*, dalam berkomunikasi dengan *server* tidak menggunakan *browser*. Oleh karena itu, untuk mengambil manfaat dari kemampuan *built-in encryption* adalah dari *browser*. Hal ini dapat dimungkinkan untuk menggunakan algoritma *encryption* searah yang di tulis dalam komunikasi data *ActionScript ke encrypt*.

2.6 Crypto Bulevard (Keamanan untuk satu arah)

Kadang-kadang untuk mendapatkan HTTP *server* sangat sulit. Situasi seperti ini kita dapat menyimpan dan mengamankan sebagian data kita dengan menggunakan fungsi *hash cryptographic*. Fungsi *hash crypto* adalah memberikan hasil masukan dari suatu keluaran yang tidak dapat digunakan untuk menentukan nilai masukan, misalnya fungsi *hash crypto* satu arah, untuk mendapatkan keluaran dari masukan tersebut kita tidak dapat menentukan masukannya dan tidak dapat melihat keluarannya juga.

Fungsi *hash Crypto* sangat bermanfaat ketika *user* mengetahui *passwordnya*, kita juga mengetahuinya sehingga harus mengamankan *password user* tersebut dan menyimpannya ke *server* dan tidak boleh di beritahukan kepada publik.

Cara menggunakan fungsi *hash crypto* sebagai berikut :

1. SWF kita harus dihubungkan ke *server* dan memintanya untuk di acak,
2. kemudian SWF menggabungkannya dengan memberi isyarat kepada seorang *user* dengan terlebih dahulu memasukan *password* dan memberikan hasil melalui algoritma *crypto hashnya* (kita sebut A),
3. sekarang SWF mengirim A ke *server*. *Server* kemudian akan menemukan nilai pengiriman tersebut dari database dan menggabungkannya dengan *password* para *user*. Kemudian *server* melau fungsi *hash crypto* dapat digunakan oleh *client* (kita sebut B),
4. jika A dan B bertemu, *server* mengetahuinya bahwa dari masing-masing *passwordnya* adalah benar adanya dan *password* itu tidak dikirim kesembarang tempat dan akan diamankan oleh SWF *server*.

Ada beberapa jenis algoritma yang dikenal, tetapi yang paling polpuler ada 2 algoritma yaitu MD5 dan SHA1.

SHA1 secara umum dianggap dapat mengamankan untuk 2 user sebab outputnya lebih panjang bila dibandingkan dengan MD5.

2.7 Encryption data searah dengan menggunakan MD5

Md5 adalah suatu algoritma *encryption* searah dan telah menjadi *ported* ke *ActionScript*, yang memungkinkan *developer* untuk mengamankan data searah dengan menggunakan algoritma MD5 ini dikirim melalui Macromedia Flash movie ke *server*.

Pembuatan algoritma MD5 untuk mengamankan *one-way* dari *string*. *String* ini tidak dapat meng-*unencrypted* ke dalam *string* originalnya, tetapi dapat dibandingkan terhadap perlawanan data lain yang menggunakan algoritma *encrypted* MD5. Hal ini tidak dilakukan untuk mencoba kemampuan *encrypt* dan *decrypt* data seperti pada SSL, manfaatnya adalah jika memindahkan data rahasia dari *browser*, tidak dapat dengan menggunakan *capabilitas* SSL, kasus ini meliputi :

- suatu movie dapat dijalankan di dalam *browser* dari ketidakmampuannya SSL,
- kita tidak mempunyai hak akses ke *SSL-capable web server*,
- berkomunikasi dengan *server* harus menggunakan *socket XML*,
- penggunaan Macromedia Flash projector hanya untuk menjalankan movie.

Proses encrypting data adalah sebagai berikut :

1. *encrypsi* data pada Macromedia Flash adalah searah dengan *one-way hash*
2. macromedia flash mengirim data ke *server*,
3. server menerima data,
4. *server* memvalidasi *one-way hash* terhadap *pre-existing hash*.

Algoritma md5 ditulis dalam *ActionScript*.

2.8 Keamanan Transfer data dari Macromedia Flash Projectors

Macromedia Flash *projector executables* dapat dijalankan di luar *browser*, dan dapat diambil manfaat *browsersnya* dari *capabilitas* SSL. Oleh sebab itu, jika mentransfer informasi yang rahasia antara *projector* dan *server*, kita harus :

- data kita di *encrypsi* dengan algoritma *ActionScript* seperti md5 yang telah disinggung diatas,
- user memerlukan suatu jaminan keamanan koneksi jaringan bagi *server* seperti koneksi jaringan pribadi *Virtual Private Network (VPN)* .

2.9 Keamanan dari Format Teknologi Terbuka (*open format technology*)

Seperti diketahui sebelumnya, hubungan Macromedia Flash movie dengan halaman web adalah gagal dalam melindungi data pada saat men-format file SWF, hal ini dapat saja terjadi penyadapan data dan algoritma dalam Macromedia Flash movie. Hal serupa, bagaimana kode HTML dan *JavaScript* mendeteksi atau memandang para *user*, maka dibuat kode Macromedia Flash movie yang lebih sulit untuk dideteksi, disadap dan diamati. Meng-*compile* file SWF tidak *human-readable* seperti HTML atau *JavaScript*

Tetapi keamanan tidak didapat melalui ketidakjelasan. telah banyak tulisan tentang cara menyadap data dari meng-*compile file* SWF. Paling populer terdapat pada *ActionScript Viewer (ASV)*, dimana *tools* ini tidak mensupport semua versi dari file SWF, olehnya itu kerja sama dibutuhkan untuk mensupport dalam penambahan file tersebut.

Pada intinya, data, variabel atau kode *ActionScript* yang di-*compile* dalam Macromedia Flash movie atau *projector* tidak mempertimbangkan faktor keamanannya.

2.10 Pencegahan pengamanan data dalam Macromedia Flash movie

Dalam meng-*compile* data dan algoritma dalam Macromedia Flash movie dapat saja disadap, bukan berarti bahwa kerahasiaan informasi tidak terlindungi. Ada sejumlah teknik yang dapat digunakan untuk mengamankan kerahasiaan informasi bagi pengguna Macromedia Flash movie.

1. rahasia informasi jangan di *hard-code*-kan seperti *username*, *password* atau *statements SQL* dalam Macromedia Flash movies,
2. jika anda terpaksa mengakses informasi rahasia pada Macromedia Flash movie, dimana informasi tersebut tersimpan dalam movie dari *server runtime*

maka bagian data di-*compile* dari file SWF dengan begitu tidak dapat disadap oleh *user* lain. Pastikan bahwa mekanisme penggunaan perpindahannya dapat terjamin keamanannya seperti penyimpanan data dalam SSL,

3. terapkan algoritma rahasia pada *server* sebagai ganti *ActionScript*,
4. sebarkan aplikasi web anda dari *trusted server*. Cara lainnya adalah adanya kesepakatan kita terhadap aspek *server-side*.

III. Hubungan Keamanan terhadap Virus atau Trojan

Macromedia Flash movie yang dijalankan dalam *web browser* adalah aman. Sampai saat ini belum ada virus atau *trojan* yang menyerang pengguna Macromedia Flash movies yang bermain melalui *web server*. Macromedia Flash Player mempunyai kebijakan keamanan dan mau tidak mau harus membatasi akses ke sumber sistemnya.

3.1 Playback Macromedia Flash melalui Projector

Ketika file di *executable*, ada isu potensial yang dijalankan dengan Macromedia Flash *projector* dari sumber yang tidak dapat dipercaya, bahwa isu tersebut mendesak untuk berbuat dengan menjalankan *executable* bila dibandingkan dengan Macromedia Flash *projectornya* sendiri. Secara umum, para *user* tidak perlu menjalankan Macromedia Flash *executable* atau file lain yang *executable* kecuali jika telah diterima dari sumber terpercaya.

Untuk diketahui bahwa bagian ini hanya berlaku pada Macromedia Flash *projector* yang dijalankan atau Macromedia Flash movie yang dijalankan dalam *standalone* Macromedia Flash player (secara umum hanya tersedia untuk developers Macromedia Flash). Hal tersebut tidak berlaku pada Macromedia Flash movie yang dijalankan dalam *web browser*.

Ada tiga masalah pokok yang terkait ketika menjalankan Flash movie :

1. macromedia Flash movie sebagai *carrier* dari virus,
2. penyamaran file *malicious* sebagai file Macromedia Flash,
3. macromedia flash movie berisi kode *malicious*.

3.1.1 Macromedia Flash projector carrier dari virus

Seperti file yang di *executable*, ada kemungkinan developer *malicious* menyertakan virus ke file Macromedia Flash player. Jika semua pemakai file macromedia flash *projector* datang dari sumber terpercaya dan telah dikonsultasikan dengan memperbaharui scanner virus, maka isu ini tidak menjadi suatu ancaman bagi pemakai Macromedia Flash player.

Secara umum isu yang berkembang bahwa pada file yang di *executable* tidak dapat mengamankan Macromedia Falsh itu sendiri.

3.1.2 Penyamaran File *malicious* sebagai file Macromedia Flash

Trojan adalah suatu file yang menyamar sebagai dirinya sendiri, tetapi sebenarnya berbeda. Tujuannya adalah untuk mengecoh (meyakinkan) seorang *user* bahwa file yang dijalankan adalah aman walaupun sebenarnya file tersebut tidak aman bila dijalankan.

Dalam kaitan popularitasnya isi dari Macromedia Flash, individu *malicious* mencoba menyembunyikan program mereka sebagai Macromedia Flash movie dengan harapan bahwa banyak orang dapat menggunakan file tersebut.

Ada dua cara untuk memenuhi hal tersebut :

1. file yang didistribusikan dibuat oleh pemikiran *user* yang merupakan Macromedia Flash movie, seperti "*cool_flash_movie.exe*",
2. *icon* untuk *executable* diubah dari Macromedia Flash Player.

Para *user* hanya *mengexecute file* dari sumber terpercaya, dan hanya dari *file trojan*, karena ada beberapa *scanner* virus tidak dapat mendeteksinya.

Secara umum file yang di *executable* tidak menjamin mengamankan dan mengeluarkan Macromedia Flash itu sendiri dari serangan *trojan*.

3.1.3 Isi Macromedia Flash movie dari code *malicious*

Bila tidak ada masalah yang terjadi, secara teoritis untuk Macromedia Flash *projector* atau Macromedia Flash movie *standalone player* pada sistem operasi windows untuk melaksanakan aksi *malicious*. Resiko ini hanya terjadi bila isi

malicious dimainkan kembali (*playback*) dalam *standalone* Macromedia Flash Player dan tidak mempengaruhi permainan movie dalam *browser*.

Sebagaimana tersebut di atas, para *user* hanya perlu menjalankan Macromedia Flash *projector* atau *movie locally*, jika file yang datang dari sumber terpercaya telah diteliti dan diperbaharui dengan penyaring (*update*) *scanner* virus. Ini adalah suatu teoritis yang belum tentu kebenarannya.

3.2 Macromedia Flash movies sebagai *Attachments E-mail*

Ada dua cara pada macromedia Flash movie yang dapat ditransfer melalui *e-mail*. Pertama adalah hanya untuk mengirim movie sebagai suatu pasangan (*attachment*) pada *e-mail*.

kedua adalah meletakkan movie dalam HTML didasarkan pada tujuan *e-mail* dari movie yang dijalankan dari dipandang *e-mail*.

Dalam kedua kasus tersebut di atas, macromedia Flash movie dipandang sebagai sistem file, dengan begitu movie bukan mengamankan tapi melayani Macromedia Flash movie dari *web server* dan *played* di *web server*.

Para *user* hanya perlu menjalankan Macromedia Flash movie dari sistem file atau *e-mail client* jika file tersebut dari sumber terpercaya. Dalam kasus pada Macromedia Flash movie yang diletakan dalam *e-mail*, para *user* perlu mengkonsultasi dokumentasinya pada *e-mail client* untuk memerintah, bagaimana melumpuhkan *autoplay otomatis* dari kendali *autoplay ActiveX* dalam *e-mail*.

IV. Macromedia Flash Player 6 *sandbox*

Ada 6 implementasi Macromedia Flash Player dari rencana keamanan *browser-like sandbox* untuk keamanan dan *privacy* dari kedua Macromedia Flash movi, seperti *machine client*.

4.1 Apa itu *sandbox*

Sandbox menggambarkan satu tempat yang terbatas dimana Macromedia Flash movie dijalankan dalam Macromedia Flash Player untuk beroperasi. Tujuan

utamanya adalah untuk mengintegrasikan keamanan dari *machine client*, seperti halnya keamanan Macromedia Flash movie yang dijalankan dalam *player*.

Konsep dari *sandbox* sangat sederhana. Macromedia Flash movie dalam melaksanakan informasi apapun dalam *sandbox* dapat dikomunikasikan dengan domain dari movie mana yang datang juga di batasi dalam mengakses informasi *sandbox* dari luar.

Macromedia Flash movie *sandbox* terdiri dari :

- macromedia flash movie *sandbox* yang dapat dimasukkan dalam file SWF,
- tindakan *user* diarahkan ke flash movie,
- *domain server* dimulai dari mana Macromedia Flash movie (lihat *authenticati domain-based* dibawah),
- *SharedObjects* yang ditulis dengan Macromedia Flash movie dari *domain* yang sama (lihat akses *file local I/O* di bawah),
- konfigurasi komputer informasi diatasnya terbatas yang dijalankan oleh Macromedia Flash movie.

Dalam mendukung *developmen* dan uji coba Macromedia Flash movie oleh *developers*, movie dapat mengakses file lain dengan pemakai disk atau pada *server* LAN yang tidak dibatasi *sandbox*. Ini konsisten dengan penambahan perhatian para *user* yang selalu mengambil dan menjalankan jenis file tersebut.

4.2 Authenticasi Domain-based

Sebagaimana diketahui bahwa *sandbox* meliputi semua *domaion server* atau "*nth-level sub-domain*", dari mana movie dimulai. *Membership* dalam domain dapat di cek dengan membandingkan nama *server* -nya.

Ada dua *server domain* yang sama adalah sebagai berikut :

1. nama *server* harus mengenal baik tanda (*token*) yang sama,
2. ada sedikitnya 3 tanda,
3. semua tanda adalah sama kecuali tanda pertama yang berbeda.

Jika nama *server* dinyatakan dalam satu atau dua tanda (contoh, "foo" untuk foo.macromedia.com), maka *server* tersebut adalah *domain* dengan sendirinya. Jika

server dikenali dan menunjuk IP-nya, maka *server* tersebut menjadi domain dengan sendirinya juga.

Contoh :

- A.B.Macromedia.com dan C.b.Macromedia.Com adalah domain yang sama,
- A.b.Macromedia.com dan www.macromedia.com bukanlah domain yang sama (berbeda jumlah tandanya),
- A.b.Macromedia.com dan a.c.macromedia.com bukanlah domain yang sama (tanda yang kedua berbeda).

Idealnya adalah bahwa suatu domain yang besar harus diijinkan untuk dipecah menjadi sub-sub domain yang terisolasi dengan menggunakan tambahan tanda dalam nama *server*-nya. Algoritma ini berlaku untuk URLS.

Pembatasan pada *one-end two-token server names* dan *server* yang dikenali oleh alamat IP untuk menghindari kasus *hard-to-compute*, seperti ketika seorang *user* mempunyai beberapa pencarian domain dapat dilihat dari konfigurasi TCP/IP-nya.

Ada dua contoh yang dapat dilihat sebagai berikut :

- foo dan bar bukan domain yang sama, tetapi foo.macromedia.com dan bar.macromedia.com adalah berada dalam domain yang sama,
- www.macromedia.com dan macromedia.com bukan di domain yang sama, walaupun fakta menunjukkan bahwa mereka berada dalam server yang sama.

Alasan untuk pembatasan domain adalah untuk melindungi *domain server* dari serangan *firewalls* oleh mesin luar *firewall*. Movie dari *outside.hacker.org* harus melindungi pembacaan data dari penyimpanan file pada *inside.macromedia.com* ketika movie sedang dijalankan pada *machine Macromedia firewall*.

4.3 Bagaiman mengimplementasikan *sandbox*

4.3.1 Mengakses I/O file

Macromedia Flash Palyer 6 memungkinkan pembatasan penyimpanan file melalui penggunaan *SharedObjects*. *SharedObjects*, dalam kaitannya dengan *web-browser*, juga memungkinkan *developer* untuk menyimpan dan mendapat kembali informasi sistem file para *user*.

Pembatasan *SharedObjects* adalah sebagai berikut :

- data hanya dapat ditulis sama persis dan benar dari spesifikasi *directori*, dimana *developer* tidak dapat mengendalikan *direktori* tersebut,
- user mempunyai kontrol dari penyimpanan beberapa data termasuk kemampuan untuk menghilangkan (melumpuhkan) penyimpanan data tersebut pada suatu basis per-domain,
- data yang disimpan pada sistem file adalah data yang dijadikan *serial biner* yang dikendalikan oleh Macromedia Flash Player. Dimana semua file mempunyai *standard header*,
- untuk mengakses data terbatas oleh autentikasi melalui peraturan *domain-based* tentang Macromedia Flash Player (lihat *domain-based autentikasi* di atas)

Semua pergantian file melalui Macromedia Flash Player dibatasi oleh suatu spesifikasi *directori* dengan *machine-client* yang diciptakan Macromedia Flash Player 6. Berikut adalah pembatasan *directori* ketika diakses dari permainan Macromedia Flash movie di dalam *web browser* :

- berapa banyak user mengontrol informasi yang tersimpan dalam suatu domain tertentu. Jumlah data yang disimpan dalam *machine user* untuk tiap domain adalah 100K, dimana masing-masing obyek menyimpan data ke dalam file tidak lebih dari 1000 byte dari batas maksimal 100K berbeda dengan *SharedObject* untuk satu domain dapat diijinkan sampai ke 100,
- data untuk tiap domain disimpan dalam direktori individual yaitu *direktori "data aplikasi"*. *Direktori* ini dapat diakses oleh *user*.

Pembatasan ini dilakukan untuk :

- informasi *Non-Flash* pada *machine user* tidak dapat *overwritten* oleh Macromedia Flash movie yang dijalankan dalam *browser*,
- ancaman dari serangan *Denial* disebabkan oleh pengisian pemakai data diperkecil.

Informasi yang disimpan dalam file Macromedia Flash Player movie dapat di kontrol dan diformat oleh *Player*. Developer tidak mempunyai kewenangan untuk menformat dan mengontrol data. Informasi adalah data yang disimpan dalam *binari serialisasi* dan tidak diperbolehkan untuk menyerang pada *machine client*.

Data yang dimasukkan dalam *SharedObjects* boleh saja diakses di bawah peraturan *domain-based*. Sebagai contoh, data yang disimpan dalam *SharedObjects* dari movie *www.domain1.com*. tidak dapat diakses oleh movie dari *www.domain2.com*.

4.3.2 Komunikasi Cross-movie

Macromedia Flash Player *sandbox* juga digunakan ketika *loading* SWT ke dalam Macromedia Flash movie yang ada. Bagaimanapun, movie dapat dipisahkan dari domain yang tertadapat dalam *sandbox* itu sendiri. Movie ini dapat mengisolasi *playing* dari *player*. Content dalam *sandbox movie* tidak dapat di baca dari luar *sandbox*, dan content di luar *sandbox* tidak dapat pula di ketahui maksud yang tersembunyi dalam *sandbox* tersebut.

Ini adalah untuk komunikasi movie antara satu dengan lainnya melalui penolakan *LocalConnection*. Model keamanan *sandbox* bagi komunikasi *cross-movie* menggunakan metode tersebut. Aturannya adalah movie di domain A tidak dapat menyadap informasi apapun dari movie di domain B kecuali bila movie dari domain B memberikan sinyal eksplisit ke domain A untuk dapat diakses. Hal tersebut merupakan suatu aturan yang harus dipenuhi untuk mencegah SWF dari public internet terhadap *loading firewall* dan penyadapan data. Sebagai solusinya, tiap *sandbox* hanya menerima copiannya sendiri secara global. Kode dalam *sandbox* hanya dapat mengakses objek global *sandbox* itu sendiri.

Olehnya itu, kadang-kadang penting untuk dua movie berada dalam domain yang terpisah untuk mengakses data satu sama lain. Contohnya Macromedia Answers Panel yang berada dalam disk melalui sarana Macromedia Flash, tetapi bila ingin memperbaharainya dapat mengakses Macromedia website. Dalam hal ini diperlukan isi movie dari *www.macromedia.com* untuk ditukar dengan data isi movie dari disk tersebut.

Dimana kebutuhan ini menggunakan "*tunnelling*" dari *sandbox*. Implementasi *tunnelling* diterapkan oleh *ActionScript method System.security.allowDomain* dan mempunyai sintaks sebagai berikut :

System.security.allowDomain("macromedia.com")

Perintah ini untuk menambah macromedia.com pada pendaftaran *shim "friends"*. SWF yang memuat macromedia.com atau sub domain, seperti sub.macromedia.com sekarang dapat mengakses variabel *shim* SWF

Sekali mengakses tidak dapat di tarik kembali walaupun mendapat ijin dari domain untuk mendaftar kembali.

Komunikasi data dibatasi antara *sandbox*. Kode *ActionScript* oleh satu *sandbox* dapat memperoleh acuan obyek tertinggi dari *sandbox* lain dan memodifikasi movie clip *_level10.mcHolder*, dengan menggunakan :

- kode *movie1.swf* dapat mengakses *_level10.mcHolder* dan memodifikasi movie clip properties,
- *movie1.swf* tidak dapat mengakses properties *_level10.cmHolder* lainnya.

Properties movie clip yang tersedia terdaftar dalam brosur properties Action Panel's toolbox meliputi :

<i>_alpha</i>	<i>_target</i>
<i>_currentframe</i>	<i>_totalframe</i>
<i>_droptarget</i>	<i>_url</i>
<i>_focusrect</i>	<i>_visible</i>
<i>_framesloaded</i>	<i>_x</i>
<i>_height</i>	<i>_xmouse</i>
<i>_name</i>	<i>_xscale</i>
<i>_quality</i>	<i>_y</i>
<i>_rotation</i>	<i>_ymouse</i>
<i>_soundbuftime</i>	<i>_yscalel</i>

Berikut ini adalah tabel basic dari fungsi *ActionScript* dan batasan keamanannya

Tabel 1: Batasan keamanan pada *basic action*.

<i>Action</i>	<i>Security restrictions</i>
<i>GotoAndPlay/gotoAndStop</i>	No security restriction unless a target path is specified, for example: <i>GotoAndPlay("mc : 1");</i> If a target path is specified the caller must be in the same sandbox as the target movie clip.
<i>play</i>	No security restrictions
<i>stop</i>	No security restrictions
<i>toggleHighQuality</i>	No security restrictions
<i>stopAllSounds</i>	No security restrictions
<i>getURL</i>	No security restrictions
<i>FSCCommand</i>	Caller must be in the same sandbox as HTML page
<i>loadMovie</i>	No security restrictions
<i>loadMovieNum</i>	No security restrictions
<i>unloadMovie</i>	No security restrictions
<i>unloadMovieNum</i>	No security restrictions
<i>loadVariables</i>	Caller must be in the same sandbox as the target movie clip
<i>loadVariablesNum</i>	Caller must be in the same sandbox as the target movie clip
<i>tellTarget</i>	Caller must be in the same sandbox as the target movie clip
<i>ifFrameLoaded</i>	No security restrictions
<i>print</i>	No security restrictions
<i>printNum</i>	No security restrictions
<i>printAsBitmap</i>	No security restrictions
<i>printAsBitmapNum</i>	No security restrictions

Pembatasan pada *sandbox* tidak berlaku bagi Macromedia Flash movie dan *projektor* dari sistem file, kecuali *Sandbox* dari Flash movie bukan dari file, dan tidak dapat mengakses *sandbox* file tersebut. contoh movie1.swf pada *user disk* yang memuat movie2.swf dari server HTTP, hal ini dapat dibenarkan, sebab :

- movie1.swf dan movie2.swf terpisah dari *sandbox*,
- movie1.swf boleh mengakses indeks (isi) dari movie2.swf,
- movie2.swf dihalangi untuk mengakses indeks os movie1.swf.

V. Komunikasi *Flash-JavaScript*

Macromedia Flash Player mensupport *JavaScript* API untuk mengendalikan *properties movie*, menentukan jumlah variabel serta fungsi panggilan. Kapabilitas ini juga terbatas pada Model keamanan *sandbox* Macromedia Flash Player.

5.1 LiveConnect API

Netscape menghubungkan versi Macromedia Flash Player dan API melalui penghubung *Netscape LiveConnect Navigator*. API ini dimungkinkan untuk diakses oleh *JavaScript* yang dijalankan dalam *Netscape Navigator*.

Pembatasan keamanan dengan *LiveConnect* API adalah pembatasan paralel *sandbox* untuk *ActionScript*. *Sandbox* yang sama diciptakan untuk halaman HTML dengan menggunakan URL HTML. Untuk *sandbox* ini, dipakai dengan menggunakan aturan yang normal. Dimana variabel dari *sandbox* tidak dapat diakses dari *sandbox* yang lain karena *Movie clip* dalam *sandbox* lain tidak dapat dikendalikan *Movie clip properties* dari *movie clip top-level sandbox* lainnya hanya dapat membaca data tetapi tidak untuk menulis.

Berikut adalah tabel dari semua *properties* dan *methode di LiveConnect* API serta pembatasan hubungan keamanan.

Tabel 2 : Pembatasan keamanan pada Macromedia Flash Player LiveConnect API.

<i>Java/JavaScript method</i>	<i>Security restrictions</i>
<i>boolean IsPlaying();</i>	<i>No Security restriction</i>
<i>void Play();</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>void StopPlay();</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>int totalFrames();</i>	<i>No Security restriction</i>
<i>int CurrentFrame();</i>	<i>No Security restriction</i>
<i>void GotoFrame(int position);</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>void Rewind();</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>void Back();</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>void Forward();</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>int PercentLoaded();</i>	<i>No Security restriction</i>
<i>boolean FrameLoaded(int frameNum);</i>	<i>No Security restriction</i>
<i>int FlashVersion();</i>	<i>No Security restriction</i>
<i>void Pan(int x, int y, int mode);</i>	<i>No Security restriction</i>
<i>void Zoom(int percent);</i>	<i>No Security restriction</i>
<i>void SetZoomRect(int left, int top, int right, int bottom);</i>	<i>No Security restriction</i>
<i>void LoadMovie(int layer, String url);</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>void TGotoFrame(String target, int frameNum);</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>

<code>void TCurrentLabel(String target, String label);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>int TCurrentFrame(String target);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>String TCurrentLabel(String target);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>void TPlay(String target);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>void TStop(String target);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>void SetVariable(String name, String value);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>String GetVariable(String name);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>void TSet Property(String target, int property, String value);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>String TGetProperty(String target, int property);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>Void TCallFrame(String target, int frameNum</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>void TcallLabel(String target, String label);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>double TGetPropertyAsNumber(String target, int property);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<code>void TSetProperty(String target, int property, double value);</code>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>

Catatat : *Jika liveConnect tidak tersedia, maka perintah ini tidak dapat digunakan dan tidak menggunakan batasan keamanan.*

5.2 Control ActiveX API

API sering digunakan dalam kode *JavaScript* dan halaman (*page*) HTML sebagai Macromedia Flash movie. *ActiveX* API menyerupai *LiveConnect* API yang disajikan ke *JavaScript* dalam *Netscape Navigator*.

Pembatasan keamanan dengan *ActiveX control* API adalah batasan paralel *sandbox* untuk *ActionScript* dan merupakan aturan normal pada *sandbox* yang sama. Variabel dari *sandbox* ini tidak dapat diakses dari *sandbox* yang lain. *Movie clip* dalam *sandbox* yang lain dari suatu *movie clip properties top-level* dari *sandbox* yang lain hanya untuk membaca dan bukan untuk menulis.

Berikut adalah Tabel dari semua properties dan *methoda ActivwX Control* API dan hubungannya dengan keamanan restrictions.

Tabel 3 : Batasan keamanan pada Macromedia Flash Player ActiveX Control API

<i>Property/method</i>	<i>Security restrictions</i>
<i>SWRRemote property</i>	<i>No Security restriction</i>
<i>TGetPropertyNum method</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<i>TSetPropertyNum method</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<i>TCallLabel method</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<i>TCallFrame method</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<i>TGetProperty method</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<i>TSetProperty method</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<i>GetVariable method</i>	<i>HTML page must be in the same sandbox as the variable's container object</i>
<i>SetVariable method</i>	<i>HTML page must be in the same sandbox as the variable's container object</i>
<i>TStopPlay method</i>	<i>HTML page must be in the same sandbox as the variable's container object</i>
<i>TPlay method</i>	<i>HTML page must be in the same sandbox as the variable's container object</i>
<i>TCurrentLabel method</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<i>TCurrentFrame method</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<i>TGotoLabel method</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<i>TGotoFrame method</i>	<i>HTML page must be in the same sandbox as the targeted movie clip</i>
<i>Quality2 property</i>	<i>No Security restriction</i>
<i>BGColor property</i>	<i>No Security restriction</i>
<i>EmbedMovie property</i>	<i>No Security restriction</i>
<i>DeviceFont property</i>	<i>No Security restriction</i>
<i>Scale property</i>	<i>No Security restriction</i>
<i>Based property</i>	<i>HTML page must be in the same sandbox as _level0</i>
<i>Menu property</i>	<i>No Security restriction</i>
<i>SAlign property</i>	<i>No Security restriction</i>
<i>WMode property</i>	<i>No Security restriction</i>
<i>PercentLoaded property</i>	<i>No Security restriction</i>
<i>Pan method</i>	<i>No Security restriction</i>
<i>Zoom method</i>	<i>No Security restriction</i>
<i>SetZoomRect method</i>	<i>No Security restriction</i>
<i>FrameNum property</i>	<i>READ: No Security restriction; WRITE: HTML page must be in the same sandbox as _level0</i>
<i>Movie property</i>	<i>No Security restriction</i>
<i>Loop property</i>	<i>READ: No Security restriction; WRITE: HTML page must be in the same sandbox as _level0</i>
<i>BackgroundColor property</i>	<i>No Security restriction</i>
<i>AlignMode property</i>	<i>No Security restriction</i>
<i>ScaleMode property</i>	<i>No Security restriction</i>
<i>Quality property</i>	<i>No Security restriction</i>

<i>Playing property</i>	<i>READ: No Security restriction; WRITE: HTML page must be in the same sandbox as_level0</i>
<i>TatalFrames property</i>	<i>No Security restriction</i>
<i>ReadyState property</i>	<i>No Security restriction</i>
<i>FalshVersion method</i>	<i>No Security restriction</i>
<i>FrameLoaded method</i>	<i>No Security restriction</i>
<i>CurrentFrame method</i>	<i>No Security restriction</i>
<i>GotoFrame method</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>Rewind</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>Forward</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>Back</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>IsPlaying</i>	<i>No Security restriction</i>
<i>Stop</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>StopPlay</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>Play</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>LoadMovie</i>	<i>HTML page must be in the same sandbox as_level0</i>
<i>FlashVers property</i>	<i>HTML page must be in the same sandbox as_level0</i>

VI. Hubungan Keamanan Lainnya

Sebagai tambahan tentang mengamankan Macromedia Flash adalah dengan melihat sejumlah pesan *e-mail* dan *post* pada sistem file, kamera dan kapabilitas microphone dari Macromedia Flash dalam mengimplementasikannya.

Semua fitur ini akan menjamin keamanan setiap *domain browser*. Juga, dalam rangka mencegah serangan *cross-site*, data pada SWF harus dibatasi dari setiap domain di mana SWF sendiri yang menyimpannya. Movie pada www.macromedia.com dapat memuat informasi dari *foo*. [Macromedia.com](http://www.macromedia.com) tetapi bukan dari www.nastyhackerperson.com.

VII. Penutup

Dengan dukungan terhadap Macromedia Flash belakangan ini semakin luas, makin penting bagi perusahaan-perusahaan pengelola Macromedia Flash MX, karena semakin pesatnya kemajuan teknologi Macromedia Flash makin mudah para penyusup untuk merusak sistem.

Menurut survei 98% browser telah terpasang Flash Player untuk dapat menampilkan animasi Flash dengan baik. Dengan demikian adalah suatu tantangan bagi perusahaan-perusahaan yang bergerak dibidang Macromedia Flash MX ini secara ideal dapat menjamin kepercayaan para user.

Tugasi ini hanya menguraikan sebuah studi dengan pemikiran bahwa perkembangan teknologi Macromedia Flash MX dapat berkembang secara komersil.

Referensi

- [1] Macromedia Security Zone
<http://www.macromedia.com/v1/developer/SecurityZone/>
- [2] Contains security bulletins and technical briefs about security issues.
Macromedia Flash Player
<http://www.macromedia.com/software/flashplayer/>
- [3] Contains information and resources for Macromedia Flash Player.
Macromedia Flash MX
<http://www.macromedia.com/software/flash/>
- [4] Contains information and resources for the Macromedia Flash MX authoring environment.
Macromedia Flash Support Center
<http://www.macromedia.com/support/flash/>
- [5] Contains technical resources and information, including security related information, on Macromedia Flash development.
Macromedia Designer & Developer Center
<http://www.macromedia.com/desdev/>
- [6] Contains articles, tutorials, and other information on Macromedia Flash development.
Macromedia Flash Player security e-mail address
flashplayer_security@macromedia.com