IMPLEMENTASI MEKANISME KEAMANAN DATA DENGAN ENKRIPSI SYMMETRIC KEY DAN CHECKSUM VARIABLE PADA GAME ONLINE

Herru Darmadi

Computer Science Department, School of Computer Science, Binus University
Jl. K.H. Syahdan No. 9, Palmerah, Jakarta Barat 11480
me@herrucules.com

ABSTRACT

This study discusses specifically the encryption mechanism that maintains the integrity of the game *data on online* games*. The purpose of this study is to implement symmetric key encryption and checksum for score calculation. There are two methods conducted in this study: analysis and design. The analysis method is performed in two ways: analysis of existing systems and analysis of literature study. System design method used is prototyping. The results obtained are implemented on online* game *competition 'Expedisi Indosat' through Facebook from 3 – 30 July 2011. The implementation of symmetric key encryption mechanism and the checksum is useful to maintain data integrity on* games *especially the results score when the* game *is running and when there is data transmission from the client to the server.*

Keywords: *symmetric key, checksum, encryption, decryption, integrity of the data* game

ABSTRAK

Pada penelitian ini dibahas secara spesifik mekanisme enkripsi dalam permainan game *sehingga menjaga integritas data pada* game *online. Tujuan dari penelitian ini adalah mengimplementasikan symmetric key encryption dan checksum untuk mekanisme perhitungan skor. Metode penelitian yang dilakukan dalam penelitian ini ada dua, yaitu metode analisis dan metode perancangan. Metode analisis dilakukan dengan dua cara, yaitu dengan analisis system yang telah ada dan analisis studi pustaka. Metode perancangan sistem yang digunakan adalah prototyping. Hasil yang diperoleh diimplementasikan pada kompetisi* game *online Ekspedisi Indosat melalui social media Facebook pada periode 3 Juli 2011 sampai 30 Juli 2011. Implementasi mekanisme enkripsi symmetric key dan checksum berguna untuk menjaga integritas data pada* game *khususnya skor yang dihasilkan dari* game *pada saat* game *dimainkan dan pada saat terjadi transmisi data dari client ke server.*

Kata kunci: *symmetric key, checksum, enkripsi, dekripsi, integritas* game *data*

PENDAHULUAN

Populernya permainan game *online* diminati oleh para pemilik usaha untuk memasarkan dan mempromosikan produknya, salah satunya dengan pemberian *reward* melalui kompetisi-kompetisi yang dilakukan melalui media. Sarana dari game *online* juga didukung dengan adanya *platform* media sosial sehingga terbentuklah *social media* games yang tentunya akan semakin memperluas *target market* dari pemasang promosi.

Mekanisme kompetisi pun harus dibuat semudah mungkin sehingga peserta akan semakin banyak dan dapat mengikuti kompetisi dengan mudah, dan dicarilah pemenangnya melalui pemain dengan jumlah skor tertinggi dan frekuensi permainan terbanyak. Isu utama dari sebuah game *online* adalah adanya sejumlah pemain yang bertindak curang/*cheaters* yang bertujuan untuk mendapatkan skor tertinggi dan menjadi pemenang. Keserakahan adalah penyebab utama dari kecurangan pada game *online* (Glenn, 2007).

Dengan adanya kecurangan ini, akan terjadi kesulitan penentuan pemenang dan menimbulkan ketidakadilan bagi pemain-pemain lain yang berusaha untuk mendapatkan nilai terbaik. Pengembang game dan tim sudah menentukan batas rasional untuk nilai, tetapi tidak dimungkinkan untuk membatasi nilai tersebut karena mungkin ada pemain yang bisa mencetak skor melebihi yang ditentukan. Selain itu secara psikologis akan berdampak negatif dan tentu akan mempengaruhi pemain lainnya.

Menurut Ferretti (2007), mencegah modus tindakan kecurangan harus dilakukan dengan memikirkan keseluruhan kemungkinan yang dapat mengeksploitasi kelemahan pada game. Salah satu solusi adalah dengan membatasi ruang gerak pemain dengan memaksa pemain menghasilkan game *event* pada selama durasi waktu tertentu, tetapi kecurangan pada waktu dapat dengan mudah dilakukan.

Maka dari itu, harus dibuatkan sebuah mekanisme untuk menjaga integritas data dan mengurangi tingkat kecurangan dan pendeteksian tingkat kecurangan berdasarkan rangkuman data hasil permainan. Mekanisme ini bertujuan untuk menjaga integritas data skor pada game sehingga meminimalisasi pemain untuk bertindak curang dengan mengubah data pada game.

Symmetric key encryption lebih unggul dibandingkan dengan *public key encryption* dari segi kecepatan dan kecilnya *overhead chipertext* yang dihasilkan. Namun kekurangannya adalah *shared-key* antara kedua pihak untuk proses enkripsi dan dekripsi (Scammell, 2004). *Checksum variable* juga dipergunakan untuk menjaga integritas pada saat pengiriman data dari game *client* ke *server* untuk menghindari terjadinya perubahan informasi melalui serangan *Man In the Middle* atau *sniffing* (Rivello, 2011). Tujuan dari penelitian ini adalah mengimplementasikan enkripsi *symmetric key* untuk mekanisme perhitungan skor dan *checksum* untuk integritas data. Hasil/manfaat dari penelitian ini adalah terciptanya aplikasi game dengan integritas data skor.

METODE

Penelitian ini dimulai dengan menganalisis kebutuhan dari user yaitu Ide 73 selaku agensi multimedia rekanan Indosat yang akan membuat sebuah kompetisi dalam bentuk games *online* melalui *social* media *Facebook*. Melalui tahap wawancara yang dilakukan oleh peneliti dengan user, ditemukan kebutuhan dan permasalahan yang muncul pada kompetisi sejenis sebelumnya. Permasalahan utama yang muncul yaitu terjadinya kecurangan yang dilakukan oleh beberapa pemain

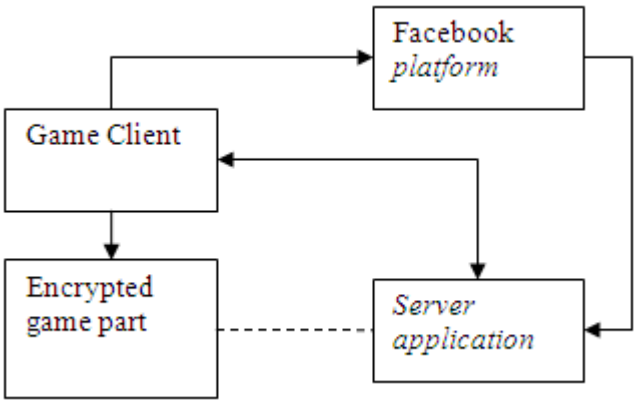
dalam hal pengumpulan nilai/skor pada permainan. Selanjutnya dilakukan studi kepustakaan mengenai metode-metode pengamanan data baik pada game dan juga pada transmisi data yang terjadi pada jaringan internet antara *client* dan *server*. Kemudian dilakukan perancangan aplikasi games yang menerapkan teknik-teknik pengamanan data melalui enkripsi dan *checksum variable*. Pengujian dan implementasi game online Ekspedisi Indosat pada Facebook dilakukan pada periode 3 Juli 2011 sampai 30 Juli 2011. Setelah itu, kami membandingkan hasil skor game antara yang menggunakan mekanisme enkripsi dan *checksum* dan yang hanya menggunakan *checksum*.

Penelitian ini akan berfokus pada penerapan *symmetric key encryption* dan *checksum variable* untuk mekanisme perhitungan skor pada game *online* berbasis Adobe Flash. Algoritma yang dipergunakan dalam enkripsi *symmetric key* ini adalah *Advanced Encryption Standard* (AES). Ditambahkan dengan *checksum* kunci rahasia yang diciptakan pada *server* untuk mengecek integritas data ketika dikirimkan dari *client* ke *server*. Kunci rahasia ini ditanamkan pada *swf file* sebelum diunduh ke *client*. Selanjutnya *swf file* ini akan dienkripsi dengan metode *obfuscation* sehingga tidak dapat didekripsi di client dan pada kunci rahasia yang ditanamkan hanya berlaku pada saat *session* game berlangsung dan periode game tertentu.

HASIL DAN PEMBAHASAN

Rancangan Sistem

Diagram blok game dapat dilihat pada Gambar 1.



Gambar 1 Diagram blok system

Secara garis besar aplikasi dibagi menjadi dua, yaitu aplikasi pada *server* dan aplikasi pada *client*. Aplikasi pada *server* yaitu Facebook *platform* dan *server application* sebagai *host* game. Aplikasi pada *client* yaitu game *client* dan *encrypted* game *part*. *Server application* akan mendapatkan *unique* ID dari *Facebook platform* untuk media authentifikasi kemudian *server* akan menginisialisasi game untuk segera diunduh ke *client*. kemudian *server* akan men-*generate server secret key* dan *shared-key* untuk *checksum* yang akan ditanamkan pada Encrypted game part yang akan diunduh oleh game *client*. Encrypted game part akan mengalami *obfuscation* yaitu proses pengacakan kode sumber program, sehingga bila terjadi proses dekompilasi yang dilakukan oleh pemain yang curang, kode sumber program akan sulit terbaca. *Server secret key* dan *shared-key* hanya aktif pada sesi permainan berlangsung dan mempunyai batasan waktu kadaluarsa tertentu yang telah ditentukan di *server*.

Pada saat game berjalan, proses enkripsi-dekripsi akan terjadi apabila terjadi perubahan pada data skor. Proses ini ditujukan untuk menghindari terekspos nya data skor pada *memory* komputer pada saat game client berjalan. *RAM Hacking* dengan *tools* seperti *Cheat Engine* atau *Quick Memory Editor* akan sulit untuk mendeteksi alamat *memory* pada game yang berisi data skor pada saat game client berjalan (Rivello, 2011). Setelah game *client* selesai dimainkan, terjadi mekanisme untuk transmisi data nilai skor dari *client* ke *server* dan disinilah *checksum variabel* dipergunakan untuk menjaga integritas nilai skor yang akan dikirimkan. *Checksum variabel* akan berisi nilai skor yang terenkripsi digabungkan dengan *secret key* dan juga *player ID*. Kemudian dikirimkan ke *server* untuk dicek kebenarannya dengan membandingkannya dengan *server checksum variable.* Bila cocok, *server* akan melakukan proses penyimpanan nilai skor ke *database*. Rancangan diagram alir sistem mulai dari inisialisasi sampai ke tahap akhir ditampilkan pada Gambar 2.

Pengembangan Game

Pengembangan game dibagi menjadi dua bagian besar,yaitu: (1) game *client* yang menggunakan teknologi Adobe Flash, dan Facebook API; (2) game *server* yang menggunakan teknologi PHP, MySQL, dan Facebook API.

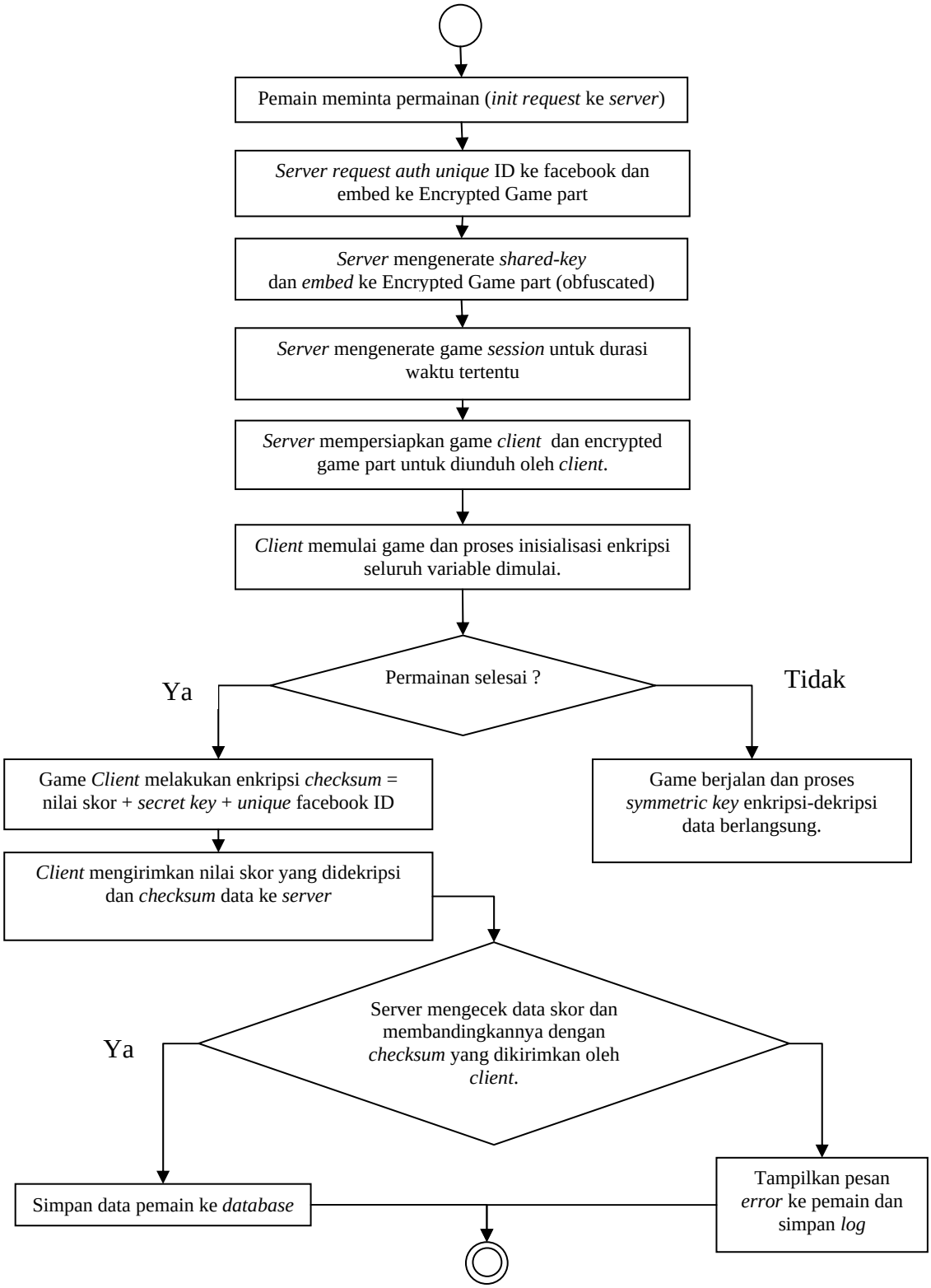
Berikut ini adalah *header* kode program enkripsi pada game *client*.

```
function setCheckSum(fbid:String, score:uint):String;
function initScore():String;
function initFeul():String;
function decryptScore(val:String):String;
function decryptFeul(val:String):String;
function setScore(val:String, additional:int):String;
function setFeul(val:String, additional:int):String;
function encrypt(input:String, key:String):String;
function decrypt(input:String, key:String, iv:String):String;
```

Proses enkripsi/dekripsi akan dilakukan pada dua variabel utama yang menunjang interaksi permainan yaitu *score* dan *feul*. Proses enkripsi untuk variabel *score* akan dilakukan dengan memanggil *initScore*() pada saat game dimulai dan *setScore*(*val*, *additional*) setiap kali isi variabel berubah karena adanya aksi dalam game. Untuk menampilkan isi variabel *score* kepada pemain, isi variabel ini harus didekripsi dengan memanggil *decryptScore*(*val*) dan ditampilkan kelayar monitor pemain. Proses enkripsi untuk variabel *feul* akan dilakukan dengan memanggil *initFeul*() pada saat game dimulai dan *setFeul*(*val*, *addition*) setiap kali isi variabel berubah karena adanya aksi dari dalam game.

Pada Gambar 3 tampak bahwa variabel *score* ditampilkan dalam bentuk tipe data *integer* yang terdapat dibawah foto dan nama profil pemain. Namun pada game, variabel ini tidak disimpan di *memory* dalam bentuk *integer*, melainkan dalam bentuk *String* yang telah dienkripsi (Scammell, 2004) (Yaiser, 2011). Variabel *feul* direpresentasikan dalam bentuk blok batang yang terdapat di kanan atas game yang juga direpresentasikan dalam bentuk skala dengan tipe data integer.

Berikut ini adalah fungsi untuk menginisialisasi komponen enkripsi-dekripsi variabel yang akan dipergunakan dalam game dan *checksum*.

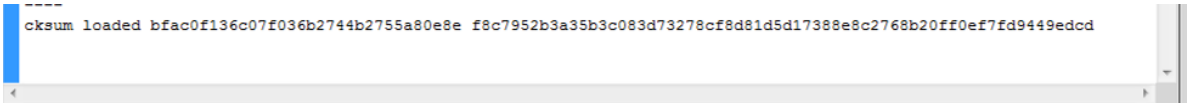


Gambar 2 *Flowchart* mekanisme game dan proses enkripsi

Gambar 3 Implementasi game *online* pada Facebook

```
function loadCheckSUM(){
	var loader:SWFLoader = new SWFLoader("checksum.swf",
				{onComplete:loadCheckSUMComplete});
	loader.load();
	trace("loading cksum");

	function loadCheckSUMComplete(e:LoaderEvent){
		CheckSUM = loader.rawContent;

		gameScore = CheckSUM.initScore();
		gameScoreStage3 = gameScoreStage2 = gameScoreStage1 = gameScore;

		pemain.feul = CheckSUM.initFeul();
		pemain.feul = CheckSUM.setFeul(pemain.feul, 4);

		trace("cksum loaded "+CheckSUM.setCheckSum("test",1234) + " " + gameScore);

	}
}
```

Log output dari variabel *checksum* dan *score* yang telah dienkripsi ditampilkan pada Gambar 4.



Gambar 4 Log output dari variabel checksum dan score yang telah dienkripsi

Berikut ini adalah kode program pada game *server*.

```
function validate_checksum($user_id, $score, $_POST['checksum']);
function generate_checksum($session_id);
```

Kedua fungsi utama ini yang berperan dalam melakukan inisialisasi *key* yang akan diembed kedalam game client sebelum ditransmisikan ke komputer pemain. Dan melakukan proses validasi dan pemyimpanan data kedalam database system.

Berikut ini adalah potongan kode program dalam bentuk *Javascript* pada game *server* yang diciptakan dengan PHP pada *Facebook Apps*.

```
<script type="text/javascript">
    /*<![CDATA[*/
            swfobject.embedSWF('game.swf', 'game', '640', '480', '9.0.45',
                'swfobject/expressinstall.swf', {domain: '*',fbuid: '<?php echo
$user?>', fbname: '<?php echo $user_profile['name']?>'} ,
{allowFullScreen:'true',align:'middle', allowscriptaccess: 'always', menu:
'false', wmode:'transparent'}, {id: 'game'});
        /*]]>*/
</script>
```

Hasil/manfaat dari penelitian ini adalah terciptanya aplikasi game dengan integritas data skor. Seluruh variabel yang berhubungan dengan game*play* akan dienkripsi dengan menggunakan *symmetric key* AES yang terdapat pada *Encrypted* game *part*. Proses dekripsi hanya berlaku ketika sebuah variabel mengalami perubahan nilai dan seluruh nilai yang mempengaruhi perubahan nilaipun mengalami enkripsi dan dekripsi pada saat terpanggil saja, sehingga nilai-nilai variabel tidak akan terekspos secara publik selama game berjalan. Keuntungan dari metode ini adalah akan mempersulit program game *hacking* seperti *memory scanner* untuk mencari dan mengubah nilai-nilai variabel yang mengalami perubahan. Kekurangannya dari metode ini adalah akan adanya *overhead* pada setiap kali terjadi perubahan nilai pada variabel yang bersangkutan. Transmisi data antara client dan *server* tidak dilakukan melalui *secure channel* dan untuk menangani data *alteration* melalui serangan *Man In the Middle*, maka dipergunakan mekanisme *checksum* untuk memastikan integritas data (Yaiser, 2011).

Implementasi game diterapkan dalam empat periode yang terdiri dari satu minggu di mana sesi per-game terdiri dari empat menit. Pada periode pertama dan kedua tidak diterapkan mekanisme *checksum*, sehingga didapatkan 1053 sesi permainan dan skor tidak wajar dalam 49 sesi permainan. Pada periode ketiga dan keempat diterapkan mekanisme enkripsi *symmetric key* dan *checksum*, sehingga didapatkan 1683 sesi permainan dan hanya 9 sesi permainan dengan skor yang tidak wajar. Dari hasil ini berarti dengan mekanisme enkripsi *symmetric key* digabungkan dengan *checksum* akan menekan angka skor yang tidak wajar dari 4% menjadi 0.5%.

Penentuan pemenang dari kompetisi ditentukan juga dari frekuensi permainan dan nilai skor yang didapatkan dari setiap sesi permainan yang dimainkan oleh pemain tersebut. Bila didapati kecenderungan yang tidak wajar, pemain tersebut tidak akan berhak untuk menjadi pemenang.

Masih ditemukan beberapa kelemahan dari mekanisme ini yaitu bila *encrypted* game *part* berhasil didekripsi dan pemain menemukan *secret key* dari *server*, pemain tersebut dapat memanipulasi data skor. Selain itu, adanya faktor lain yaitu *database security* dimana adanya akses dari *remote* memungkinkan manipulasi data (yang tidak dibahas pada penelitian ini).

PENUTUP

Mekanisme enkripsi *symmetric key* dan *checksum* pada game *data* mengurangi tingkat kecurangan pada data skor pada game *online*, sehingga akan meningkatkan integritas data yang akan selanjutnya berguna dalam menentukan pemenang dari kompetisi berdasarkan skor *online* dan frekuensi permainan dari pemain tersebut. Guna pengembangan lebih lanjut disarankan implementasi enkripsi lainnya untuk mendapatkan hasil terbaik.

DAFTAR PUSTAKA

Ferretti, Stefano. (2007). *Cheating Detection Through Game Time Modeling: A Better Way to Avoid Time Cheats tn P2P Mogs?* New York: Springer.

Glenn, David. (2007). *Video Games and Cheating.* Washington: The Chronicle of Higher Education.

Rivello, Samuel. (2011). *Understanding Game Development with Flash Technologies.* Diakses 10 Oktober 2012 dari http://www.adobe.com/devnet/games/articles/getting-started-flash-games.html#articlecontentAdobe_numberedheader_9.

Scammell, Rupert. (2004). *Cryptography for Game Developers*. San Fransisco: Game Developer.

Yaiser, Michelle. (2011). *ActionScript 3 Fundamentals: Data Types*. Diakses 10 Oktober 2012 dari http://www.adobe.com/devnet/actionscript/learning/as3-fundamentals/data-types.html.