Aplikasi Tanda Tangan Digital sebagai Tindakan Antisipatif Pembajakan Akun Facebook

Yosef Ardhito Winatmoko/13509052[1]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13509052@std.stei.itb.ac.id

***Abstraksi*—Saat ini internet sudah menjadi kebutuhan yang dimiliki hampir setiap orang. Kekurangan dari komunikasi via internet adalah setiap tindakan yang dilakukan tidak memiliki identitas. Identitas hanya dapat dijaga dengan menggunakan *username* dan *password* namun otentikasi tidak dapat dilakukan terus – menerus. Sebenarnya masalah ini dapat diselesaikan dengan memanfaatkan tanda tangan digital. Implementasi tanda tangan digital itu sendiri tidak boleh mengganggu kelebihan internet yaitu kemudahannya. Implementasi akan dilakukan pada jejaring sosial *facebook* dengan kasus pada saat melakukan *posting*.**

***Index Terms*— *facebook*, otentikasi, *posting*, tanda tangan digital**

I. PENDAHULUAN

Pada beberapa tahun terakhir, jejaring sosial merupakan tujuan utama sebagian besar pengguna internet misalnya *facebook*, *twitter*, dan *LinkedIn*. Fenomena pengguna jejaring sosial yang terus meningkat dengan drastis ternyata tidak didukung dengan peningkatan fitur keamanan jejaring sosial tersebut. Pada dasarnya, kelemahan pada fitur keamanan kebanyakan muncul karena faktor manusianya sendiri. Misalnya kebanyakan pengguna *facebook* dan *twitter* seringkali selalu memiliki akun mereka terbuka pada media tertentu. Tidak hanya itu, banyak aplikasi yang menyediakan kemudahan akses jejaring sosial setiap saat tanpa memperhatikan faktor keamanan pengguna.

Makalah ini akan berfokus pada jejaring sosial *facebook* karena menurut survey didapatkan bahwa pengguna *facebook* di Indonesia adalah yang ketiga terbesar di dunia[1]. Penggunaan *facebook* sendiri saat ini sangat beragam terutama di Indonesia misalnya digunakan untuk bertemu dengan teman lama, sebagai tempat dokumentasi citra/gambar, sampai dengan tindakan – tindakan yang dapat dinilai kreatif misalnya berjualan di *facebook* atau menggunakan *facebook* sebagai media komunikasi suatu mata kuliah. Dengan begitu luasnya fitur yang ditawarkan oleh *facebook*, sebaiknya ada suatu mekanisme khusus agar akun *facebook* kita tetap aman sekaligus nyaman.

Salah satu tindakan merugikan yang sering dilakukan beberapa bulan terakhir adalah kasus pembajakan akun *facebook*. Pembajakan disini dapat diartikan sebagai adanya suatu pihak yang menggunakan hak yang dimiliki oleh pemilik suatu akun tertentu untuk melakukan sesuatu dengan menggunakan akun tersebut dan membuat seolah – olah pemilik akun itu sendiri yang melakukannya. Tindakan ini mirip dengan skema *man-in-the-middle* tetapi pada kasus ini, penerima pesan berjumlah banyak, dapat juga dikatakan sebagai *broadcast message*.

Kasus pembajakan *facebook* sendiri tidak bisa dianggap ringan karena ternyata banyak orang yang menggunakan kelemahan ini untuk mencari keuntungan dengan merugikan orang lain[2]. Tidak sedikit juga pihak yang berbaik hati menyalurkan saran – saran mereka agar pembajakan *facebook* tidak menimpa[3]. Biarpun *facebook* akan menjadi lebih aman, ada pengorbanan pada sisi kenyamanan yang dikorbankan karena keamanan diperoleh dengan sedikit mempersulit proses yang biasanya dilakukan. Pengguna harus keluar dari kebiasaan, misalnya kebijakan untuk membuat *password* dengan gabungan huruf dan angka. Seringkali ada pengguna yang kesulitan mengingat *password* unik mreka tersebut dan tidak jarang yang lupa dan meninggalkan akunnya begitu saja.

Dengan memanfaatkan ilmu kriptografi sebenarnya kasus pembajakan ini memiliki banyak solusi. Salah satu solusiyang ditawarkan adalah *sign and verify*, sangat akrab dengan algoritma tanda tangan digital (*digital signature*). Implementasi tanda tangan digital memungkinkan pengguna dan pembaca memastikan hal – hal yang dilakukan memang dilakukan oleh pengguna dan pengguna tidak juga dapat mengelak lagi. Bagaimanapun, faktor kenyamanan juga jelas menjadi perhatian utama. Aplikasi tanda tangan digital ini tidak boleh merepotkan pengguna, lebih lagi mengharuskan pengguna mempelajari terlebih dahulu cara memakainya.

Aplikasi akan dikembangkan dengan memanfaatkan *facebook API*. Pada dasarnya, bagian terpenting mungkin bukan aplikasinya tersebut tetapi lebih ke bagaimana prosedur yang tepat dan instruksi – instruksi apa yang dapat menghindarkan dari sikap pembajakan. Dengan mendefinisikan dengan baik prosedur yang seharusnya dilakukan, jumlah pembajakan *facebook* diharapkan dapat diatasi dengan membedakan pesan mana yang diyakini ditulis oleh pengguna dan pihak lain yang tidak bertanggung jawab. *Digital Signature* ini memanfaatkan algoritma enkripsi RSA dan algoritma hash MD5. Dengan

memanfaatkan algoritma kunci publik pada tanda tangan digital, pengirim dan semua pembaca pesan dapat memastikan bahwa pesan memang ditulis oleh pengirim dan tidak diubah selama proses pengirimannya.

II. DASAR TEORI

A. Tanda Tangan Digital

Penggunaan tanda tangan pada kehidupan sehari – hari sudah tidak perlu lagi dijelaskan. Hampir setiap dokumen mulai dari dokumen akademik sampai dengan dokumen rahasia perusahaan, selalu terdapat tanda tangan biasanya pada bagian akhir. Kegunaan tanda tangan ini paling utama adalah menunjukkan bahwa pihak yang sudah membubuhkan tanda tangan telah menyetujui isi dokumen sekaligus membuat pihak yang menandatangani dokumen tidak bisa mengelak lagi karena tanda tangan setiap orang umumnya unik.

Konsep tanda tangan konvensional ini kemudian diadopsi untuk dokumen – dokumen yang sifatnya *softcopy*. Bagaimanapun, penggunaan tanda tangan berupa citra jelas tidak sesuai dengan konsep awal keberadaan tanda tangan. Permasalahannya, tanda tangan dalam bentuk citra dapat dengan mudah disalin dan ditempel pada berbagai dokumen, menghilangkan esensi dari keaslian suatu tanda tangan. Untuk mengatasi hal ini, diperlukan pendekatan lain dalam mengimplementasikan tanda tangan namun tetap memegang prinsip awal penggunaan tanda tangan itu sendiri.

Tanda tangan digital merupakan salah satu metode yang dapat dilakukan untuk mengatasi permasalahan adopsi tanda tangan konvensional untuk dokumen *softcopy*. Adapun prinsip yang harus dipertahankan adalah:

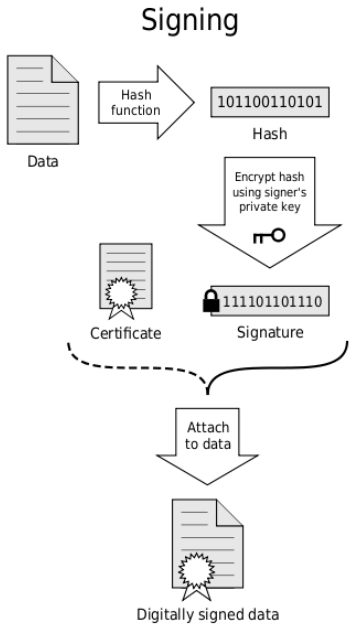
1. Tanda tangan tidak mudah dipalsukan dan sifatnya otentik.
2. Pemilik tanda tangan tidak dapat menyangkal bahwa ia telah menandatangani suatu dokumen.
3. Tanda tangan tidak mudah disalin atau digunakan ulang untuk dokumen lain.
4. Dokumen yang sudah diberikan tanda tangan tidak dapat diubah.

Sebuah tanda tangan digital dapat digunakan sebenarnya tidak hanya untuk sebuah dokumen. Tanda tangan digital dapat dimanfaatkan untuk berbagai hal yang lebih sederhana seperti suatu pesan singkat, atau hal yang lebih rumit daripada dokumen. Dokumen yang akan ditanda tangani juga tidak masalah jika dienkripsi maupun tidak dienkripsi.

Tanda tangan digital umumnya tidak menghasilkan suatu citra namun sebuah deret huruf yang tidak dapat dibaca atau tidak dapat dimengerti maksudnya. Panjang tanda tangan digital juga tidak ditentukan. Memang tidak ada aturan pasti dari suatu tanda tangan digital dan karena itu tanda tangan digital memiliki berbagai macam metode implementasi misalnya tanda tangan digital yang memanfaatkan enkripsi kunci simetris, ada pula yang memanfaatkan kriptografi kunci publik. Masing – masing dari macam metode implementasi tersebut memiliki keunikannya, kelebihan dan kekurangan yang berbeda – beda.

Pembahasan berikutnya akan menjelaskan mengenai tanda tangan digital dengan menggunakan algoritma kriptografi kunci publik karena implementasi yang digunakan pada makalah ini menggunakan algoritma tesebut. Tanda tangan digital dengan memanfaatkan kriptografi kunci publik memiliki beberapa komponen utama yaitu:

1. Pesan yang akan ditandatangani.
2. Fungsi Hash yang sudah ditentukan.
3. Algoritma kriptografi kunci publik yang sudah ditentukan.
4. Kunci private penulis pesan.
5. Kunci publik penulis pesan.
6. *Certificate* dari CA yang terpercaya.

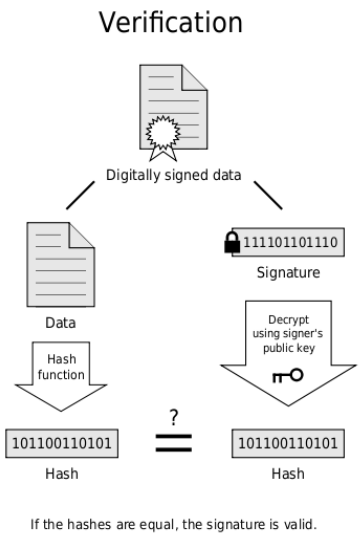


Gambar 1. Algoritma *sign* dengan memanfaatkan kriptografi kunci publik. (sumber: en.wikipedia.org)

Algoritma tanda tangan digital cukup sederhana dan dibagi menjadi dua tahap utama yaitu *sign* (menandatangani) dan *verify*(pengujian). Tahap pertama yaitu *sign* ditunjukkan oleh Gambar 1. Langkah – langkah dalam menandatangani adalah sebagai berikut:

1. Melakukan fungsi hash terhadap pesan menggunakan algoritma hash yang sudah ditentukan dan menghasilkan *message digest*.
2. *Message digest* kemudian dienkripsi dengan algoritma enkripsi tertentu menggunakan *private key* penulis pesan.
3. Hasil enkripsi beserta sertifikasi CA ditempelkan pada pesan.

Dengan demikian, setiap pengubahan pada dokumen akan menyebabkan tanda tangan tidak valid. Selain itu, *private key* hanya diketahui oleh pihak penulis pesan sehingga ia tidak dapat menyangkal telah menandatangani pesan tersebut. Disini, peran CA adalah sebagai penanda bahwa proses penandatanganan dapat dipercaya.



Gambar 2. Algoritma *verify* dengan memanfaatkan kriptografi kunci publik. (sumber: en.wikipedia.org)

Untuk algoritma *verify* merupakan algoritma *sign* yang urutannya dibalik. Lebih jelasnya adalah langkah pada tahap verifikasi adalah sebagai berikut:

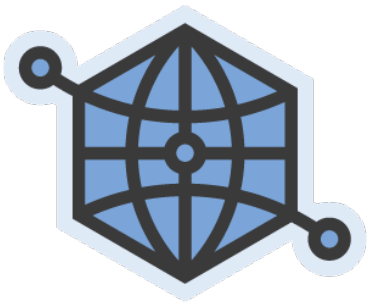
1. Melakukan fungsi hash berdasarkan algoritma hash yang sama dengan tahap *sign*(pihak yang melakukan verifikasi harus mengetahui algoritma hash yang digunakan oleh pihak yang melakukan penandatanganan).
2. Tanda tangan yang sudah ditempelkan pada pesan diambil dan didekripsi menggunakan kunci publik penulis pesan.
3. Membandingkan hasil dekripsi dengan *message digest* pada langkah 1. Tanda tangan dapat dinyatakan valid jika keduanya merupakan kumpulan karakter yang identik.

B. Facebook Platform

Jejaring sosial *facebook* menyediakan sejumlah API dan berbagai macam bantuan yang memudahkan bagi *developer* lain untuk mengakses dan menggunakan berbagai layanan yang berbasis *facebook*. Aplikasi yang dapat dikembangkan dengan berbasis *facebook* sungguh luas, mulai dari membangun aplikasi langsung di *canvas* yang disediakan oleh *facebook*, maupun hanya mengakses informasi pengguna dan menggunakan website selain *facebook.com*.

Integrasi antar *developer* juga difasilitasi dengan baik oleh *facebook* menggunakan protokol yang disebut *open graph*. Gambar 3 menunjukkan logo daripada *open graph protocol* milik *facebook*. Pada bulan Mei 2010, platform *facebook* menunjukkan statistik sebagai berikut:

1. Terdapat lebih dari satu juta *developer* yang berasal tidak kurang dari 180 negara.
2. Terdapat 550.000 aplikasi yang berjalan dengan berbasis *facebook*.
3. Lebih dari 250.000 website luar yang sudah berintegrasi dengan layanan *facebook*.
4. Seratus juta pengguna *facebook* mengakses website jejaring sosial ini melewati aplikasi/website selain yang disediakan resmi oleh *facebook*.



Gambar 3. Lambang *open graph protocol*.

Salah satu konsep inti yang akan diimplementasikan adalah fitur otentikasi yang disediakan oleh *facebook*. Melalui otentikasi ini pengguna *facebook* dapat memberikan *permission* kepada aplikasi/website lain untuk melakukan berbagai kegiatan dengan menggunakan akun pengguna tersebut. Hak akses yang harus diberikan adalah *publish_action*. Dengan hak akses ini, aplikasi lain dapat melakukan *posting* dan *notes*.

III. METODE

A. Analisis Masalah

Pada saat ini, skenario pengguna *facebook* yang ingin melakukan *posting* dengan kasus normal(melalui website *facebook*) adalah sebagai berikut:

1. Pengguna membuka website *facebook*.
2. Pengguna memasukkan *username* dan *password*.
3. Pengguna memasukkan post terbaru yang akan dikirimkan.
4. Post terkirim dan akan muncul di *wall* pengguna serta *newsfeed* mereka yang berhak melihat post pengguna tersebut.
5. Pengguna melakukan *log out*.

Jika dilihat pada skenario normal, proses pengiriman post tidak bermasalah pada identitas dan akan aman dari pengguna lain yang sebetulnya tidak berhak tetapi melakukan *posting* dengan menggunakan akun tersebut. Namun permasalahannya muncul jika langkah ke 5 disubtitusi dengan kegiatan lain, dan umumnya hampir setiap pengguna *facebook* tidak langsung melakukan *log out* setelah melakukan kegiatan. Hal ini kemungkinan disebabkan karena:

1. Pengguna lupa melakukan *log out*.

2. Pengguna mengakses *facebook* dalam frekuensi yang cukup sering sehingga proses *log in* dan *log out* akan memakan waktu dan melelahkan.
3. Pengguna memang sengaja membuat *facebook* mengingat akun mereka karena fitur seperti ini memang disediakan oleh *facebook*.

Berkaca pada alasan pengguna tidak melakukan *log out* dari *facebook* yang telah dijabarkan sebelumnya, terdapat celah yang menyebabkan muncul pihak lain yang menggunakan akun pengguna tersebut untuk melakukan *false posting* atau pengubahan post yang tidak benar. Umumnya, lebih buruk lagi, post yang baru ini bersfiat merugikan bagi pemilik asli akun dan tidak jarang juga merugikan pihak selain pemilik maupun pembajak (pihak lain yang melakukan *false posting*).

Melihat tren daripada penggunaan *facebook* untuk kegiatan – kegiatan yang sebetulnya bersifat formal dan suatu informasi bernilai penting juga untuk banyak orang, kasus pembajakan post ini bisa menyebabkan masalah besar jika tidak diselesaikan. Berkaitan dengan alasan – alasan pengguna tidak *log out*, masalah yang dapat muncul dapat disimpulkan sebagai munculnya informasi yang tidak tepat padahal informasi tersebut seringkali berhubungan dengan banyak orang/pengguna lain baik yang hanya membaca maupun sebenarnya terlibat secara tidak langsung dalam post yang sebenarnya hasil pembajakan.

Selain masalah keaslian suatu post, ada satu batasan yang perlu diperhatikan juga yaitu mengenai kemudahan. Keunggulan jejaring sosial *facebook* antara lain bahwa *sharing* dapat dilakukan dengan mudah. Kemudahan dapat menjadi batasan karena dalam proses penyelesaian masalah, sebisa mungkin solusi yang diberikan tidak mengorbankan faktor kemudahan yang sudah dimiliki pengguna *facebook*.

B. Desain Solusi

Masalah yang ingin diselesaikan sebenarnya cukup sederhana: bagaimana memastikan bahwa suatu informasi dari sebuah post memang diberikan oleh pemilik akun yang bersangkutan tanpa mengurangi kemudahan pemilik akun saat melakukan *posting*. Untuk menyelesaikan masalah tersebut, tentu saja terdapat berbagai macam solusi yang dapat diimplementasikan. Dengan melakukan pendekatan dari sisi kriptografi, salah satu solusi yang bisa diterapkan adalah dengan memanfaatkan skema tanda tangan digital.

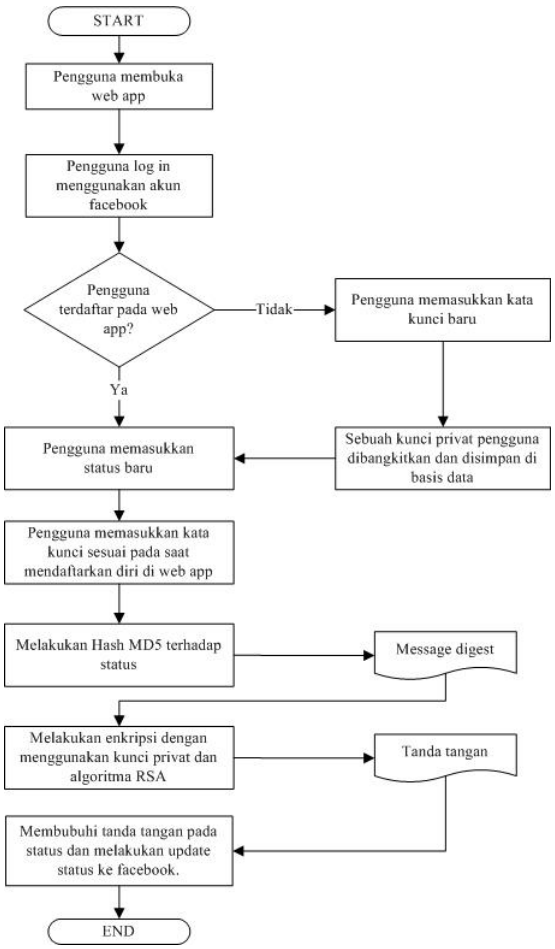
Aplikasi akan memanfaatkan tanda tangan digital untuk memberi tanda tangan di setiap *posting* yang dilakukan oleh pengguna. Untuk mengatasi batasan bahwa pengguna masih dapat melakukan *posting* dengan mudah, pengguna dapat memilih untuk membubuhkan tanda tangan atau tidak. Jika pengguna tidak ingin membubuhkan tanda tangan maka informasi yang disampaikan tidak dapat dipastikan merupakan asli diberikan oleh pemilik akun. Disini perlu diperhatikan bahwa jika tanda tangan tidak dibubuhkan bukan berarti informasi itu salah tetapi keasliannya hanya tidak dapat dipastikan.

Untuk fungsi hash yang digunakan adalah MD5 dengan alasan lebih mudah berkolaborasi dengan *facebook platform*. Kriptografi kunci publik dipilih sebagai metode enkripsi dan dekripsi karena diharapkan setiap orang yang membaca post yang sudah dibubuhi tanda tangan supaya dapat memeriksa keaslian isi post. Algoritma yang akan digunakan adalah RSA untuk mempermudah penyimpanan kunci. Dengan demikian, spesifikasi dari solusi yang akan dibangun adalah:

1. Sebuah *web app* yang terintegrasi dengan facebook.
2. Aplikasi menyediakan fungsi *sign* dan *verify*.
3. Algoritma Hash yang digunakan adalah MD5.
4. Algoritma enkripsi dan dekripsi yang digunakan adalah RSA.
5. Pengguna tetap diperkenankan melakukan *posting* tanpa membubuhi tanda tangan.

Untuk kegiatan *sign* sendiri, rancangan langkah – langkah pembubuhan tanda tangan ditunjukkan oleh Gambar 4. Poin yang perlu diperhatikan adalah terdapat sebuah kata kunci baru yang diperlukan oleh *web app*. Kata kunci ini diperlukan sebagai pemicu pemanggilan kunci privat pengguna yang disimpan pada basis data. Bagaimanapun, kesalahan pada saat melakukan *input* kata kunci bukan berarti pengguna tidak dapat melakukan *posting*, hanya saja kunci privat yang digunakan akan tidak bersesuaian dengan kunci privat seharusnya sehingga pada saat verifikasi akan menghasilkan tidak valid.

Dalam mengatasi keterhubungan antara kata kunci dengan kunci privat, solusi yang digunakan adalah dengan tidak menyimpan kata kunci pada basis data. Kunci privat dari RSA adalah sebuah angka yang sangat besar, kunci ini akan disimpan dalam basis data dan nilainya akan dikurangi sebuah angka yang diperoleh dari hasil bilangan acak dengan menggunakan kata kunci sebagai *seed* dari pembangkitan bilangan acak tersebut.



Gambar 4. Langkah – langkah pembubuhan tanda tangan(*sign*) pada post.

Mengenai kegiatan *verify*, permasalahan yang muncul adalah karena verifikasi bisa dilakukan oleh setiap orang yang membaca post, oleh karena itu harus dibuat sebuah skema supaya kunci publik penulis post, yang digunakan untuk melakukan dekripsi, harus dapat digunakan oleh pihak yang ingin melakukan verifikasi.

Poin penting disini adalah bahwa sebetulnya pihak yang akan melakukan verifikasi tidak perlu mengetahui dengan persis kunci publik dari penulis post, ia hanya harus dapat menggunakannya. Dari sini dapat ditarik kesimpulan bahwa seharusnya pembangkitan kunci privat dan publik dilakukan bersamaan dan keduanya disimpan pada basis data *web app*. Untuk kegiatan *verify*, pihak yang akan melakukan verifikasi juga tidak perlu terdaftar di *web app*. Langkah – langkah kegiatan *verify* lebih sederhana yaitu:

1. Pihak yang melakukan verifikasi membuka *web app*.
2. Memasukkan post yang ingin diverifikasi beserta tanda tangan yang sudah dibubuhi pada post tersebut.
3. Memasukkan *user id* dari pengguna *facebook* yang ingin ia verifikasi postnya.
4. Mendapatkan *feedback* apakah post tersebut memang betul ditulis oleh pemilik akun atau tidak dapat dipastikan/diragukan.

Sesuai pemaparan mengenai kegiatan *sign* dan *verify*, diperlukan sebuah basis data pada *web app* yang akan dibuat. Dapat disimpulkan hanya dibutuhkan satu buah tabel saja dan ditunjukkan oleh Tabel I.

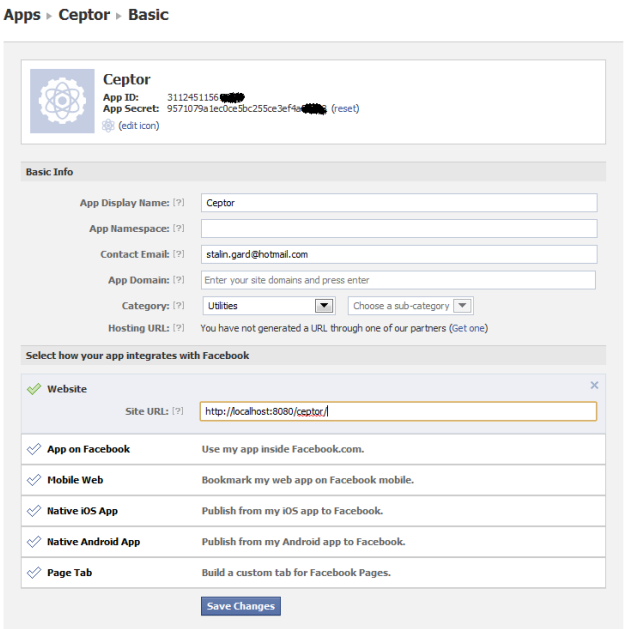
Tabel I. Tabel pengguna pada *web app*

Atribut	**Tipe**	**Keterangan**
id	INT	primary key
fbid	VARCHAR (20)	User ID facebook user
privkey	TEXT	hasil perhitungan private key sesungguhnya dikurangi kata kunci milik pengguna.
pubkey	TEXT	kunci publik dari pengguna.

IV. IMPLEMENTASI

A. Mendaftarkan Aplikasi di Facebook

Implementasi pertama kali dilakukan dengan mendaftarkan sebuah aplikasi baru di *facebook*. Untuk percobaan kali ini aplikasi diberi nama Ceptor dan karena dikembangkan pada alamat lokal maka *Site URL* menggunakan *localhost* seperti ditunjukkan oleh Gambar 5.



Gambar 5. Mendaftarkan aplikasi Ceptor di *facebook*

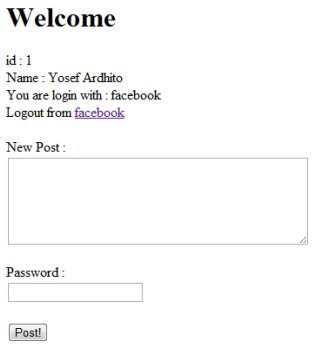
Dengan mendaftarkan aplikasi di *facebook*, berbagai API yang disediakan sudah dapat digunakan. Salah satu contoh hasil penggunaan *facebook* API untuk mendapatkan informasi basis milik pengguna yang dioper oleh *facebook* sebagai sebuah *array* sebagai berikut:

```
Array
(
    [id] => 1280818877
    [name] => Yosef Ardhito
    [first_name] => Yosef
    [last_name] => Ardhito
    [link] => http://www.facebook.com/yosefardhito
    [username] => yosefardhito
    [hometown] => Array
        (
            [id] => 102173726491792
            [name] => Jakarta, Indonesia
        )

    [quotes] => "You have absolutely nothing to
threaten me with.." - Joker
    [education] => Array
        (
           ...
        )

    [gender] => male
    [timezone] => 7
    [locale] => en_US
    [verified] => 1
    [updated_time] => 2012-03-13T10:33:37+0000
)
```

B. Halaman Utama

Halaman utama menyediakan tempat bagi pengguna untuk masuk ke *web app* menggunakan akun *facebook* seperti ditunjukkan oleh Gambar 6. Jika pengguna sudah memulai suatu *session* maka pengguna akan dihadapkan pada sebuah antarmuka yang menyediakan tempat memasukkan post baru dan kata kunci seperti ditunjukkan oleh Gambar 7.

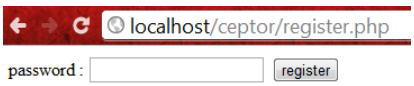
Gambar 6. Antarmuka halaman utama jika pengguna belum *log in*.



Gambar 7. Antarmuka halaman utama jika pengguna sudah *log in* dengan *facebook* dan pengguna sudah terdaftar di *web app*.

C. Halaman Registrasi

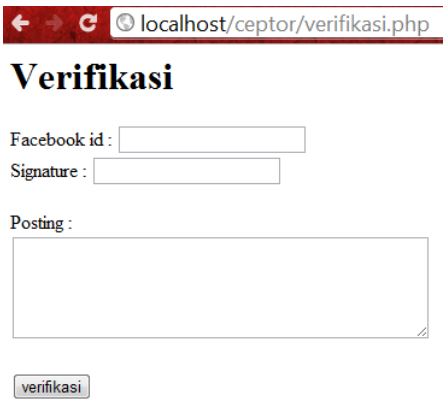
Untuk halaman registrasi hanya dibutuhkan satu buah field yang harus diisi yaitu kata kunci karena untuk *field* fbid(*facebook* id) didapatkan melalui fitur *facebook* API. Kata kunci ini sendiri tidak disimpan ke dalam basis data tetapi akan digunakan untuk memanipulasi *private key*. Antarmuka untuk hasil implementasi halaman registrasi ditunjukkan oleh Gambar 8.



Gambar 8. Antarmuka Ceptor halaman registrasi hanya terdiri dari input kata kunci saja

D. Halaman Verifikasi

Diperlukan tiga informasi untuk melakukan verifikasi yaitu *facebook id*, dan tentu saja post beserta *signature* yang sudah dibubuhi pada post tersebut. *Facebook id* dibutuhkan untuk melihat ke basis data *web app*, kunci publik daripada orang tertentu.

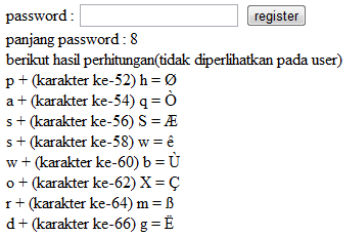


Gambar 9. Halaman verifikasi meminta tiga *input* dari pihak yang akan melakukan verifikasi dan tidak perlu *log in* terlebih dahulu.

V. PENGUJIAN

A. Mendaftar pada Web App

Misalkan mendaftarkan diri dengan kata kunci ‘password’ dan melakukan pembangkitan kunci publik dan kunci privat. Kunci privat selanjutnya dimodifikasi menggunakan kata ‘password’ sesuai ditunjukkan oleh Gambar 10. Data yang disimpan pada basis data ditunjukkan oleh Tabel II.



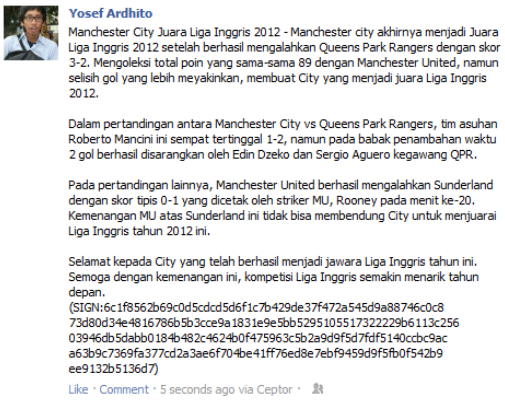
Gambar 10. Perhitungan kunci privat yang akan disimpan

Tabel II. Data yang disimpan pada basis data dengan karakter berwarna merah menunjukkan huruf yang diubah berdasarkan kata kunci pengguna

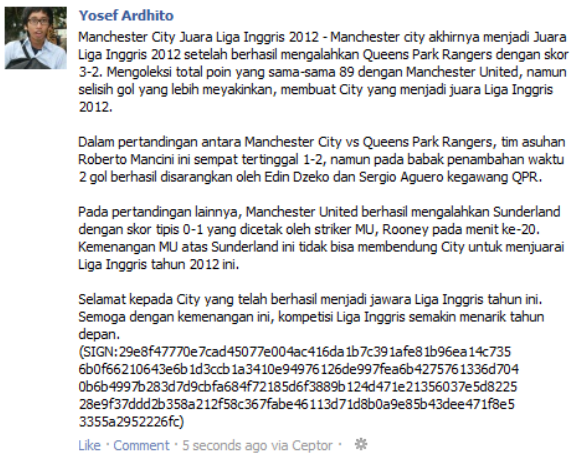
ID	1
FBID	1280818877
PRIVATE KEY	
-----BEGIN RSA PRIVATE KEY----- MIICXAIBAAKBgQCCcnI›tËkÆFÀJ°V½wå8ÑRPbQ3 8YqpezApWFb2M83dQgHApfvC61PuLKvphTvuO9w pkNW4fkTlTb4q3CSVVQh7fRgV0g8ky/xX6gwpf0 zgVh2mvDoFRbT4yqo0GTH6wWJ57SokgmZmb4kiZ VpbhIg3QQ5l4jIuuX4w2UMrib4TbiQIDAQABAoG AFe7eyS8T6PefHPol/BO0bItiWN+RFFfb3A4TPS 4Zj2pNEZkTqmJRuhxblX11fBwK/f7i2YFuQVplm TF7OrrfnDwg9DpUKsN5Slo/tXmrxcb3+i7kBgT0 jpK2+MX2zqtCNw9/btjvZ21BuBJWKNKrb61w/Th KxwCNQzE1F2SoYDUCQQCGscdkUg1YAYsUVMQp8O K1ydWKdQ2qewUQJefInXqGYIQKvwpIqRxSdTvEX S+EmK4rtcQjs6JmoqTT8rCB3GDdAkEA9+1q9ohv dqa8y1oHH70OVsF1B09j2b+v1fgDGNSNLZlb8w9 RCFDis3FdcS8aImKsTEYfp2P6cpuAvPDmZqIEnQ JBAIOzMJMW55U1bRfkCio4LnQxdWp8KXCX+MthR NETqcnadUFMqnGBaP/3QQkvVFcCMsrqX23JJvHT 9faHRa60HMECQHXos7YLrPcA1UjgyAKQJwGbf8y 6NQKZ50YbbV2huIK3LMOrxR3vWeF9SLNLc2JnCw o7T1KMec+HpPI2n1ywMQECQFdPHpU3jbqXNW0A6 3u82XLpeOuutEHoxZo0HDqMnW81oFbXMIUM4v+m telgtTUSSA9BG0LBqb1goW2KN58JhEg=----- END RSA PRIVATE KEY-----	
PUBLIC KEY	
-----BEGIN PUBLIC KEY----- MIGJAoGBAIJycj62ORIUwn1U3CzyZE9tDfxiql7 MClYVvYzzd1CAcCl+8LrU+4sq+mFO+473CmQ1 bh+ROVNvircJJVVCHt9GBXSDyTL/FfqDCl/TOBW Haa8OgVFtPjKqjQZMfrBYnntKiSCZmZviSJlW luEiDdBDmXiMi65fjDZQyuJvhNuJAgMBAAE= -- ---END PUBLIC KEY-----	

B. *Melakukan Posting*

Akan dilakukan dua buah percobaan yaitu dengan kata kunci benar dan kata kunci salah. Gambar 11 menunjukkan sebuah post yang dibuat menggunakan kata kunci yang benar yaitu ‘password’ sementara Gambar 12 menunjukkan hasil *posting* dengan kata kunci yang salah yaitu ‘katakunci’. Tabel III menunjukkan data yang digunakan untuk *posting*.



Gambar 11. Sebuah post yang dibuat menggunakan aplikasi Ceptor menggunakan kata kunci ‘password’



Gambar 12. Sebuah post yang dibuat menggunakan aplikasi Ceptor menggunakan kata kunci ‘katakunci’

Tabel III. Data yang digunakan untuk *posting* dengan kata kunci yang benar dan kata kunci yang salah

Post
Manchester City Juara Liga Inggris 2012 - Manchester city akhirnya menjadi Juara Liga Inggris 2012 setelah berhasil mengalahkan Queens Park Rangers dengan skor 3-2. Mengoleksi total poin yang sama-sama 89 dengan Manchester United, namun selisih gol yang lebih meyakinkan, membuat City yang menjadi juara Liga Inggris 2012. Dalam pertandingan antara Manchester City vs Queens Park Rangers, tim asuhan Roberto Mancini ini sempat tertinggal 1-2, namun pada babak penambahan waktu 2 gol berhasil disarangkan oleh Edin Dzeko dan Sergio Aguero kegawang QPR. Pada pertandingan lainnya, Manchester United berhasil mengalahkan Sunderland dengan skor tipis 0-1 yang dicetak oleh striker MU, Rooney pada menit ke-20. Kemenangan MU atas Sunderland ini tidak bisa membendung City untuk menjuarai Liga Inggris tahun 2012 ini. Selamat kepada City yang telah berhasil menjadi jawara Liga Inggris tahun ini. Semoga dengan kemenangan ini, kompetisi Liga Inggris semakin menarik tahun depan.
Hash (MD5)
3c2d84ae515260abb2cb5a8951e749af
Tanda Tangan Digital (RSA, kata kunci : password)
6c1f8562b69c0d5cdcd5d6f1c7b429de37f472a545d9a88746c0c873d80d34e4816786b5b3cce9a1831e9e5bb5295105517322229b6113c25603946db5dabb0184b482c4624b0f475963c5b2a9d9f5d7fdf5140ccbc9aca63b9c7369fa377cd2a3ae6f704be41ff76ed8e7ebf9459d9f5fb0f542b9ee9132b5136d7
Tanda Tangan Digital (RSA, kata kunci : katakunci)
29e8f47770e7cad45077e004ac416da1b7c391afe81b96ea14c7356b0f66210643e6b1d3ccb1a3410e94976126de997fea6b4275761336d7040b6b4997b283d7d9cbfa684f72185d6f3889b124d471e21356037e5d822528e9f37ddd2b358a212f58c367fabe46113d71d8b0a9e85b43dee471f8e53355a2952226fc

C. Melakukan Verifikasi

Verifikasi juga dilakukan sebanyak dua kali yaitu kasus kata kunci salah dan kata kunci benar. Untuk kata kunci salah ditunjukkan oleh Gambar 13 yang menunjukkan sebelum dan sesudah verifikasi, begitu juga dengan Gambar 14 menunjukkan kondisi kata kunci salah.

(a) (b)

Gambar 13. Verifikasi dengan signature hasil enkripsi dengan kata kunci benar. a) menunjukkan sebelum dilakukan verifikasi dan (b) menunjukkan hasil verifikasi.

(a) (b)

Gambar 14. Verifikasi dengan signature hasil enkripsi dengan kata kunci salah. (a) menunjukkan sebelum dilakukan verifikasi dan (b) menunjukkan hasil verifikasi.

V. ANALISIS HASIL PENGUJIAN

Dengan mengacu pada proses pengujian dan hasil yang didapatkan dapat dilakukan analisis yaitu penggunaan tanda tangan digital sebagai tindakan preventif memiliki beberapa karakteristik antara lain:

1. Proses pendaftaran(*registrasi*) berlangsung cukup lama terutama disebabkan proses pembangkitan kunci acak untuk algoritma RSA yaitu sebanyak 1024-bit.
2. Penggunaan kunci 1024-bit menyebabkan panjang pesan menjadi sepanjang kunci, hal ini menyebabkan penggunaan tanda tangan digital kurang efektif jika digunakan untuk pesan yang singkat.
3. Masih terdapat intervensi terhadap kemudahan *posting* yang awalnya sudah dimiliki pengguna yaitu dengan keberadaan kata kunci. Kata kunci diperlukan untuk menentukan kunci privat yang tepat terutama untuk proses verifikasi.
4. Kolaborasi antara penggunaan kata kunci dan kunci privat masih kurang baik. Ada kemungkinan hasil pengubahan kunci privat menyebabkan kunci privat menjadi tidak prima lagi dan tidak dapat digunakan untuk melakukan enkripsi. Sejauh ini, untuk mengatasi hal ini maka jika hasil dari kunci sudah bukan bilangan prima maka proses dihentikan karena pengguna jelas telah memasukkan kata kunci yang salah walaupun seharusnya pengguna tetap dapat melakukan *posting* dengan *sign* yang keliru sehingga verifikasi akan menyatakan bahwa pesan *unverified*.
5. Pengaburan dengan kata kunci masih sederhana dan kemungkinan besar bisa dideteksi byte daripada kunci privat yang diubah.
6. Integrasi dengan fungsi – fungsi yang diberikan oleh *facebook* masih bisa ditingkatkan terutama pada saat harus memasukkan *facebook id*. Seharusnya aplikasi yang menentukan *facebook id* daripada pengguna yang melakukan posting pesan tertentu.
7. Untuk jejaring sosial lain, skema ini dapat juga diimplementasikan tetapi akan kurang efektif jika digunakan untuk *posting* yang sifatnya singkat seperti misalnya jejaring sosial *twitter*.
8. Selain jejaring sosial, situs berita *online* adalah salah satu bidang lain yang dapat memanfaatkan skema tanda tangan digital sebagai penanda keaslian antara berita dan penulisnya.

DAFTAR REFERENSI

1. http://www.socialbakers.com (diakses 23 April 2012)
2. http://www.antaranews.com/berita/276991/mahasiswi-unsoed-jadi-korban-pembajakan-akun-facebook (diakses 23 April 2012)
3. http://wildancahyo.wordpress.com/2011/01/23/tips-mencegah-pembajakan-facebook/ (diakses 23 April 2012)
4. http://developers.facebook.com/ (diakses 11 Mei 2012)
5. http://rahmat-set.web.ugm.ac.id/web/04/tanda-tangan-digital (diakses 12 Mei 2012)
6. Munir, Rinaldi. 2006. Kriptografi. Bandung: Informatika

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, 14 Mei 2012

Yosef Ardhito Winatmoko - 13509052