Pengembangan Prototipe *Certification Authority* untuk Layanan *Short Message Service* pada *Platform* Android

Certification Authority Prototype Development for Short Message Service in Android Platform

Daniel Widya Suryanata dan Dr. Ir. Rinaldi Munir, M. T.
Departemen Teknik Informatika, Institut Teknologi Bandung, Bandung. Email: 13509083@std.stei.itb.ac.id
Departemen Teknik Informatika, Institut Teknologi Bandung, Bandung. Email: rinaldi@informatika.org

Abstrak- Pada saat ini, salah satu media yang secara luas digunakan untuk berkomunikasi adalah *Short Message Service* (SMS). SMS murah dan dapat digunakan pada ponsel apapun. Di sisi lain, penipuan via SMS juga meningkat, dan alasan utamanya adalah karena tidak adanya otorisasi pada SMS itu sendiri. Tanda tangan digital adalah sebua solusi untuk memastikan otentiksai SMS yang dikirimm, tetapi tanda tangan digital rentan terhadap serangan *man in the middle*. Satu cara untuk mengatasi hal ini adalah dengan mengimplementasikan infrastruktur kunci publik untuk SMS, dimana terdapat *certification authority* di dalam infrastruktur tersebut. *Certification authority* sederhana untuk SMS memungkinkan untuk diimplementasikan pada seluruh *platform* ponsel cerdas. Salah satu *platform* yang paling populer adalah Android yang menyediakan semua fungsi yang dibutuhkan untuk mengimplementasikan prototipe *certification authority*.
Kata kunci---*Certification Authority*, Infrastruktur Kunci Publik, Tanda Tangan Digital, *Short Message Service*, Android

***Abstract*- Nowadays, one widely used media for communicating is Short Message Service (SMS). SMS is cheap and can be used in any cell phone. Meanwhile, SMS fraud is increasing as well, and the main reason is because there is no authorization of the SMS itself. Digital signature is a solution to ensure sender's message authentication, but digital signature is prone to the man in the middle attack. One way to overcome this problem is by implementing public key infrastructure for SMS, which has certification authority element inside the infrastructure. Simple certification authority for SMS is possible to be implemented in all smartphone platforms. The most popular one is Android platform which support all functions needed for implementing a certification authority model.**
***Keywords*—Certification Authority, Public Key Infrastructure, Digital Signature, Short Message Service, Android**

I. PENDAHULUAN

SMS adalah media yang secara luas digunakan untuk berkomunikasi dan jumlah SMS yang dikirim meningkat dari tahun ke tahun. Sebanyak 9,8 triliun SMS dikirim pada tahun 2012 [1], jumlah ini meningkat dari tahun 2011 dimana SMS dikirim sebanyak 7,8 triliun [2]. Jumlah penipuan via SMS juga meningkat. Enam puluh Sembilan persen dari pengguna SMS mendapatkan *spam* atau SMS yang tidak diinginkan. Dengan jumlah yang tidak sedikit tersebut, sesuatu harus dilakukan untuk memastikan keotentikan pesan sehingga penipuan via SMS dapat dikurangi.

Tanda tangan digital dibuat dengan menandatangani hasil *hash* dari pesan menggunakan kunci privat pengirim. Kemudian penerima menggunakan kunci publik pengirim untuk melakukan verifikasi pesan. Tanda tangan digital adalah metode yang sangat baik untuk memastikan keotentikan pesan, karena jika ada pihak ketiga, yang ingin diakui sebagai pengirim yang sah, tidak memiliki kunci privat dari pengirim yang sah, pihak ketiga tersebut tidak akan diakui sebagai pengirim yang sah oleh pihak yang melakukan verifikasi.

Apabila tanda tangan digital hanyalah satu-satunya metode yang digunakan untuk melakukan verifikasi keotentikan pesan, metode tersebut akan rentan terhadap serangan *man in the middle*. Sebagai contoh, Alice ingin mengirim sebuah pesan kepada Bob dengan menggunakan tanda tangan digital di akhir pesan tersebut. Skenario normalnya adalah Alice menghitung nilai *hash* dari pesan tersebut kemudian Alice menandatangani (mengenkripsi) pesan tersebut menggunakan kunci privatnya. Ketika Bob menerima pesan Alice, Bob mengambil kunci publik Alice dan melakukan verifikasi (mendekripsi) tanda tangan digital Alice menggunakan kunci publik tersebut. Bob juga menghitung nilai hash dari pesan yang dikirim tanda tanda tangan digital tersebut. Apabila nilai keduanya sama, maka dapat disimpulkan bahwa pesan tersebut otentik.

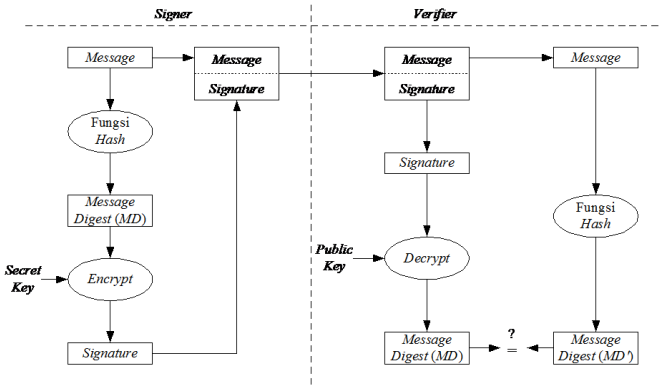
Tetapi bagaimana apabila Malory, seorang pihak ketiga, berpura-pura menjadi Alice? Yang harus ia lakukan hanyalah mendapatkan pesan yang dikirim oleh Alice, menghapus tanda tangan digital Alice, menandatangani pesan tersebut menggunakan kunci privatnya, dan mengirimnya kepada Bob. Terakhir, ia harus meyakinkan Bob bahwa kunci publiknya adalah kunci publik Alice. Bob tidak memiliki cara untuk memastikan keotentikan kunci publik Alice. Maka itu, *Certification Authority* (CA) menjadi sangat penting. CA melakukan verifikasi kunci publik pengirim sehingga tidak ada seorang pun dapat berpura-pura menjadi pengirim yang sah.

Ada beberapa makalah yang sudah dipublikasikan sebelumnya mengenai tanda tangan digital pada SMS. Salah satunya adalah makalah karya Putranto [3] yang membahas mengenai implementasi tanda tangan digital pada SMS. Namun seluruh makalah tersebut mengabaikan peranan CA. Makalah ini akan membahas mengenai CA aplikatif untuk SMS.

II. DASAR TEORI

A. Tanda Tangan Digital dan Sertifikat Digital

Tanda tangan digital bukanlah tanda tangan pengirim yang di-scan, melainkan nilai *hash* pesan yang dienkripsi menggunakan kunci privat pengirim dan ditempelkan di akhir pesan. Untuk melakukan verifikasi tanda tangan digital, penerima hanya perlu untuk mengambil tanda tangan digital yang terletak di akhir pesan, mendekripsinya dengan kunci publik pengirim, menghitung nilai *hash* dari pesan, dan mencocokan keduanya. Apabila hasil keduanya sama, maka tanda tangan digital tersebut bernilai valid. Skema detil tanda tangan digital dapat dilihat pada Gambar 1. Skema Tanda Tangan Digital.



Gambar 1. Skema Tanda Tangan Digital [4]

Kunci publik pengguna disimpan dalam bentuk sebuah sertifikat digital. Selain mengandung informasi mengenai kunci publik, sertifikat digital juga berisi informasi lain mengenai pengguna serta masa berlaku sertifikat digital tersebut.

B. *Certification Authority*

Certification Authority adalah suatu elemen dari infrastruktur kunci publik. Infrastruktur kunci publik dapat diartikan sebagai suatu infrastruktur yang memungkinkan penggunanya untuk membuat kunci publik, mempertukarkan, dan memverifikasi kunci publik orang lain untuk tujuan berkomunikasi dengan aman. Infrastruktur kunci publik memiliki beberapa tujuan, yaitu memastikan bahwa pengirim dan penerima adalah pengirim dan penerima yang seharusnnya berkomunikasi, bukan seseorang yang menyamar sebagai pengirim atau pun penerima. Tujuan kedua adalah untuk memastikan terjaganya integeritas data [5].

Sebuah infrastruktur kunci publik memiliki beberapa elemen [5] yaitu Certification Authority (CA) yang bertugas untuk membuat sertifikat digital, mengatur informasi status sertifikat digital dan mengeluarkan *Certificate Revocation List* (CRL), mempublikasikan sertifikat digital dan CRL, dan mengatur *archives*. Elemen kedua adalah Registration Authority yang mengurus pendaftaran pengguna yang akan menggunakan CA. Elemen ketiga adalah repositories yang merupakan suatu basis data untuk menyimpan semua sertifikat digital yang dikeluarkan CA. Elemen keempat adalah archives yang bertugas sebagai tempat penyimpanan jangka panjang. Elemen terakhir adalah pengguna infrastruktur kunci publik yang terbagi menjadi subjek dan *relying party*.

III. ANALISIS

Sebelum merancang perangkat lunak yang memenuhi syarat sebuah CA, berbagai analisis perlu dilakukan. Perlu dipertimbangkan pemilihan algoritma tanda tangan digital. Algoritma tersebut harus menghasilkan tanda tangan digital yang cukup panjang sehingga cukup aman untuk digunakan namun cukup pendek agar tidak memakan tempat terlalu banyak pada SMS. Penyimpanan kunci privat oleh pengguna juga perlu dipertimbangkan. Kunci privat tersebut sebaiknya diletakan di tempat yang sulit untuk dijangkau oleh pengguna untuk memberikan pengamanan lebih kepada kunci privat. Analisis lain adalah bagaimana menyebarkan kunci publik pada pengguna lainnya. Usia dari pasangan kunci pengguna dan CA juga perlu diperhitungkan. Usia tersebut sebaiknya tidak terlalu singkat dan tidak pula terlalu panjang. Apabila usia pasangan kunci terlalu panjang, maka pasangan kunci tersebut akan relatif lebih tidak aman, sedangkan apabila usia pasangan kunci terlalu pendek, proses *revocation* akan lebih sering terjadi. Pasangan kunci CA memiliki usia yang lebih panjang dibandingkan usia pasangan kunci pengguna, hal ini dikarenakan pentingnya peran pasangan kunci CA, yaitu untuk menandatangani seluruh sertifikat digital pengguna. Apabila sertifikat CA memiliki usia yang singkat, maka seluruh sertifikat digital pengguna yang ditandatangani oleh kunci privat CA tersebut harus di-*revoke* dan hal ini membutuhkan biaya yang besar apabila jumlah sertifikat digital pengguna besar.

Untuk menjawab analisis tersebut, pada perancangan perangkat lunak ini digunakan algoritma RSA 512 bit. Algoritma tersebut dipilih karena tanda tangan digital yang dihasilkan akan dapat dimasukan ke dalam 1 pesan SMS. Algoritma untuk mencari nilai *hash* yang digunakan adalah algoritma SHA-1. Kunci privat pengguna disimpan pada tempat penyimpanan internal Android yang sulit dijangkau oleh pengguna. Kunci publik pengguna akan disebarkan dengan cara direktori yang dapat diakses oleh semua pengguna, namun pengguna tidak dapat melihat seluruh sertifkat digital pengguna yang ada di *repository*, melainkan hanya dapat mengakses sertifikat digital pengirim pesan saja. Hal ini bertujuan untuk melindungi kerahasiaan nomor telepon pengguna lainnya. Usia pasangan kunci pengguna adalah 1 tahun sedangkan usia pasangan kunci CA adalah 5 tahun.

IV. SISTEM YANG DIUSULKAN

CA yang dibangun terdiri dari 2 aplikasi yang berbeda, yaitu aplikasi *client* dan aplikasi *server*.

A. Rancangan Aplikasi *Client*

Aplikasi *client* berbentuk aplikasi Android dan pemakai dari aplikasi *client* ini hanya pengguna saja. Pada aplikasi *client*, pengguna dapat membuat sertifikat digital, memeriksa keabsahan sertifikat digital serta tanda tangan digital yang terkandung di dalam SMS, serta menandatangani pesan.

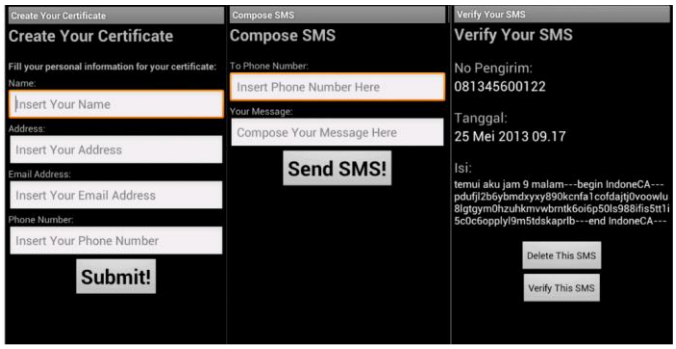
Pada menu membuat sertifikat digital, pengguna harus memasukkan informasi yang berupa nama pengguna, alamat pengguna, alamat *email* pengguna, serta nomor HP pengguna. Sertifikat digital unik berdasarkan nomor HP, maka apabila ada nomor HP yang sama yang sudah tersimpan pada basis data *server*, permintaan pembuatan sertifikat digital tersebut akan ditolak. Apabila seluruh informasi tersebut memenuhi format yang benar, data tersebut akan dikirim pada *server* dan nilai kunci privat pengguna akan disimpan ke dalam penyimpanan internal Android.

Pada menu memeriksa keabsahan sertifikat digital, pengguna dapat memeriksa keabsahan dari SMS yang telah diterima oleh pengguna. SMS tersebut disimpan ke dalam basis data yang ada pada aplikasi *client*. Berikut adalah seluruh kemungkinan kasus untuk hasil verifikasi suatu SMS:

1. Sertifikat digital pengirim tidak terdaftar pada *server*. Hal ini terjadi apabila SMS tersebut berasal dari nomor yang belum memiliki sertifikat digital resmi yang terdaftar pada basis data *server*.
2. SMS tidak mengandung tanda tangan. Hal ini terjadi apabila sertifikat digital pengirim dikenali namun SMS tersebut tidak mengandung tanda tangan digital atau tanda tangan digital tidak benar formatnya.
3. Sertifikat digital pengguna telah di-*revoke*. Hal ini terjadi apabila sertifikat digital pengguna terdaftar pada *server* dan SMS mengandung tanda tanda tangan digital, namun sertifikat digital pengirim telah di-*revoke* oleh *server* sehingga menyebabkan tanda tangan digital tersebut tidak valid.
4. Sertifikat digital CA tidak valid. Hal ini terjadi apabila sertifikat digital pengguna terdaftar pada *server*, SMS mengandung tanda tangan, dan sertifikat digital penggna belum di-*revoke*. Namun Sertifikat digital CA itu sendiri tidak valid. Sehingga menyebabkan sertifikat digital pengguna dan tanda tangan digital pada SMS tidak valid.
5. Sertifikat digital pengirim tidak valid. Hal ini terjadi apabila sertifikat digital pengguna terdaftar pada *server*, SMS mengandung tanda tangan, sertifikat digital penggna belum di-*revoke*, sertifikat digital CA valid, namun sertifikat digital pengguna tidak valid apabila dicocokan menggunakan sertifikat digital CA. Hal ini mengakibatkan tanda tangan yang terkandung dalam SMS menjadi tidak valid.
6. Tanda tangan digital pada SMS tidak valid. Hal ini terjadi apabila sertifikat digital pengguna terdaftar pada *server*, SMS mengandung tanda tangan, sertifikat digital penggna belum di-*revoke*, sertifikat digital CA valid, dan sertifikat digital pengguna juga valid, namun nilai dari tanda tangan digital tersebut tidak valid apabila dicocokan menggunakan sertifikat digital pengguna.
7. Tanda tangan digital pada SMS valid. Hal ini terjadi apabila keenam hal diatas tidak terjadi, yang berarti pesan tersebut otentik.

Pada menu menandatangani pesan, pengguna diminta untuk mengetikkan nomor HP penerima yang dapat dipisahkan dengan tanda koma apabila pengguna ingin mengirim ke lebih dari satu penerima. Pengguna juga diminta untuk mengetikkan pesan. Setelah itu aplikasi *client* akan secara otomatis menambahkan tanda tangan digital pengguna pada akhir SMS dengan menggunakan kunci privat pengguna.

Gambar 2 menunjukkan tampilan beberapa layar dari aplikasi *client*, yaitu tampilan pembuatan sertifikat digital (gambar kiri), penandatanganan pesan (gambar tengah), dan verifikasi tanda tangan digital (gambar kanan).



Gambar 2. Tampilan Aplikasi *Client*

B. Rancangan Aplikasi *Server*

Aplikasi *server* berbentuk *website* dan pemakai dari aplikasi *server* ini adalah pengguna dan administrator. Pada aplikasi *server*, pengguna dapat melakukan *log in* untuk mengubah *password* pengguna pada *website* CA, atau pun melakukan permintaan *revocation* sertifikat digitalnya kepada administrator. *Username* dan *password* pengguna untuk masuk ke dalam *website* diberikan saat pengguna membuat sertifikat digital dengan cara dikirimkan via *email*.

Untuk mengganti *password*, pengguna hanya perlu memasukkan *password* lama, *password* baru, dan sekali lagi memasukkan *password* baru, apabila *password* lama yang dimasukkan benar dan *password* baru sama dengan *password* baru yang dimasukkan sekali lagi, maka *password* akan sukses diganti. Sedangkan untuk mengirimkan permintaan *revocation* kepada administrator, pengguna hanya perlu mengisi kolom alasan mengapa ingin diadakan *revocation* setelah itu *revocation* akan dilakukan.

Pada aplikasi *server*, administrator dapat melihat permintaan *revocation* sertifikat digital yang dikirim oleh pengguna. Setelah melihat permintaan *revocation*, administrator berhak untuk me-*revoke* sertifikat digital pengguna. Pada menu ini, administrator cukup menekan tombol '*Revoke'* untuk melakukan *revocation* sertifikat digital pengguna. Sebuah

email secara otomatis akan dikirim pada pengguna untuk mengabarkan bahwa *revocation* berhasil dilakukan.

Selain itu, administrator juga dapat melakukan *revocation* sertifikat digital CA. Hal ini dapat dilakukan apabila administrator merasa sertifikat digital CA sudah tidak aman lagi atau pun karena hilangnya kunci privat CA. Sama seperti *revocation* sertifikat digital pengguna, administrator hanya perlu menekan tombol '*Revoke* Sertifikat Digital CA' untuk melakukan *revocation* sertifikat digital CA. Apabila dilakukan *revocation* sertifikat digital CA, seluruh sertifikat digital pengguna yang ditandatangani oleh sertifikat digital ini akan di-*revoke* dan sebuah *email* akan dikirim pada pengguna untuk mengabarkan hal ini. Setelah *revocation* sertifikat digital CA dilakukan, pembuatan sertifikat digital CA yang baru akan otomatis dilakukan. Tampilan layar utama aplikasi *server* dapat dilihat pada Gambar 3.

Gambar 3. Tampilan Aplikasi *Server*

V. IMPLEMENTASI DAN PENGUJIAN

Implementasi dilakukan dengan menggunakan bahasa Java untuk aplikasi *client* serta bahasa HTML dan PHP untuk aplikasi *server*. Versi minimum SDK dari Android yang harus dimiliki oleh aplikasi client adalah versi 10. *Framework* PHP yang dipakai adalah CodeIgniter_2.1.3.

Beberapa pengujian dilakukan terhadap perangkat lunak yang dibangun. Pengujian-pengujian tersebut dilakukan untuk menguji keamanan dari perangkat lunak. Pengujian dibedakan menjadi 2, yaitu pengujian pada aplikasi *server* dan pengujian pada aplikasi *client*.

Pada aplikasi server, dilakukan pengujian SQL *injection* dan *Cross Site Scripting* (XSS). Pengujian ini dilakukan untuk melihat masukan pengguna pada kolom yang dapat diisi pada aplikasi *server*, yaitu pada menu pengguna. SQL *injection* diuji coba pada saat *log in* dan pada saat pengguna mengubah *password*. Sedangkan XSS diuji coba saat pengguna mengubah *password* dan memasukkan alasan *revocation* sertifikat digital.

Sedangkan pada aplikasi *client*, pengujian keamanan dilakukan dengan menguji masukan yang dimasukan pengguna pada kolom-kolom yang dapat diisi yaitu pada menu pembuatan sertifikat dan penandatanganan pesan. Pengguna harus memasukkan alamat *email* dan nomor HP yang valid.

Contoh dari pesan yang diberi tanda tangan digital dapat dilihat pada Tabel 1.

Tabel 1. Uji Penambahan Tanda Tangan Digital

Pesan Awal	halo, ini pengujian
Nilai Kunci Privat	594962774155238278184157232634503324680115966657918046580643394231747932737722410250551209175235515823909159571172789174435418163241380805021694314546778464327717 47
Nilai N	27783410110835775863365072252658963277330745879003995412029325446294255795781952279761111678912809415080146127991524394650037723451246909436820073657160452532389579 21
Pesan Akhir	halo, ini pengujian---begin IndoneCA---1w21hoj8mpxab6woy9f0rhxunv5knxw3jvgapanyd0ckypx325hf81xofixittgorpkpzq9hb9yk1796sh4g60li0w2y2roiqggonvvej96---end IndoneCA---

Selain pengujian tersebut, dilakukan beberapa pengujian. Pengujian lainnya meliputi waktu yang dibutuhkan oleh aplikasi CA *client* untuk membuat sertifikat digital, menandatangani pesan, serta melakukan verifikasi pesan. Pengujian ini dilakukan terhadap beberapa ponsel Android yang memiliki spesifikasi berbeda untuk melihat performa dari masing-masing ponsel tersebut.

Pengujian performansi tersebut dilakukan pada 2 jenis ponsel Android, yaitu:

1. Samsung Galaxy 5 I5500 yang memiliki spesifikasi versi Android 2.1, RAM 256 MB, dan kecepatan CPU sebesar 600 MHz.
2. Samsung Galaxy Gio S5660 yang memiliki spesifikasi versi Android 2.3.4, RAM 278 MB, dan kecepatan CPU sebesar 800 MHz.
3. Samsung Galaxy Tab 7.0 yang memiliki sepesifikasi versi Android 3.2, RAM 1 GB, dan kecepatan CPU 1,2 GHz.
4. Samsung Galaxy S3 I9300 yang memiliki spesifikasi versi Android 4.0.4, RAM 1 GB, dan kecepatan CPU sebesar 1,4 GHz.
5. Asus TF300TG yang memiliki spesifikasi versi Android 4.0, RAM 1 GB, dan kecepatan CPU 1,2 GHz.

Tabel 2 menunjukkan hasil pengujian pada pembuatan pasangan kunci pengguna. Dilakukan tiga kali pengulangan pada pengujian ini dan hasil dari pengujian ini adalah jenis ponsel tidak menentukan lamanya waktu pembuatan kunci. Hal ini dikarenakan bilangan yang dibangkitkan bersifat acak sehingga apabila nilai bilangan acak tersebut besar, waktu komputasi yang dibutuhkan juga akan semakin besar.

Hasil pengujian performansi penandatanganan pesan ditunjukkan pada Tabel 3, sementara pengujian performansi verifikasi tanda tangan digital ditunjukkan pada Tabel 4. Pada kedua uji performansi tersebut, jenis ponsel sangat menentukan waktu yang dibutuhkan untuk melakukan kedua hal tersebut. Waktu yang dibutuhkan oleh Samsung Galaxy S3 lebih sedikit disbanding waktu yang dibutuhkan oleh Samsung galaxy Gio.

Pada pengujian panjang SMS yang dihasilkan setelah tanda tangan digital ditambahkan, nilai kunci privat dan nilai N pengguna memengaruhi panjang pesan yang dihasilkan seperti

yang diperlihatkan pada Tabel 6 dan Tabel 7. Dapat dipastikan bahwa panjang tanda tangan digital yang ditambahkan pada akhir pesan SMS tidak akan melebihi 1 SMS.

Berikut adalah pesan yang digunakan dalam pengujian ini:

1. Pesan 1 (pada Tabel 5): "halo".
2. Pesan 2 (pada Tabel 6): "Pemberitahuan, besok tolong berkumpul jam 6 tepat, terima kasih".
3. Pesan 3 (pada Tabel 7): "kriptografi dapat diartikan sebuah studi mengenai teknik matematis yang berkaitan dengan aspek keamanan informasi seperti kerahasiaan, integritas data, otentikasi, dan nir penyangkalan".

Tabel 2. Uji Pembuatan Pasangan Kunci

Jenis HP	Waktu 1 (ms)	Waktu 2 (ms)	Waktu 3 (ms)
Samsung Galaxy 5	859	2092	778
Samsung Galaxy Gio	668	1600	722
Samsung Galaxy Tab 7.0	576	447	159
Samsung Galaxy S3	828	427	405
Asus TF300TG	1282	212	286

Tabel 3. Uji Penandatanganan Pesan

Pesan	Waktu (ms)				
	Samsung Galaxy 5	Samsung Galaxy Gio	Samsung Galaxy Tab 7.0	Samsung Galaxy S3	Asus TF300TG
halo	836	799	481	325	180
Pemberitahuan, besok tolong berkumpul jam 6 tepat, terima kasih	1631	1427	735	398	376

Tabel 4. Uji Verifikasi Tanda Tangan Digital

Pesan	Waktu (ms)				
	Samsung Galaxy 5	Samsung Galaxy Gio	Samsung Galaxy Tab 7.0	Samsung Galaxy S3	Asus TF300TG
halo + (tanda tangan digital)	1015	830	276	175	232
Pemberitahuan, besok tolong berkumpul jam 6 tepat, terima kasih + (tanda tangan digital)	1865	1487	393	317	436

Nilai kunci privat yang digunakan pada Tabel 5, Tabel 6, dan Tabel 7 adalah:

1. Kunci Privat 1:
   1734541758185147634309819727107359290043872143742617681347806362292746329621995319120691612068183967515491012086738915926687657520763825266585387721550446000131201947.
2. Kunci Privat 2:
   1701038428645694856860733675202526582277058328574049723019820903851514827106709226133816556829456257629121394831435503788966656407441775879155225838159413026931634 1.
3. Kunci Privat 3:
   2911580587813640258380574036420435958233862331427503040302244518469259071245027745444689920368882113684259051954568700911357787364021796518723876131344276496936068945.

Sedangkan nilai N yang digunakan adalah:

1. N1:
   2781199815482518754062467756837729957904594297194300973700634173888596800469575243049533185040109279025859064604113956009562492595067484718673013971286771688388513349.
2. N2:
   3347215340631929894956521058098879835057009298065645526544438301106966269338429823102885576197164240994734031232920698100695153776158566218210549855622289958174572067.
3. N3:
   3046118536195379857081820256430345915351956071557246636395532289329657596944087510182032540677728265572407203863335864806383646681938663732412805499649735047119892357.

Tabel 5. Uji Panjang SMS Pesan 1

Panjang Sebelum	Jumlah SMS Sebelum	Panjang Sesudah	Jumlah SMS Sesudah	Kunci Privat	N
4	1	149	1	Kunci Privat 1	N1
4	1	149	1	Kunci Privat 2	N2
4	1	149	1	Kunci Privat 3	N3

Tabel 6. Uji Panjang SMS Pesan 2

Panjang Sebelum	Jumlah SMS Sebelum	Panjang Sesudah	Jumlah SMS Sesudah	Kunci Privat	N
63	1	208	2	Kunci Privat 1	N1
63	1	208	2	Kunci Privat 2	N2

63	1	207	2	Kunci Privat 3	N3

Tabel 7. Uji Panjang SMS Pesan 3

Panjang Sebelum	Jumlah SMS Sebelum	Panjang Sesudah	Jumlah SMS Sesudah	Kunci Privat	N
184	2	329	2	Kunci Privat 1	N1
184	2	329	2	Kunci Privat 2	N2
184	2	328	2	Kunci Privat 3	N3

KESIMPULAN DAN PENGEMBANGAN

Model *Certification Authority* tidak hanya dapat diimplementasikan pada *website* saja, namun juga pada SMS. *Certification Authority* tersebut dapat diimplementasikan dengan aplikasi *client-server*. Aplikasi Android menjadi aplikasi *client* yang dipakai oleh pengguna dan berfungsi untuk membuat sertifikat digital, memastikan keabsahan sertifikat digital CA dan pengguna, serta menandatangani SMS. Sedangkan *website* menjadi aplikasi *server* dan befungsi untuk melakukan *revocation* sertifikat digital pengguna dan CA, membuat sertifikat CA, serta sebagai *repository* CA.

Tanda tangan digital cocok untuk SMS. Hal ini dibuktikan dengan pemrosesan yang tidak memakan banyak waktu pada saat membuat sertifikat digital, menandatangani pesan, maupun melakukan verifikasi pesan. Selain itu tanda tangan digital yang tidak akan melebihi 1 SMS juga merupakan *trade off* yang sesuai.

Untuk pengembangan selanjutnya, akan sangat baik apabila:

1. *Registration Authority* diimplementasikan karena pada makalah ini, seluruh masukan identitas pengguna dianggap valid, sedangkan infrastruktur kunci publik yang ideal melakukan validasi terhadap identitas pengguna.
2. SMS yang diterima sebelum pengguna meng-*install* aplikasi *client* sebaiknya juga dimasukan ke dalam basis data *client* agar dapat diverifikasi.
3. Komunikasi antara *client* dan *server* hendaknya memiliki tingkat keamanan yang lebih tinggi. Hal ini dapat diimplementasikan dengan melakukan enkripsi terhadap pesan yang dikirim.
4. Algoritma tanda tangan digital yang lebih efisien, seperti *Elliptic Curve Digital Signature Algorithm*, hendaknya dipakai. Hal ini bertujuan untuk memperpendek kunci pengguna.

UCAPAN TERIMA KASIH

Terima kasih kepada Tuhan Yang Maha Esa karena berkat bantuan dan rahmat-Nya lah makalah ini dapat diselesaikan.

Terima kasih kepada Dr. Ir. Rinaldi Munir, M. T. selaku pembimbing penulis dalam penulisan makalah ini. Penulis juga berterima kasih kepada seluruh dosen, tata usaha, dan karyawan Departemen Teknik Informatika Institut Teknologi Bandung yang telah secara langsung maupun tidak langsung membantu penulisan makalah ini.

Penulis juga berterima kasih pada teman-teman penulis yang telah bersedia penulis tanyai dalam pembuatan aplikasi CA ini, yaitu Edwin Lunando, Georgius Rinaldo, Hartono Sulaiman Wijaya, dan Irvan Jahja.

Terakhir, penulis mengucapkan terima kasih kepada seluruh pihak yang telah membantu penulis dalam penulisan makalah dan pengerjaan aplikasi ini yang namanya tidak dapat disebut satu per satu.

DAFTAR PUSTAKA

[1] CMO Council, “Facts Tagged With SMS”. Diperoleh 16 Juli 2013, dari http://www.factbrowser.com/tags/sms/, November 2012.

[2] Pew Research Center, “Facts Tagged With SMS”. Diperoleh 16 Juli 2013, dari http://www.factbrowser.com/tags/sms/, Agustus 2012.

[3] Putranto, A Kurniawan Dwi, “Penerapan Digital Signature pada Aplikasi SMS Android”, dari http://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2011-2012/Makalah2-2012.htm, Mei 2012.

[4] Munir, R, “Diklat Kuliah IF3058, Kriptografi”, penerbit Informatika, Bandung, 2009.

[5] Kuhn, D R, Hu, V C, Polk, W T, Chang, S J, “Introduction to Public Key Technology and the Federal PKI Structure”, dari http://www.nist.gov, Februari 2001.