Studi dan Implementasi Kriptografi Kunci-Publik untuk Otentikasi Perangkat dan Pengguna pada Komunikasi Bluetooth

Made Harta Dwijaksara [1)]
Laboratorium Ilmu dan Rekayasa Komputasi
1) Program Studi Teknik Informatika, ITB, Bandung 40132, email: made.harta@gmail.com

Abstraksi – *Pada makalah ini akan dibahas mengenai mekanisme otentikasi yang dilakukan pada suatu komunikasi Bluetooth. mulai dari metode yang sudah diimplementasikan saat ini sampai dengan metode yang diusulkan untuk meningkatkan keamanan proses otentikasi. Mekanisme otentikasi yang sudah diimplementasikan pada komunikasi Bluetooth (yang berlandaskan kriptografi kunci simetri) memiliki celah keamanan karena data yang dipertukarkan pada saat mekanisme otentikasi dijalankan dapat digunakan untuk menerka PIN. Disamping itu, belum diaplikasikannya mekanisme otentikasi pengguna juga semakin memudahkan penyerang untuk melakukan serangan guna mendapatkan data yang dipertukarkan. Maka dari itu akan dibangun suatu protokol otentikasi baru untuk komunikasi Bluetooth. Kriptografi kunci publik digunakan untuk membangun protokol otentikas yang terdiri dari protokol otentikasi pengguna dan perangkat. Untuk otentikasi pengguna memanfaatkan prinsip Challenge and Response, dengan tujuan untuk memastikan bahwa data kunci privat yang digunakan oleh pihak yang diajak berkomunikasi merupakan pasangan salah satu data kunci publik yang telah dimiliki. Selanjutnya untuk otentikasi perangkat digunakan mekanisme pertukaran kunci menggunakan protokol Diffie-Hellman dengan penambahan digital signature untuk menghindari man-in-the-midle-attack. Tujuan utama dari mekanisme otentikasi perangkat ini adalah untuk mempertukarkan kunci rahasia dengan pihak yang otentik.*

Kata Kunci: *Bluetooth, kriptografi kunci-publik, Challenge and Response, protokol Diffie-Hellman, digital signature, man-in-the-midle-attack.*

1. PENDAHULUAN

Dewasa ini keamanan sistem komunikasi menjadi tuntutan yang harus dipenuhi oleh semua pihak yang terlibat didalamnya, apalagi komunikasi digital yang sangat rentan terhadap praktik-praktik penyadapan. Maka dari itu segala bentuk upaya digunakan untuk meningkatkan aspek keamanan pada suatu sistem komunikasi. Salah satu sistem komunikasi yang sampai saat ini masih terus ditingkatkan aspek keamanannya adalah komunikasi menggunakan media *Bluetooth*. *Bluetooth*, salah satu standar komunikasi lokal tanpa kabel (*local wireless communication*) [SUN05], merupakan media komunikasi yang semakin populer dewasa ini. Kepopuleran ini didorong oleh banyaknya *vendor* perangkat *mobile* yang menyertakan *Bluetooth* pada produk mereka sebagai salah satu fasilitas pengiriman data. Kemudahan dalam penggunaan mengakibatkan banyak pengguna menggunakan fasilitas ini.

Sistem komunikasi yang terbilang relatif baru ini jika dibandingkan dengan standar komunikasi tanpa kabel lain seperti *wireless fidelity* (WiFi) dan *infrared*, masih memiliki kelemahan-kelemahan yang bisa dimanfaatkan oleh pihak yang tidak berhak untuk melakukan penyusupan ataupun penyadapan. Dua diantara kelemahan tersebut dijelaskan dibawah ini.

Kelemahan pertama terletak pada mekanisme pertukaran kunci untuk proses otentikasi perangkat. Pada mekanisme ini dipertukarkan data yang bisa dimanfaatkan untuk melakukan penyerangan guna mendapatkan *Personal Identity Number* (PIN) yang digunakan dalam komunikasi. Kelemahan yang terdapat pada algoritma pembangkitan kunci (algoritma SAFER+) akan mempercepat pemecahan PIN oleh pihak penyerang. Seketika PIN berhasil didapatkan maka seluruh komunikasi oleh perangkat tersebut sudah tidak aman lagi.

Kelemahan kedua adalah pada komunikasi *Bluetooth* belum terdapat otentikasi terhadap pengguna, proses otentikasi hanya dilakukan untuk perangkat, sehingga jika perangkat yang berkomunikasi sama maka dianggap proses komunikasi dilakukan dengan pengguna yang benar.

2. KRIPTOGRAFI KUNCI-PUBLIK

Kriptografi adalah ilmu/seni penyadian pesan ke dalam bentuk yang tidak dipahami oleh orang lain. Kriptografi kunci-publik menggunakan sepasang kunci, satu kunci untuk enkripsi dan satu kunci untuk dekripsi [MUN06]. Kunci untuk enkripsi bersifat publik (tidak rahasia) sehingga dinamakan kunci publik (public key), sedangkan kunci dekripsi bersifat rahasia sehingga dinamakan kunci rahasia (*private key* atau *secret key*).

2.1 RSA

Salah satu algoritma kriptografi kunci-nirsimetri (kunci-publik) yang paling terkenal adalah RSA (Rivest, Shamir, Adleman). Algoritma ini dibuat oleh Ron Rivest, Adi Shamir, dan Leonard Adleman. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima. Pemfaktoran ini dilakukan untuk memperoleh kunci rahasia. Selama pemfaktoran bilangan besar menjadi faktor-faktor prima belum ditemukan algoritma maka selama itu pula keamanan algoritma RSA tetap terjamin [MUN06].

Untuk membangkitkan pasangan kunci (kunci publik dan kunci rahasia), beberapa langkah yang harus dilakukan antara lain:

1. Pilih dua bilangan prima sembarang p dan q
2. Hitung n = p.q. Sebaiknya p ≠ q maka n = p2 sehingga p dapat diperoleh dengan menarik akar kuadrat dari n
3. Hitung φ(n) = (p - 1).(q - 1)
4. Pilih kunci publik e yang relatif prima terhadap φ(n)
5. Bangkitkan kunci rahasia d dengan persamaan:
   e.d ≡ 1 (mod φ(n)) .....(1)
   Perhatikan bahwa persamaan diatas ekivalen dengan e.d ≡ 1 + k φ(n), sehingga d dapat dihitung dengan:
   d = (1 + k φ(n))/e .....(2)

Disini akan didapat dua hasil perhitungan yaitu pasangan n dan d sebagai kunci rahasia (private) dan pasangan n dan e sebagai kunci publik yang sifatnya tidak rahasia [MUN06].

Untuk mengenkripsi pesan (plainteks) menjadi cipherteks, langkah-langkah yang dilakukan adalah:

1. Ambil kunci penerima pesan e dan modulus n
2. Nyatakan plainteks m menjadi blok-blok m1, m2, ... sedemikian sehingga setiap blok merepresentasikan nilai dalam selang [0,n-1]
3. Setiap blok mi dienkripsi menjadi blok ci dengan rumus

   $$E_e(m_i) = c_i \equiv m_i^e \bmod n$$ ....(3)

   Sedangkan untuk dekripsi digunakan rumus:

   $$D_d(c_i) = m_i = c_i^d \bmod n$$ ..... (4)

Kekuatan algoritma RSA ini terletak pada sulitnya memfaktorkan suatu bilangan yang besar menjadi faktor primanya. Sehingga semakin panjang pasangan kunci yang digunakan (dalam artian semakin besar bilangan kuncinya) maka algoritma RSA akan semakin aman.

2.2 Protokol Diffie-Hellman

Protokol merupakan serangkaian aturan yang terurut yang harus dilakukan untuk tercapainya tujuan. Seperti yang telah dijelaskan pada bagian sebelumnya bahwa Protokol Diffie-Hellman diperkenalkan oleh Whitfield Diffie dan Martin E. Hellman. Protokol Diffie-Hellman ini digunakan untuk mempertukarkan kunci antara pihak yang saling berkomunikasi. Protokol ini bekerja pada saluran komunikasi publik yang tidak aman, namun dapat menghasilkan kunci (*shared secret key*) secara aman.

Secara matematis Protokol Diffie-Hellman digambarkan sebagai berikut [MUN06] :

Parameter Umum

Misalkan Alice dan Bob akan mempertukarkan kunci melalui saluran publik. Maka untuk dapat menggunakan protokol ini keduanya akan memilih suatu bilangan dasar n dan g sedemikian sehingga g < n. Nilai n dan g tidak perlu rahasia. Bahkan, Alice dapat membicarakannya melalui saluran yang tidak aman sekalipun.

Algoritma Diffie-Hellman

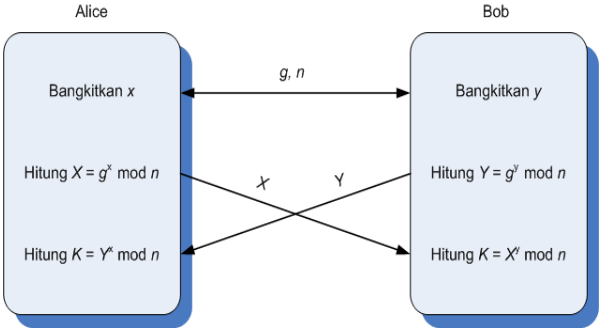
1. Alice membangkitkan bilangan bulat acak yang besar x dan mengirim hasil perhitungan berikut kepada Bob:
   X = gx mod n .....(5)
2. Bob membangkitan bilangan bulat acak besar y dan mengirimkan hasil perhitungan berikut kepada Alice:
   Y = gy mod n.....(6)
3. Alice menghitung
   K = Yx mod n..... (7)
4. Bob menghitung
   K' = Xy mod n..... (8)

Jika perhitungan dilakukan dengan benar, maka

K = K'

yang berarti kunci simetri sudah berhasil diterima oleh kedua belah pihak. Baik K dan K' sama dengan gxy mod n. Carol yang menyadap pembicaraan antara Alice dan Bob tidak dapat menghitung K. Ia hanya memiliki informasi n, g, X dan Y, tetapi ia tidak mempunyai informasi nilai x dan y. Untuk mengetahui nilai x dan y, ia perlu melakukan perhitungan logaritma diskrit, yang mana sangat sulit dikerjakan.

Gambar 1 memperlihatkan diagram algoritma pertukaran kunci Diffie-Hellman (Protokol Diffie-Hellman). Di akhir perhitungan Alice dan Bob telah memiliki kunci rahasia yang sama, K.



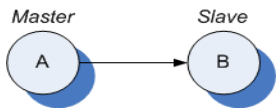
Gambar 1 Protokol Diffie-Hellman

3. *BLUETOOTH*

Bluetooth beroperasi menggunakan gelombang radio dengan frekuensi 2,4 Ghz ISM band pada 79 *channel* [LWC03]. *Bluetooth* memanfaatkan prinsip *frequency hopping spread spectrum* (FHSS) diantara 79 channels yang tersedia dengan kecepatan 1600 *hops/second*, yaitu setiap detik *Bluetooth* akan mengganti *channel* operasinya sebanyak 1600 kali. Hal ini ditujukan agar tidak terjadi bentrokan penggunaan saluran antara satu perangkat dengan perangkat lainnya. *Bluetooth* mampu berkomunikasi pada jarak antara 10 sampai 100 meter dan hanya mengkomsumsi 2,5 mW daya dengan kecepatan pengiriman data bisa mencapai 3 Mbps. Hal ini jelas memperlihatkan bahwa *Bluetooth* adalah perangkat komunikasi yang murah daya dan murah biaya.

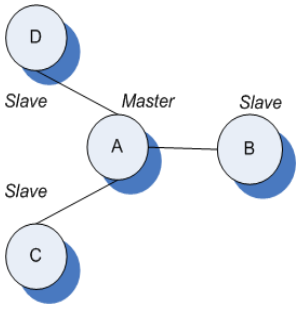
3.1 Jaringan Bluetooth

Jaringan *Bluetooth* menggunakan prinsip *master-slave* [KLI04]. Perangkat yang bertindak sebagai *master* adalah perangkat yang menginisiasi terbentuknya koneksi. Sebagai contoh misalnya terdapat dua buah perangkat Bluetooth (A dan B) yang hendak berkomunikasi. Perangkat A akan menginisiasi koneksi dengan menghubungi perangkat B, hal ini berarti perangkat A akan bertindak sebagai *master* dan perangkat B bertindak sebagai *slave* dan komunikasi seperti ini sering dikenal dengan istilah *point-to-point communication*, seperti pada Gambar 2.



Gambar 2 *Master-slave* pada jaringan *Bluetooth*

Selain komunikasi *point-to-point*, *Bluetooth* juga memungkinkan dibentuknya jaringan yang terdiri lebih dari dua perangkat. Jaringan seperti ini dikenal dengan nama *piconet*. *Piconet* adalah bentuk lazim dari jaringan *Bluetooth* yang terbentuk dari sebuah *master* dan satu atau lebih *slave*. Sebuah *piconet* dapat terdiri dari sebuah *master* dan maksimal tujuh buah *slave* yang aktif pada suatu waktu. Gambar 3 berikut menunjukan sebuah piconet.



Gambar 3 *Piconet*

3.2 Aspek Keamanan *Bluetooth*

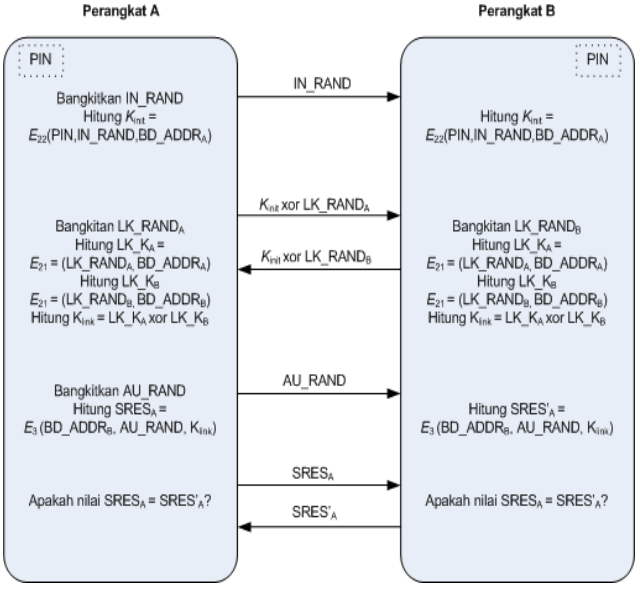
Pada *Bluetooth* terdapat tiga level kemanan yang memberikan pilihan kepada *developer Bluetooth* dimana akan mengimplementasikan sistem keamanan ini. Adapun level tersebut adalah [GEH04]:

1. *Level* 1 - *No security*, pada level ini perangkat *Bluetooth* tidak akan pernah menginisiasi prosedur keamanan.
2. *Level* 2 - *Service level enforced security*, keamanan diimplementasikan pada level layanan sehingga layanan yang menentukan apakah keamanan dibutuhkan atau tidak.
3. *Level* 3 – *Link level enforced security*, prosedur keamanan diinisiasi selama pembentukan koneksi perangkat Bluetooth. Pada level ini inisiasi dilakukan oleh layer bawah Bluetooth. Pengembang aplikasi tidak memiliki wewenang untuk menentukan metode keamanan yang akan diterapkan.

Pada level 2 terdapat tiga mode kemanan lagi, yaitu:

1. *Open services*, tanpa kebutuhan keamanan, setiap perangkat dapat mengaksesnya.
2. *Authentication-only services*, akses hanya untuk perangkat terotentikasi.
3. *Authentication and Authorization services*, akses hanya untuk perangkat yang sudah terotentikasi dan terotorisasi.

Pada mode keamanan paling tinggi maka komunikasi Bluetooth harus didahului dengan proses otentikasi, yaitu proses untuk memastikan bahwa pihak yang diajak berkomunikasi adalah pihak yang otentik, dan otorisasi, yaitu proses untuk memberikan hak akses layanan tertentu kepada suatu pihak. Adapun protokol yang diterapkan untuk proses otentikasi dapat digambarkan seperti Gambar 4.



Gambar 4 Protokol otentikasi Bluetooth

Dari gambar terlihat bahwa tujuan akhir dari protokol ini adalah untuk memastikan bahwa ke-dua pihak mendapatkan data kunci yang sama melalui proses yang dikenal dengan *mutual-authentication*. Sebelum hal itu dapat dilakukan maka perangkat harus melakukan proses pairing yang terdiri dari pembentukan kunci inisialisasi, dan kunci penghubung. Semua proses diatas mengharuskan perangkat mempertukarkan data-data satu sama lain. Data-data yang dipertukarkan pada proses otentikasi inilah yang dapat dimanfaatkan oleh pihak penyerang untuk membongkar PIN yang digunakan seingga proses komunikasi menjadi tidak aman [SHA05].

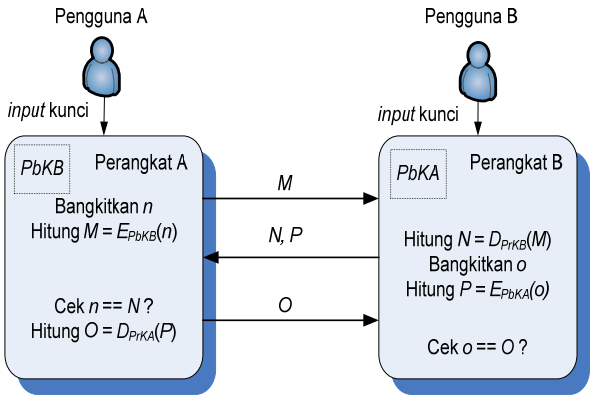
4. ANALISIS

Seperti yang diungkap pada bagian pendahuluan disini akan dibangun sebuah protokol otentikasi baru dengan memanfaatkan algoritma kriptografi kunci-publik. Adapun protokol yang akan dibangun meliputi protokol otentikasi pengguna dan perangkat.

4.1 Protokol Otentikasi Pengguna

Untuk memastikan bahwa pihak yang diajak berkomunikasi adalah pihak yang benar maka pengguna harus diotentikasi. Hal ini dilakukan karena munculnya masalah *device impersonation*, yaitu suatu tindakan untuk meniru atau memalsukan perangkat *Bluetooth* guna dapat berkomunikasi pada level keamanan tertentu dengan perangkat lain.

Otentikasi pengguna dapat dilakukan dengan melibatkan pengguna dalam proses *handshaking* suatu perangkat dengan perangkat lainnya. Keterlibatan ini dapat dijadikan sarana untuk memastikan bahwa pengguna tersebut adalah pengguna yang benar. Disini sebelum dapat berkomunikasi maka pengguna diharuskan untuk memasukan kunci (layaknya *password*) ke perangkat yang dipergunakannya. Adapun kunci yang harus dimasukan oleh pengguna adalah kunci privatnya. Selanjutnya akan diterapkan prinsip *Challenge and Response* guna memastikan data kunci yang digunakan benar. Adapun protokol otentikasi pengguna ini dapat dilihat pada Gambar

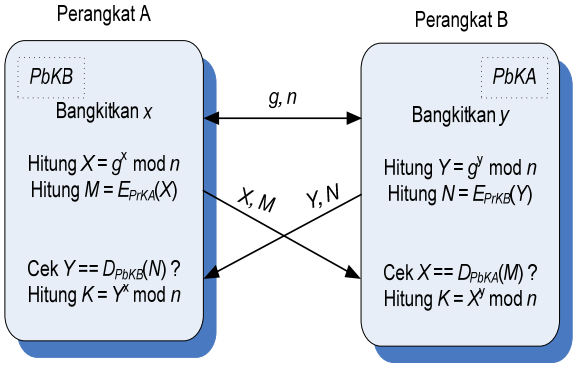


Gambar 5 Protokol otentikasi pengguna

4.1 Protokol Otentikasi Pengguna

Protokol otentikasi perangkat akan didasarkan pada protokol pertukaran kunci Diffie-Hellman. Karena inti dari kedua proses ini sebenarnya adalah bagaimana meyakinkan bahwa perangkat yang akan berkomunikasi dengan bluetooth telah memiliki kunci rahasia (*shared authentication key*) yang disepakati bersama antara pihak yang otentik. Protokol Diffie-Hellman akan dilengkapi dengan otentikasi dengan tujuan untuk mencegah terjadinya *man-in-the-middle-attack*.

Dalam protokol untuk otententikasi perangkat ini setiap data yang dikirimkan dienkripsi dengan kunci rahasia (*private key*) pengirim. Enkripsi data dengan kunci rahasia pengirim inilah yang memberikan aspek otentikasi data yang dikirimkan, karena data yang terenkripsi ini akan digunakan sebagai tanda tangan digital pengirim. Data hasil enkripsi ini dikirimkan bersamaan dengan data aslinya. Jadi setiap pengiriman bilangan, pengirim mengirimkan dua data bilangan sekaligus, bilangan aslinya dan bilangan yang terenkripsi dengan kunci rahasia. Dengan disertainya tanda tangan digital ini, pengirim seolah-olah mengatakan bahwa data yang dikirmkannya benar-benar berasal darinya. Skema dari protokol untuk otentikasi perangkat dapat dilihat pada Gambar 6.



Gambar 6 Protokol otentikasi perangkat

4.3 Pembangkitan Bilangan Acak

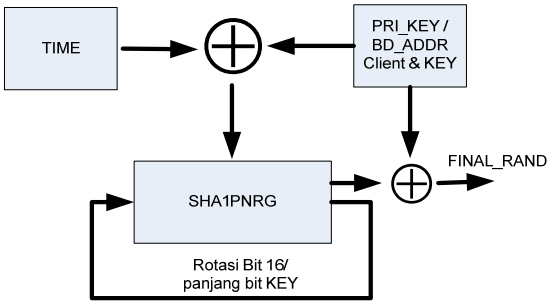
Protokol otentikasi pengguna dan perangkat membutuhkan bilangan acak yang akan digunakan untuk menginisialisasi protokol. Oleh karena itu bilangan acak mejadi suatu kebutuhan penting pada protokol ini. Atas alasan diatas maka dianggap perlu didesain suatu mekanisme pembangkitan bilangan acak untuk protokol otentikasi ini.

Untuk keperluan ini maka akan dipergunakan tenik *Secure Hash Algorithm Pseudo Random Generator* (SHA1PNRG) dengan umpan suatu bilangan yang didapat dari nilai unik *Bluetooth* atau kunci privat pengguna yang kemudian dikombinasikan dengan waktu pada saat itu. Terdapat dua mekanisme pembangkitan bilangan acak yang diperlukan, yaitu pada saat melakukan otentikasi perangkat dan

pengguna. Adapun kedua mekanisme tersebut adalah sebagai berikut:

1. Pembangkitan bilangan acak untuk otentikasi perangkat. Diperlukan masukan berupa alamat unik dari perangkat *Bluetooth client* (BD_ADDR), kunci pengguna dan waktu pada saat itu. Pertama nilai unik dari *Bluetooth* ini akan di-*xor*-kan dengan waktu. Kemudian dirotasi ke kiri sejauh panjang kunci bit dari kunci pengguna. Nilai inilah yang akan dijadikan umpan untuk SHA1PNRG. Terakhir nilai acak akan di-*xor*-kan kembali dengan kunci pengguna, sehingga dihasilkan nilai acak final (FINAL_RAND).
2. Pembangkitan bilangan acak untuk otentikasi pengguna. Pada prinsipnya pembangkitan bilangan acak disini hampir dengan pembangkitan bilangan acak pada otentikasi perangkat. Hanya saja masukan untuk proses ini adalah kunci privat pengguna (PRI_KEY) dan waktu pada saat itu. Proses selanjutnya sama dengan pada saat otentikasi perangkat, dan hasil akhirnya adalah bilangan acak final.

Proses dari pembangkitan diatas dapat dilihat pada gambar 7.



Gambar 7 Proses pembangkitan bilangan acak

5. IMPLEMENTASI

Lingkungan yang digunakan untuk implementasi Tugas Akhir terdiri dari dua jenis, yaitu lingkungan perangkat keras dan lingkungan perangkat lunak. Implementasi disini meliputi pembuatan program dan men-*debug*. Lingkungan perangkat keras merupakan tempat membuat program dan instalasi program tersebut. Adapun spesifikasi lingkungan perangkat keras adalah perangkat *mobile* yang mendukung J2ME (*Java 2 Microedition*) – *J2ME enabled device* dengan spesifikasi minimum sebagai berikut:

a. *Device Configuration* : CLDC-1.0 (*Connected Limited Device Configuration*)
b. *Device Profile* : MIDP-2.0 (*Mobile Information Device Profile*)
c. *Bluetooth enabled device*

Pada perangkat dengan spesifikasi diatas telah telah mengimplementasikan *Java API Bluetooth Wireless Technology* (JABWT). API ini akan digunakan untuk melakukan koneksi memanfaatkan *Bluetooth*. Aplikasi yang dibangun telah dites pada perangkat telepon genggam Nokia N73 dan Nokia 3110c.

Lingkungan implementasi perangkat lunak adalah perangkat yang digunakan untuk melakukan pengembangan Tugas Akhir ini. Lingkungan implementasi perangkat lunak adalah *NetBeans IDE* 6.0.1 pada sistem operasi *Windows Vista* dengan menggunakan *Java* sebagai bahasa pemrograman. *Platform Java* yang digunakan adalah J2ME. Dimana untuk platform J2ME digunakan *Wireless Toolkit* 2.5.2 (WTK 2.5.2) yang sudah terintegrasi dalam *NetBeans Mobile Module*. Gambar 8 dan 9 menunjukan contoh antarmuka perangkat lunak.

Gambar 8 Antarmuka Utama

Gambar 9 Antarmuka Daftar Perangkat

6. PENGUJIAN

Pengujian dilakukan terkait fungsionalitas perangkat lunak dengan tujuan untuk memastikan bahwa setiap fitur perangkat lunak berjalan sesuai dengan harapan. Untuk keperluan tersebut telah dirancang kasus uji untuk melakukan proses pengujian ini. Kasus uji didasarkan kepada setiap *use case* yang berhasil dianalisis. Kasus uji yang mengetes kemampuan utama perangkat lunak adalah kasus uji untuk koneksi dan otentikasi.

Dari hasil pengujian kasus uji diatas dapat disimpulkan bahwa proses koneksi oleh pengguna dengan perangkat yang ditemukan pada saat proses pencarian perangkat telah berjalan dengan baik. Segala bentuk kemungkinan kesalahan yang terjadi juga sudah ditangani oleh aplikasi. Pengujian ini sangat terkait dengan proses otentikasi, oleh karena itu jika proses otentikasi berhasil dilakukan maka otomatis perangkat akan terkoneksi satu sama lain.

Disisi lain untuk proses otenikasi memberikan kesimpulan bahwa protokol otentikasi pengguna dan perangkat yang dirancang telah dijalankan dengan

benar oleh aplikasi ketika akan melakukan koneksi dengan perangkat lain. Proses validasi protokol juga dilakukan sehingga untuk dapat terkoneksi masing-masing perangkat tidak pernah melanggar protokol ini. Proses otentikasi yang dilakukan ini bertujuan untuk mengamankan layanan yang dipublikasikan ke umum dari akses yang tidak diinginkan. Jalannya proses otentikasi sangat bergantung pada data kunci yang terdapat pada masing-masing perangkat. Kunci yang tidak sesuai akan mengakibatkan proses otentikasi gagal dilakukan dan berarti layanan tidak berhasil dikoneksi.

7. KESIMPULAN

Kriptografi kunci-publik merupakan salah satu alternatif lain untuk melakukan mekanisme otentikasi pada komunikasi Bluetooth, selain mekanisme yang sudah ada yaitu dengan memanfaatkan kriptografi kunci simetri. Kriptografi kunci-publik ini dapat diimplementasikan dengan baik untuk melakukan proses otentikasi perangkat dan pengguna pada komunikasi Bluetooth. Dengan memanfaatkan kriptografi kunci-publik, maka keamanan dari proses otentikasi akan lebih terjamin. Hal ini dikarenakan kenyataan bahwa belum ditemukannya algoritma yang mangkus untuk memfaktoran bilangan yang sangat besar menjadi faktor-faktor primanya. Pernyataan terakhir inilah yang menjamin kekuatan dari algoritma kritografi kunci-publik dan sekaligus akan menjamin keamanan proses otentikasi yang dilakukan dengan memanfaatkan algoritma ini.

Perangkat lunak BlueSeFT telah mampu mengakomodasi kebutuhan untuk melakukan proses otentikasi perangkat dan pengguna pada suatu komunikasi Bluetooth. Namun, karena keterbatasan implementasi maka proses otentikasi yang diaplikasikan hanya mampu sebatas layanan yang dipublikasikan oleh perangkat lunak BlueSeFT ini saja. Terlepas dari keterbatasan ini, konsep otentikasi yang diterapkan pada BlueSeFT memberikan ide bahwa pengamanan pada layanan merupakan salah satu metode yang sangat cocok diterapkan, disamping menggunakan mekanisme keamanan yang sudah ada, untuk menghindari berbagai kemungkinan serangan yang bisa dilakukan oleh pihak yang akan diajak berkomunikasi.

DAFTAR REFERENSI

[GEH04] Gehrmann,Christian dkk. (2004). *Bluetooth Security*. Artech House.

[KLI04] Klingsheim, Andre N. (2004). J2ME *Bluetooth Programming*.

[LWC03] Local Wireless Communication project team. (2003). *Bluetooth Threats and Security Measures*.

[MUN06] Munir, Rinaldi. (2006). *Dikat Kuliah IF5054 Kriptografi*.

[SHA05] *Shaked, Yaniv and Wool, Avishai. (2005). Cracking the Bluetooth PIN.*

[SUN05] Sun, Jun-Zhao dkk. (2005) . *Design, Implementation, and Evaluation of Bluetooth Security*.