KEAMANAN KOMUNIKASI MOBILE AGENT DENGAN MENGGUNAKAN *KEY ENCRYPTION KEY* (KEK) PADA SYMBIAN OS

Wiwin Suwarningsih
Bidang Sistem Informasi, Pusat Penelitian Informatika, Lembaga Ilmu Pengetahuan Indonesia
Jln. Sangkuriang 154 D Cisitu (Komplek LIPI) Bandung
e-mail: wiwin@informatika.lipi.go.id

ABSTRAKSI

Komunikasi yang dilakukan dengan sarana wireless (tanpa kabel) salah satunya adalah mobile. Semakin canggihnya teknologi mobile semakin banyak juga pertukaran data yang dapat dilakukan. Tidak hanya suara, gambar bahkan file pun bisa. Sehingga diperlukan suatu sequritas terhadap data yang akan dikirimkan tersebut. Makalah ini membahas perancangan aplikasi yang akan digunakan pada system keamanan komunikasi pada Symbian OS dengan melakukan enkripsi data, sehingga data menjadi aman.

***Kata kunci**: Enkripsi, mobile agent, security, komunikasi*

1. PENDAHULUAN

Sistem Operasi sebuah *smartphone* merupakan komponen software yang sangat kritis karena mengatur keseluruhan dari prangkat ponsel. Kebutuhan yang penting dari sebuah system operasi adalah multitasking, multithreading, real time operasi, manajemen daya yang efektif, mempunyai code yang kecil, mudah menambah fungsi-fungsi baru, dapat digunakan kembali, modular, connectivity dan handal.

Telah ada beberapa system operasi yang menjadi basis beberapa tipe ponsel dan perangkat bergerak lainnya seperti PDA, diantaranya yaitu Windows CE, Palm OS, Embedded Linux, Pocket PC dan Symbian OS. Namun untuk kalangan ponsel-ponsel kelas menengah ke atas dikuasai symbian OS yang merupakan system operasi hasil kolaborasi dari para pembuat ponsel terbesar di dunia, seperti Nokia, Motorola, Samsung, Sony Ericson dan Siemens.

Dalam tulisan ini akan membahas system operasi symbian OS yang memiliki fasilitas komunikasi dan keamanan. Dari segi keamanan data akan dibahas bagaimana caranya melakukan enkripsi data sehingga data tersebut relative aman. Sehingga data yang terkompresipun lebih cepat terkirim Karena ukurannya menjadi lebih kecil dibanding data sebelum di kompres.

Tujuan dari enkripsi data ini adalah untuk meningkatkan proteksi dan meningkatkan *confidentiality.*

2. KEAMANAN KOMUNIKASI MOBILE AGENT

2.1 Ukuran Keamanan pada Symbian OS

Ada tiga jenis ukuran keamanan dalam berkomunikasi yaitu *confidentiality, integrity* dan *availability*. *Confidentiality* berarti tidak ada data yang bisa didapat oleh orang yang tidak berkepentingan. Integriry berarti tidak ada data yang bisa diubah oleh orang yang tidak mempunyai hak akses. *Availability* berarti data dan service selalu tersedia untuk yang menginginkannya, tidak boleh terjadi serangan *denial of service*.

Mekanisme proteksi dapat dilakukan dengan beberapa cara, diantaranya untuk meningkatkan tingkat *confidentiality* dapat dilakukan dengan melakukan enkripsi, walaupun tidak ada algoritma enkripsi yang tidak bias dipecahkan, tetapi tingkat kesulitan memecahkannya perlu dipertimbangkan. Untuk meningkatkan integrity, dapat digunakan message certificate seperti digital signature. *Availability* dapat ditingkatkan dengan pengecekan keaslian dengan pengecekan password.

Dalam tool Symbian OS, ada generator untuk membuat private-public key yang merupakan *asymmetric cryptography* dan dapat mengeluarkan permintaan certificate. Pada gambar 1 merupakan tahapan pembuatan private key.

2.2 Jenis Ancaman Pada System Keamanan

Ancaman pada system keamana secara umum digolongkanmenjadi tiga kelompok, yaitu: penyingkapan informasi (*disclosure of information*), penolakan pelayanan, dan perubahan informasi. Ada cara dalam menguji ancaman ini dengan melihat aplikasi secara detail pada agent systemnya. Disini kita menggunakan komponen agent system ke dalam kategori ancaman dengan melihat identitas yang sumber dan target penyerangan Ini sangat penting sebagai catatan bahwa banyak ancaman yang akan dibahas untuk mengitung bagian-bagian dalam system client server dan selalu eksis dalam beberapa bentuk yang lalu.

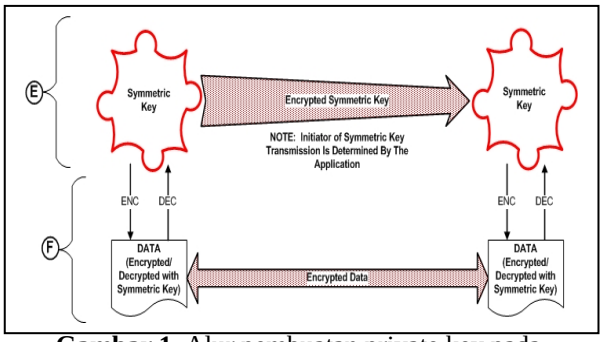
Mobile agent hanya menawarkan kesempatan lebih besar untuk salah pakai dan penyalahgunaan dalam skala besar dari ancaman yang signifikan. (lihat Gambar 1)

3. SKENARIO SISTEM KEAMANAN PADA MOBILE AGENT

Teknologi mobile agent dimulai dari membuat mobil agent dalam ruang lingkup penelitian laboratorium sampai dengan membuat

aplikasi secara komersial. Pada bagian ini akan dibahas mengenali relevansi keamanan dengan berbagai macam skenarionya.

Layer dari kebutuhan keamanan pada aplikasi dan sensitifitas kode mobile agent dan aliran data secara langsung merupakan ukuran mobilitas dari mobile agent. Sensitivitas yang diukur adalah dari penambahan kerja agent, peracangan penurunan mobilitas agent. Tugas mobile agent harus dibagi antara yang statis dan kemanaman mobile agent dalam kebutuhan transaksi.



Gambar 1. Alur pembuatan private key pada Symbian OS[1]

Pengelolaan kebijakan keamanan jaringan tidak seperti mengijinkan kode dari luar organisasi masuk ke dalam jaringan. Kebijakan ini lebih mengarah pada mengijinkan kode yang berasal dari lingkungan dalam jaringan atau kode yang berasal dari vendor luar yang sudah didefinisikan.

Hanya administrator jaringan yang dapat diijinkan untuk memperkenalkan agent ke dalam jaringan, pengendalian akses platform mobile agent, perancangan mobile agent dan pembangunannya harus konfirmasi untuk lebih mematangkan rekayasa perangkat lunak dan kualitas metoda pengendalian.

Teknik enkripsi dapat diaplikasikan pada agent dan pesan, bahkan jika pesan dari PDA (*Personal Data Assistance*) ke platform lainnya di enkripsi, maka informasi yang muncul tentang user akan ditampilkan berupa alamat tujuan.

4. PEMBUATAN KEK DENGAN METODA DEFFIE HELLMAN

Diffie-Hellman bukan metoda enkripsi dan tidak dapat digunakan untuk enkripsi data. Ini merupakan metoda pertukaran kunci sekuritas dari enkripsi data. Diffie-Hellman mengkompilasi pertukaran sekuritas dengan membuat "*shared secret*" atau disebut dengan "*Key Encryption Key*" (KEK). antara dua devices. *Shared secret* kemudian dienkripsi dengan symmetric key untuk sekuritas pengiriman. Symmetric key terkadang disebut dengan *Traffic Encryption Key* (TEK) atau *Data Encryption Key* (DEK). Terkadang KEK digunakan untuk sekuritas pengiriman dalam TEK.

Tahapan Proses KEK[1], adalah:

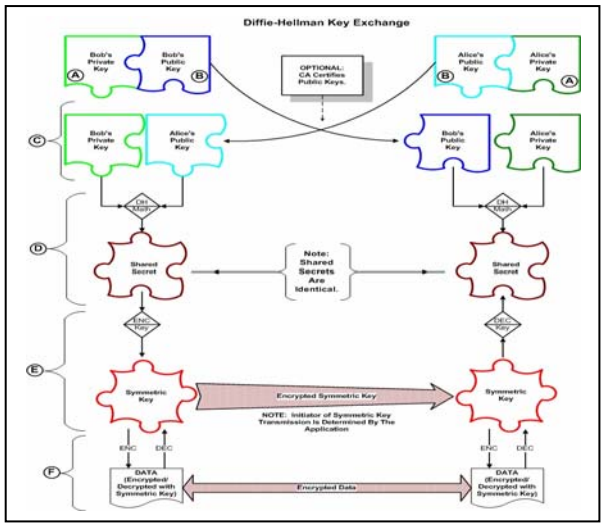
1. Setiap side mengenerate private key baik disisi penerima dan disisi pengirim.
2. Kemudian kedua system tersebut melakukan pertukaran public key.
3. Komunikasi yang dilakukan oleh setiap side menggunakan private key masing-masing dan public key dari system lainnya.
4. Protokol Deffie-Hellman mengenerate "shares secrets" untuk identifikasi kryptograpi key yang akan di share pada setiap side.
5. Kemudian share secret akan dikalkulasi dengan perhitungan secara matematis untuk membentu symmetric key.
6. Symetric key kemudian di enkripsi dan dikirm ke sisi penerima. Disisi penerima symmetric key diubah kembali menjadi shared secret.
7. Pada saat pembuatan symmetric key data di enkripsi disisi pengirim. dan data dideskripsi di sisi penerima.

Untuk lebih jelasnya dapat dilihat pada gambar 2.

5. ENKRIPSI DAN DESKRIPSI DATA

Dalam tulisan ini algoritma yang digunakan menggunakan adalah algoritma RSA yaitu blowfish algoritma dengan syntax bahasa c. Modul yang terbentuk yaitu:

a. Modul Enkripsi Data
b. Modul Deskripsi Data
c. Modul Pembuatan private key
d. Modul Inisialisasi Blowfish



Gambar 2. Proses KEK

Berikut ini modul yang dibangun:

a. Modul untuk enkripsi data

```
void Blowfish_encipher(blf_ctx *bc,
unsigned long *xl, unsigned long *xr)
{  unsigned long  Xl;
   unsigned long  Xr;
   unsigned long  temp;
   short          i;
   Xl = *xl;
   Xr = *xr;
   for (i = 0; i < N; ++i)
   {  Xl = Xl ^ bc->P[i];
      Xr = F(bc, Xl) ^ Xr;
      temp = Xl;
      Xl = Xr;
      Xr = temp;   }
   temp = Xl;
   Xl = Xr;    Xr = temp;
   Xr = Xr ^ bc->P[N];
   Xl = Xl ^ bc->P[N + 1];
   *xl = Xl;    *xr = Xr;
}
```

b. Modul deskripsi data

```
void Blowfish_decipher(blf_ctx *bc,
unsigned long *xl, unsigned long *xr)
{  unsigned long  Xl;
   unsigned long  Xr;
   unsigned long  temp;
   short          i;
   Xl = *xl;
   Xr = *xr;
   for (i = N + 1; i > 1; --i)
   {  Xl = Xl ^ bc->P[i];
      Xr = F(bc, Xl) ^ Xr;
      /* Exchange Xl and Xr */
      temp = Xl;
      Xl = Xr;
      Xr = temp;    }

   /* Exchange Xl and Xr */
   temp = Xl;
   Xl = Xr;
   Xr = temp;
   Xr = Xr ^ bc->P[1];
   Xl = Xl ^ bc->P[0];
   *xl = Xl;
   *xr = Xr; }
```

c. Modul Pembuatan private key

```
void blf_key (blf_ctx *c, unsigned char
*k, int len)
{       InitializeBlowfish(c, k, len); }
```

d. Algoritma Inisialisasi Blowfish

```
short InitializeBlowfish(blf_ctx *bc,
unsigned char key[], int keybytes)
{ short  i; short j; short k;
  unsigned long  data;
  unsigned long  datal;
  unsigned long  datar;

/* initialise p & s-boxes without file
read */
for (i = 0; i < N+2; i++)
{ bc->P[i] = bfp[i]; }
 for (i = 0; i < 256; i++)
  {     bc->S[0][i] = ks0[i];
        bc->S[1][i] = ks1[i];
        bc->S[2][i] = ks2[i];
        bc->S[3][i] = ks3[i];  }
j = 0;
for (i = 0; i < N + 2; ++i)
 { data = 0x00000000;
   for (k = 0; k < 4; ++k)
   {data = (data << 8) | key[j];
    j = j + 1;
    if (j >= keybytes)
       { j = 0; }    }
     bc->P[i] = bc->P[i] ^ data;  }
 datal = 0x00000000;
 datar = 0x00000000;
 for (i = 0; i < N + 2; i += 2)
 {  Blowfish_encipher(bc, &datal,
&datar);
  bc->P[i] = datal;
  bc->P[i + 1] = datar;  }
 for (i = 0; i < 4; ++i)
 { for (j = 0; j < 256; j += 2)
    {  Blowfish_encipher(bc, &datal,
&datar);
        bc->S[i][j] = datal;
        bc->S[i][j + 1] = datar; } }
  return 0;}
```

6. KESIMPULAN

Berdasarkan hasil analisa dan perancangan diatas maka dapat disimpulkan sebagai berikut:

- System sekuritas pada mobile agent sebenarnya tidak mutlak pada pembuatan enkripsi data saja, akan tetapi dengan adanya system enkripsi ini akan memperlambat para hacker untuk melakukan pengacakan data kita.
- Algritma yang dibuat dapat dikembangkan sesuai dengan kebutuhan untuk diterapkan pada system operasi lainnya seperti palm OS dan lain-lain.

PUSTAKA

[1] Keith Palmgren, *Diffie-Hellman Key Exchange – A Non-Mathematician's Explanation*. 2003 url: http://www.netip.com/articles/keith/diffie-helman.htm. Diakses Tanggal: 5 Maret 2007

[2] Adel Anaiba, *A Decision Support Model For Wireless Information Management Using Mobile Agent*, 2004. url : http://www.soc.staffs.ac.uk/aa11/wida.pdf diakses tanggal : 20 february 2007

[3] Wayne A. Jansen, *Countermeasures for Mobile Agent Security*, National Institute of Standards and Technology Gaithersburg, MD 20899, USA, 2002

[4] Wayne Jansen, Tom Karygiannis, *Privilege Management of Mobile Agents*, National Institute of Standards and Technology, 2001

[5] Hyungjick Lee, *The Use of Encrypted Functions for Mobile Agent Security*, Proceedings of the 37th

Hawaii International Conference on System Sciences - 2004

[6] Wayne Jansen, Serban Gavrila, *A Unified Framework for Mobile Device Security*, The National Institute of Standards and Technology, 2004.

[7] Wayne A. Jansen, *Determining Privileges of Mobile Agents*, National Institute of Standards and Technology, 2002.

[8] Srilekha Mudumbai, Abdeliah Essiari, William Johnston, *Anchor Toolkit: A Secure Mobile Agent System*, Proceedings of Mobile Agents '99 Conference, October 1999.

[9] William Farmer, Joshua Guttman, Vipin Swarup, *Security for Mobile Agents: Authentication and State Appraisal*, Proceedings of the Fourth European Symposium on Research in Computer Security (ESORICS '96), September 1996, pp.

[10] ITU-T Recommendation X.509 | ISO/IEC 9594-8: *Information Technology - Open Systems nterconnection - The Directory: Public Key and Attribute Certificate Frameworks*, March 2000.

[11] Wayne Jansen, Tom Karygiannis, *Mobile Agent Security*, National Institutes of Standards and Technology, NIST SP 800-19, August 1999.

[12] N. Suri et ali., *NOMADS: Toward a Strong and Safe Mobile Agent System*, Proceedings of the 4th International Conference on Autonomous Agents (Agents 2000) Barcelona, Catalonia, Spain, June 3-7, 2000.