IMPLEMENTASI ALGORITMA SCHMIDT-SAMOA PADA ENKRIPSI DEKRIPSI EMAIL BERBASIS ANDROID

Willy Ristanto\textsuperscript{1} \quad Willy Sudiarjo Raharjo\textsuperscript{2} \quad Antonius Rachmat C.\textsuperscript{3}
22084396@ukdw.ac.id \quad willysr@ti.ukdw.ac.id \quad anton@ti.ukdw.ac.id

Abstract

Cryptography is a technique for sending secret messages. This research builds an Android-based email client application which implement cryptography with Schmidt-Samoa algorithm, which is classified as a public key cryptography. The algorithm performs encryption and decryption based on exponential and modulus operation on text messages. The application use 512 and 1024 bit keys. Performance measurements is done using text messages with character number variation of 5 – 10.000 characters to obtain the time used for encryption and decryption process.

As a result of this research, 99.074\% data show that decryption process is faster than encryption process. In 512 bit keys, the system can perform encryption process in 520 - 18.256 miliseconds, and decryption process in 487 - 5.688 miliseconds. In 1024 bit keys, system can perform encryption process in 5626 – 52,142 miliseconds (7.388 times slower than 512 bit keys) and decryption process with time 5463 – 15,808 miliseconds or 8.290 times slower than 512 bit keys.

Keywords: Schmidt-Samoa, Encryption, Decryption, Android.

1. Pendahuluan
Penggunaan email pada smartphone sudah menjadi hal yang sangat umum, namun tanpa disadari bahwa pada dasarnya pesan email dikirimkan tanpa adanya pengamanan sama sekali. Data yang dikirimkan berupa plaintext sehingga mudah sekali untuk disadap ataupun dibaca oleh orang yang tidak berhak untuk membacanya. Penelitian ini berusaha untuk membuat sebuah aplikasi email client pada perangkat smartphone berbasis Android yang mampu menyediakan layanan keamanan berbasisan algoritma Schmidt-Samoa.

2. Teori Pendukung
2.1. Kriptografi

\textsuperscript{1} Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana
\textsuperscript{2} Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana
\textsuperscript{3} Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana
2.1. Asymmetric Encryption

*Asymmetric encryption* memiliki beberapa kelebihan dibandingkan *symmetric encryption* yaitu:

1. Tidak ada kebutuhan untuk mendistribusikan kunci privat sebagaimana pada sistem kriptografi simetri
2. Kunci publik dapat dikirim ke penerima melalui jalur yang sama dengan jalur yang digunakan untuk mengirim pesan. Saluran untuk mengirim pesan umumnya tidak aman
3. Ketiga, jumlah kunci dapat dikurangi. Pada *symmetric encryption*, semakin banyak penerima yang akan menerima pesan terenkripsi, maka akan semakin banyak pula kunci yang diperlukan untuk melakukan proses enkripsi dan dekripsi. Tetapi bila dengan menggunakan *asymmetric encryption*, cukup dibutuhkan 1 kunci yang digunakan untuk enkripsi pesan dan 1 kunci privat yang dimiliki oleh penerima pesan untuk melakukan proses dekripsi.

2.2. Algoritma Schmidt-Samoan
Schmidt-Samoan (2006) menjelaskan algoritma Schmidt-Samoan sebagai berikut:

Pembangkitan Kunci
1. Pertama pilih terlebih dahulu 2 bilangan prima p dan q, kemudian hitung nilai N (public key)
   \[ N = p^2 q \]  
   \[  \text{[1]} \]
2. Tentukan nilai d sebagai private key
   \[ d = N^{-1} \mod \text{lcm}(p-1, q-1) \]  
   \[  \text{[2]} \]

N adalah public key dan d adalah private key.
\text{lcm} adalah least common multiple, kelipatan persekutuan terkecil

Untuk mengenkripsi pesan m menjadi c kita menghitung sebagai berikut

\[ c = m^d \mod N. \]  
   \[  \text{[3]} \]

Untuk mendekripsi pesan ciphertext c kita menghitung sebagai berikut:

\[ m = c^d \mod pq, \]  
   \[  \text{[4]} \]

3. Hasil dan Pembahasan
Pada hasil penelitian, terlihat bahwa setelah dienkripsi, jumlah karakter ciphertext menjadi lebih banyak dibandingkan jumlah karakter pesan asli. Hal ini dikarenakan 1 karakter pesan asli yang diubah ke 3 karakter kode ASCII masing masing karakter, yang kemudian dimasukkan ke dalam rumusan enkripsi Algoritma Schmidt-Samoan yang menghasilkan bilangan yang sangat besar.
Tabel 1.
Hasil Pengujian Enkripsi Dekripsi Dengan Kunci 512 bit

<table>
<thead>
<tr>
<th>Jumlah karakter Plaintext</th>
<th>Waktu Eksekusi (dalam miliseconds) encrypt</th>
<th>decrypt</th>
<th>Jumlah karakter ciphertext</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>520</td>
<td>542</td>
<td>629</td>
</tr>
<tr>
<td>17</td>
<td>613</td>
<td>518</td>
<td>629</td>
</tr>
<tr>
<td>23</td>
<td>517</td>
<td>487</td>
<td>629</td>
</tr>
<tr>
<td>33</td>
<td>526</td>
<td>503</td>
<td>629</td>
</tr>
<tr>
<td>41</td>
<td>528</td>
<td>489</td>
<td>629</td>
</tr>
<tr>
<td>56</td>
<td>643</td>
<td>493</td>
<td>629</td>
</tr>
<tr>
<td>64</td>
<td>639</td>
<td>504</td>
<td>629</td>
</tr>
<tr>
<td>72</td>
<td>603</td>
<td>490</td>
<td>629</td>
</tr>
<tr>
<td>98</td>
<td>540</td>
<td>511</td>
<td>629</td>
</tr>
<tr>
<td>99</td>
<td>524</td>
<td>506</td>
<td>629</td>
</tr>
<tr>
<td>121</td>
<td>640</td>
<td>506</td>
<td>1253</td>
</tr>
<tr>
<td>122</td>
<td>620</td>
<td>508</td>
<td>1253</td>
</tr>
<tr>
<td>138</td>
<td>591</td>
<td>566</td>
<td>1253</td>
</tr>
<tr>
<td>150</td>
<td>586</td>
<td>531</td>
<td>1253</td>
</tr>
<tr>
<td>157</td>
<td>661</td>
<td>524</td>
<td>1253</td>
</tr>
<tr>
<td>168</td>
<td>639</td>
<td>507</td>
<td>1253</td>
</tr>
<tr>
<td>177</td>
<td>668</td>
<td>551</td>
<td>1253</td>
</tr>
<tr>
<td>185</td>
<td>650</td>
<td>535</td>
<td>1253</td>
</tr>
<tr>
<td>191</td>
<td>569</td>
<td>533</td>
<td>1253</td>
</tr>
<tr>
<td>195</td>
<td>614</td>
<td>522</td>
<td>1253</td>
</tr>
<tr>
<td>204</td>
<td>661</td>
<td>551</td>
<td>1877</td>
</tr>
<tr>
<td>213</td>
<td>658</td>
<td>551</td>
<td>1877</td>
</tr>
<tr>
<td>224</td>
<td>668</td>
<td>564</td>
<td>1877</td>
</tr>
<tr>
<td>247</td>
<td>674</td>
<td>556</td>
<td>1877</td>
</tr>
<tr>
<td>255</td>
<td>674</td>
<td>566</td>
<td>1877</td>
</tr>
<tr>
<td>262</td>
<td>671</td>
<td>550</td>
<td>1877</td>
</tr>
<tr>
<td>278</td>
<td>662</td>
<td>553</td>
<td>1877</td>
</tr>
<tr>
<td>283</td>
<td>679</td>
<td>560</td>
<td>1877</td>
</tr>
<tr>
<td>287</td>
<td>667</td>
<td>574</td>
<td>1877</td>
</tr>
<tr>
<td>295</td>
<td>681</td>
<td>570</td>
<td>1877</td>
</tr>
<tr>
<td>320</td>
<td>720</td>
<td>555</td>
<td>2505</td>
</tr>
<tr>
<td>328</td>
<td>751</td>
<td>569</td>
<td>2505</td>
</tr>
<tr>
<td>339</td>
<td>729</td>
<td>553</td>
<td>2505</td>
</tr>
<tr>
<td>345</td>
<td>773</td>
<td>578</td>
<td>2505</td>
</tr>
<tr>
<td>358</td>
<td>788</td>
<td>551</td>
<td>2505</td>
</tr>
<tr>
<td>367</td>
<td>764</td>
<td>560</td>
<td>2505</td>
</tr>
</tbody>
</table>
Tabel 1. (Lanjutan)
Hasil Pengujian Enkripsi Dekripsi Dengan Kunci 512 bit

<table>
<thead>
<tr>
<th></th>
<th>370</th>
<th>378</th>
<th>386</th>
<th>393</th>
<th>567</th>
<th>672</th>
<th>862</th>
<th>1490</th>
<th>1812</th>
<th>2790</th>
<th>3950</th>
<th>4342</th>
<th>5581</th>
<th>6146</th>
<th>7451</th>
<th>8685</th>
<th>9060</th>
<th>10121</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>786</td>
<td>751</td>
<td>743</td>
<td>768</td>
<td>850</td>
<td>898</td>
<td>1102</td>
<td>1213</td>
<td>1599</td>
<td>2660</td>
<td>4388</td>
<td>5054</td>
<td>7427</td>
<td>8619</td>
<td>11890</td>
<td>15325</td>
<td>16756</td>
<td>18256</td>
</tr>
<tr>
<td></td>
<td>569</td>
<td>546</td>
<td>577</td>
<td>568</td>
<td>630</td>
<td>668</td>
<td>697</td>
<td>722</td>
<td>789</td>
<td>1002</td>
<td>1517</td>
<td>1731</td>
<td>2423</td>
<td>2856</td>
<td>3723</td>
<td>4495</td>
<td>5133</td>
<td>5688</td>
</tr>
<tr>
<td></td>
<td>2505</td>
<td>2505</td>
<td>2505</td>
<td>2505</td>
<td>3753</td>
<td>4381</td>
<td>5630</td>
<td>9382</td>
<td>11887</td>
<td>17516</td>
<td>25021</td>
<td>27522</td>
<td>35027</td>
<td>38780</td>
<td>46910</td>
<td>54415</td>
<td>56292</td>
<td>59986</td>
</tr>
</tbody>
</table>

Dari hasil penelitian dengan menggunakan kunci 512 bit, terlihat bahwa, peningkatan jumlah karakter plaintext yang dienkripsi belum tentu juga akan diikuti dengan peningkatan waktu proses enkripsi dan dekripsi. Dari hasil penelitian tersebut menunjukkan karakter plaintext dengan jumlah 10.000 karakter dapat dienkripsi di bawah 20 detik, tepatnya 18.256 detik, dan proses dekripsi hanya memakan waktu 5.688 detik. Dalam penelitian pada kunci 512 bit juga terlihat bahwa proses dekripsi memakan waktu yang lebih singkat dibandingkan proses enkripsi, yaitu sekitar 98,148% (53 data penelitian dari total keseluruhan 54 data). Hal ini dikarenakan proses enkripsi perlu dilakukan pengecekan setiap karakter yang kode ASCII nya di bawah 3 digit, akan ditambahkan angka 0 supaya menjadi 3 karakter kode ASCII. Dengan 1 karakter pesan asli yang diwakilkan 3 karakter kode ASCII, akan mempermudah proses dekripsi pesan yang memecah deretan angka hasil penghitungan rumusan dekripsi setiap 3 karakter.

Pada jumlah karakter ciphertext yang dihasilkan, terlihat bahwa jumlah karakter ciphertext yang merupakan hasil enkripsi dari plaintext yang memiliki karakter 8 – 99 sama yaitu 629 karakter. Plaintext yang memiliki rentang 101-200 karakter menghasilkan ciphertext 1253 karakter, plaintext dengan rentang 201-300 karakter menghasilkan ciphertext 1877 karakter, dan plaintext yang terletak di antara 301-400 karakter menghasilkan ciphertext 2505 karakter. Berdasarkan setiap rentang 100 karakter, proses enkripsi menghasilkan jumlah karakter ciphertext yang sama. Tabel 1 dapat disederhanakan lagi menjadi sebuah tabel baru berdasarkan jumlah karakter ciphertext yang dihasilkan sebagai berikut:
Tabel 2.
Hasil Pengujian Enkripsi Dekripsi Dengan Kunci 512 bit berdasarkan rentang karakter plaintext

<table>
<thead>
<tr>
<th>Karakter Plaintext</th>
<th>Proses Enkripsi (dalam miliseconds)</th>
<th>Proses Dekripsi (dalam miliseconds)</th>
<th>Karakter Ciphertext</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rata-rata</td>
<td>Minimum</td>
<td>Maksimum</td>
</tr>
<tr>
<td>1-100</td>
<td>565,3</td>
<td>517</td>
<td>643</td>
</tr>
<tr>
<td>101-200</td>
<td>633,8</td>
<td>569</td>
<td>768</td>
</tr>
<tr>
<td>201-300</td>
<td>669,5</td>
<td>658</td>
<td>681</td>
</tr>
<tr>
<td>301-400</td>
<td>757,3</td>
<td>720</td>
<td>788</td>
</tr>
</tbody>
</table>

Dari tabel 1 dikatakan bahwa peningkatan proses enkripsi dan dekripsi terjadi untuk setiap rentang 100 karakter. Semakin besar rentang karakter plaintext yang dienkripsi, maka semakin banyak jumlah karakter ciphertext yang dihasilkan, maka proses enkripsi dan dekripsi akan selalu mengalami peningkatan waktu. Pada rentang 101-200 karakter plaintext, proses enkripsi memakan waktu rata – rata 633,8 miliseconds meningkat sebesar 12,11% dari 565,3 miliseconds pada rentang karakter yang sebelumnya yaitu 1-100 karakter. Proses dekripsi memakan waktu rata-rata 541,3 miliseconds meningkat sebesar 7,33 % dari 504,3 miliseconds. Pada rentang 201-300 karakter plaintext, proses enkripsi memakan waktu rata – rata sekitar 669,5 miliseconds meningkat sebesar 5,63% dan rata-rata proses dekripsi sebesar 558,4 miliseconds atau meningkat 3,15%. Dan pada rentang karakter 301-400 proses enkripsi memakan waktu 757,3 miliseconds meningkat sebesar 13,11% sedangkan proses dekripsi memakan waktu rata-rata 562,6 miliseconds meningkat sebesar 0,75% dari rentang karakter plaintext 201-300.

Dari Tabel percobaan yang menggunakan kunci 1024 bit, dibandingkan dengan yang menggunakan kunci 512 bit proses enkripsi dan enkripsi memakan waktu lebih lama. Hal ini karena algoritma Schmidt-Samoa melakukan perhitungan matematika dengan angka yang lebih besar 2 kali lipat. Dibandingkan dengan kunci 512 bit, pada kunci 1024 bit, sistem melakukan proses enkripsi 7,388 kali lebih lama dan proses dekripsi 8,290 kali lebih lama, dan ciphertext yang dihasilkan 99,637% lebih panjang.

Tabel 3.
Hasil Pengujian Enkripsi Dekripsi Dengan Kunci 1024 bit

<table>
<thead>
<tr>
<th>Jumlah Karakter Plaintext</th>
<th>Waktu Eksekusi (dalam miliseconds)</th>
<th>Jumlah karakter ciphertext</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Encrypt</td>
<td>Decrypt</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>5632</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>5635</td>
</tr>
<tr>
<td></td>
<td>23</td>
<td>5628</td>
</tr>
<tr>
<td></td>
<td>33</td>
<td>5626</td>
</tr>
<tr>
<td></td>
<td>41</td>
<td>5631</td>
</tr>
<tr>
<td></td>
<td>56</td>
<td>5634</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>5643</td>
</tr>
<tr>
<td></td>
<td>72</td>
<td>5732</td>
</tr>
<tr>
<td></td>
<td>98</td>
<td>5635</td>
</tr>
<tr>
<td></td>
<td>99</td>
<td>5654</td>
</tr>
<tr>
<td></td>
<td>121</td>
<td>6153</td>
</tr>
<tr>
<td></td>
<td>122</td>
<td>6038</td>
</tr>
</tbody>
</table>
### Tabel 3. (Lanjutan)
**Hasil Pengujian Enkripsi Dekripsi Dengan Kunci 1024 bit**

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>138</td>
<td>5981</td>
<td>5554</td>
<td>2501</td>
</tr>
<tr>
<td>150</td>
<td>6061</td>
<td>5516</td>
<td>2501</td>
</tr>
<tr>
<td>157</td>
<td>6072</td>
<td>5600</td>
<td>2501</td>
</tr>
<tr>
<td>168</td>
<td>6079</td>
<td>5528</td>
<td>2501</td>
</tr>
<tr>
<td>177</td>
<td>6061</td>
<td>5519</td>
<td>2501</td>
</tr>
<tr>
<td>185</td>
<td>5993</td>
<td>5510</td>
<td>2501</td>
</tr>
<tr>
<td>191</td>
<td>5976</td>
<td>5506</td>
<td>2501</td>
</tr>
<tr>
<td>195</td>
<td>5952</td>
<td>5541</td>
<td>2501</td>
</tr>
<tr>
<td>204</td>
<td>6288</td>
<td>5788</td>
<td>3749</td>
</tr>
<tr>
<td>213</td>
<td>6345</td>
<td>5689</td>
<td>3749</td>
</tr>
<tr>
<td>224</td>
<td>6290</td>
<td>5692</td>
<td>3749</td>
</tr>
<tr>
<td>247</td>
<td>6355</td>
<td>5712</td>
<td>3749</td>
</tr>
<tr>
<td>255</td>
<td>6283</td>
<td>5755</td>
<td>3749</td>
</tr>
<tr>
<td>262</td>
<td>6369</td>
<td>5702</td>
<td>3749</td>
</tr>
<tr>
<td>278</td>
<td>6352</td>
<td>5739</td>
<td>3749</td>
</tr>
<tr>
<td>283</td>
<td>6296</td>
<td>5644</td>
<td>3749</td>
</tr>
<tr>
<td>287</td>
<td>6401</td>
<td>5643</td>
<td>3749</td>
</tr>
<tr>
<td>295</td>
<td>6307</td>
<td>5864</td>
<td>3749</td>
</tr>
<tr>
<td>320</td>
<td>6627</td>
<td>5719</td>
<td>5001</td>
</tr>
<tr>
<td>328</td>
<td>6699</td>
<td>5689</td>
<td>5001</td>
</tr>
<tr>
<td>339</td>
<td>6822</td>
<td>5733</td>
<td>5001</td>
</tr>
<tr>
<td>345</td>
<td>6755</td>
<td>5703</td>
<td>5001</td>
</tr>
<tr>
<td>358</td>
<td>6641</td>
<td>5761</td>
<td>5001</td>
</tr>
<tr>
<td>367</td>
<td>6532</td>
<td>5774</td>
<td>5001</td>
</tr>
<tr>
<td>370</td>
<td>6882</td>
<td>5699</td>
<td>5001</td>
</tr>
<tr>
<td>378</td>
<td>6670</td>
<td>5767</td>
<td>5001</td>
</tr>
<tr>
<td>386</td>
<td>6641</td>
<td>5780</td>
<td>5001</td>
</tr>
<tr>
<td>393</td>
<td>6831</td>
<td>5884</td>
<td>5001</td>
</tr>
<tr>
<td>567</td>
<td>7317</td>
<td>5933</td>
<td>7498</td>
</tr>
<tr>
<td>672</td>
<td>7678</td>
<td>6064</td>
<td>8750</td>
</tr>
<tr>
<td>862</td>
<td>8407</td>
<td>6227</td>
<td>11247</td>
</tr>
<tr>
<td>1490</td>
<td>9121</td>
<td>6430</td>
<td>18744</td>
</tr>
<tr>
<td>1812</td>
<td>10498</td>
<td>6852</td>
<td>23745</td>
</tr>
<tr>
<td>2790</td>
<td>14260</td>
<td>6946</td>
<td>34991</td>
</tr>
<tr>
<td>3950</td>
<td>19730</td>
<td>8454</td>
<td>49986</td>
</tr>
<tr>
<td>4342</td>
<td>21438</td>
<td>9452</td>
<td>54983</td>
</tr>
<tr>
<td>5581</td>
<td>27610</td>
<td>10611</td>
<td>69977</td>
</tr>
<tr>
<td>6146</td>
<td>30083</td>
<td>11282</td>
<td>77475</td>
</tr>
<tr>
<td>7451</td>
<td>37634</td>
<td>13321</td>
<td>93718</td>
</tr>
<tr>
<td>8685</td>
<td>45964</td>
<td>15634</td>
<td>108712</td>
</tr>
<tr>
<td>9060</td>
<td>48450</td>
<td>15702</td>
<td>112461</td>
</tr>
<tr>
<td>10121</td>
<td>52142</td>
<td>15808</td>
<td>130612</td>
</tr>
</tbody>
</table>
Dari Tabel percobaan yang menggunakan kunci 1024 bit, dibandingkan dengan yang menggunakan kunci 512 bit proses enkripsi dan enkripsi memakan waktu lebih lama. Hal ini karena algoritma Schmidt-Samoa melakukan perhitungan matematika dengan angka yang lebih besar 2 kali lipat. Dibandingkan dengan kunci 512 bit, pada kunci 1024 bit, sistem melakukan proses enkripsi 7,388 kali lebih lama dan proses dekripsi 8,290 kali lebih lama, dan ciphertext yang dihasilkan 99,637% lebih panjang.

Sama seperti pengujian pada kunci 512 bit, untuk setiap satu data sampel pada tabel 3, peningkatan jumlah karakter yang dienkripsi belum tentu juga akan dikutip peningkatan waktu proses enkripsi dan proses dekripsi. Untuk kunci yang berukuran 1024 bit, karakter dengan jumlah 10121 karakter memakan waktu 52,142 detik dan didekripsi dengan waktu 15,808 detik. Pada jumlah karakter ciphertext yang dihasilkan, terlihat bahwa jumlah karakter ciphertext yang merupakan hasil enkripsi dari plaintext yang memiliki karakter 8 – 99 sama yaitu 1253 karakter. Plaintext yang memiliki rentang 101-200 karakter menghasilkan ciphertext 2501 karakter, plaintext dengan rentang 201-300 karakter menghasilkan ciphertext 3749 karakter, dan plaintext yang terletak di antara 301-400 karakter menghasilkan ciphertext 5001 karakter. Tabel 4.3 dapat disederhanakan lagi menjadi sebuah tabel baru berdasarkan jumlah karakter ciphertext yang dihasilkan sebagai berikut:

<table>
<thead>
<tr>
<th>Karakter Plaintext</th>
<th>Proses Enkripsi (dalam miliseconds)</th>
<th>Proses Dekripsi (dalam miliseconds)</th>
<th>Karakter Ciphertext</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rata-rata</td>
<td>Minimum</td>
<td>Maksimum</td>
</tr>
<tr>
<td>1-100</td>
<td>5649,5</td>
<td>5626</td>
<td>5732</td>
</tr>
<tr>
<td>101-200</td>
<td>6036,6</td>
<td>5952</td>
<td>6153</td>
</tr>
<tr>
<td>201-300</td>
<td>6328,6</td>
<td>6283</td>
<td>6401</td>
</tr>
<tr>
<td>301-400</td>
<td>6710</td>
<td>6532</td>
<td>6882</td>
</tr>
</tbody>
</table>

Pada pengujian kunci 1024 bit dalam rentang 101-200 karakter plaintext, proses enkripsi memakan waktu rata – rata 6036,6 miliseconds meningkat sebesar 6,85% dari 5649,5 miliseconds pada rentang karakter yang sebelumnya yaitu 1-100 karakter. Proses dekripsi memakan waktu rata-rata 5531,2 miliseconds meningkat sebesar 1,46 % dari 5451,5 miliseconds. Pada rentang 201-300 karakter plaintext, proses enkripsi memakan waktu rata – rata sekitar 6328,6 miliseconds meningkat sebesar 4,83% dan rata-rata proses dekripsi sebesar 5722,8 miliseconds atau meningkat 3,46%. Dan pada rentang 301-400 proses enkripsi memakan waktu 6710 miliseconds meningkat sebesar 6,02% sedangkan proses dekripsi memakan waktu rata-rata 5742,9 miliseconds meningkat sebesar 0,35% dari rentang karakter plaintext 201-300.

4. Kesimpulan dan Saran
Kesimpulan pada penelitian ini adalah sebagai berikut:
1. Sistem telah dapat mengimplementasikan sistem keamanan dalam pesan email dengan menggunakan algoritma Schmidt-Samoa.
2. Berdasarkan hasil penelitian proses dekripsi pesan 97,67% lebih singkat dibandingkan proses enkripsi, disebabkan karena pada proses enkripsi sebelum dilakukan perhitungan matematis, dilakukan proses padding terlebih dahulu.
3. Dibandingkan dengan kunci 512 bit, sistem pada kunci 1024 bit melakukan proses enkripsi 8,43 kali lebih lama dan proses dekripsi 9,34 kali lebih lama, dan ciphertext yang dihasilkan 99,637% lebih panjang.
4. Peningkatan jumlah karakter plaintext yang dienkripsi, belum tentu akan diikuti peningkatan waktu proses enkripsi dan dekripsi. Tetapi untuk setiap rentang 100 karakter pesan dienkripsi, waktu rata-rata proses enkripsi dan dekripsi akan selalu meningkat. Pada kunci 512 bit, proses enkripsi rata-rata meningkat 10,288% dan proses dekripsi rata-rata meningkat 3,74% dalam setiap rentang 100 karakter. Sedangkan pada kunci 1024 bit, proses enkripsi rata-rata meningkat 5,90% dan proses dekripsi meningkat 1,75% dalam setiap rentang 100 karakter.

Adapun saran untuk pengembangan penelitian ini adalah sebagai berikut:
1. Pesan yang dikirimkan dapat diinputkan dalam text editor WYSIWYG.
2. Karakter pesan yang dienkripsi sebaiknya lebih luas cakupannya tidak hanya kode ASCII 0-255 saja.

Daftar Pustaka

