ISSN: 2302 -2805

SISTEM KEAMANAN DATA PADA *WEB SERVICE* MENGGUNAKAN XML *ENCRYPTION*

Ari Muzakir
Program Studi Teknik Informatika, Universitas Bina Darma, Palembang
Jl. A. Yani No. 12 Palembang
email : ariemuzakir@gmail.com

Abstrak

Web service menggunakan teknologi XML dalam melakukan pertukaran data. Bentuk pengamanan yang diterapkan pada web services adalah dengan penggunaan teknik kriptografi kunci-publik. Implementasi yang telah dilakukan dengan menggunakan library keamanan akan memberikan kemudahan dalam membangun keamanan web service karena dengan dukungan library XMLSEC sebagai library pendukung untuk melakukan enkripsi dan dekripsi data mampu mengatasi masalah keamanan pada jalur transport dan database khususnya untuk konfidensialitas. XML encryption yang memanfaatkan algoritma kriptografi RSA dengan panjang kunci 1024 bit mampu memberikan perlindungan terhadap transmisi data antara client dan server web service sampai pada database. Hasil yang diperoleh yaitu pesan SOAP request terenkripsi dan mampu didekripsi dengan baik serta integritas dan keamanan data tetap terjaga.

Kata kunci :
Keamanan data, web service, XML encryption

1. Pendahuluan

Saat ini *web service*s menjadi sangat populer di *enterprise* karena kemampuannya dalam mengintegrasikan aplikasi-aplikasi yang berbeda *platform* dengan menggunakan dokumen XML. XML (*eXtensible Markup Language*) adalah sebuah standar untuk mendefinisikan data dalam format yang sederhana dan fleksibel. Dimana *web service* mendukung komunikasi antar aplikasi dan integrasi aplikasi dengan menggunakan XML dan *Web*. Faktor keamanan pada jalur komunikasi antara *client* ke *server web service* itu belum sepenuhnya terjamin. Hal ini dibuktikan dengan banyaknya faktor yang menimbulkan celah-celah ancaman terhadap *web service* tersebut seperti yang telah dilakukan oleh penelitian terdahulu.

Selain itu, pada kerahasiaan pesan yang dikirimkan melalui *web service* masih berupa data XML. Sehingga hal ini menyebabkan terjadinya data yang tidak asli ketika sampai di sisi penerima. Walaupun pesan telah di enkripsi menggunakan suatu algoritma maka bukan berarti bahwa pesan yang di terima oleh penerima benar-benar masih asli. Kemungkinan bahwa struktur pesan telah berubah ketika pesan dikirimkan atau ketika diterima.

Kemudian masalah keamanan *web service* pada kasus-kasus sebelumnya kebanyakan penelitian dilakukan pada satu model keamanan atau standar keamanan untuk *web service*. Sehingga dengan adanya sistem keamanan yang seperti ini dirasakan masih kurang memberi suatu perlindungan yang maksimal terhadap ancaman keamanan *web service* antara *client* ke *server service* sendiri walaupun secara umum sudah mampu mencukupi. Masih adanya kendala mengenai *web service* yaitu beberapa pihak yang masih merasa ragu untuk menerapkan *web service*, khususnya mereka yang menggunakan jaringan internet pada transaksinya. Keraguan ini dilihat dari tingkat keamanan dari teknologi *web service*. Aspek keamanan menjadi sangat penting untuk menjaga data atau informasi agar tidak disalahgunakan ataupun diakses secara sembarangan [1]. WS-*Security* juga mengatur cara menyisipkan *security token* dalam pesan SOAP dalam bentuk *plaintext* maupun dalam bentuk biner, seperti sertifikat X.509 [3].

Oleh karena itu, penelitian ini akan mencoba menghadirkan sebuah implementasi dari *prototype* keamanan *web service* berbasis pengamanan *service to services*. Dari implementasi ini dapat memproses dan mengamankan data yang diterima dari *client* sebelum disimpan ke *database server*. Cara ini dilakukan dengan mengenkripsi serta menyisipkan *security token* pada pesan SOAP *request* dan *response* dengan memanfaatkan XML *encryption.*

2. Tinjauan Pustaka

Beberapa penelitian yang telah dilakukan berkenaan dengan keamanan *web service* ini. Salah satunya mengenai spesifikasi dari keamanan *web service*s dan cara tersebut menanggulangi ancaman terhadap keamanan *web service*s. Baik dari segi *security web service* masih belum matang seperti CORBA dan RMI [3].

Selanjutnya, analisa mengenai bagaimana mengatasi tantangan pada keamanan *web service* dengan menyajikan keamanan kerangka atau *framework* terpadu yang didasarkan pada penggunaan otentikasi dan kerahasiaan. Sehingga dengan adanya mekanisme integritas pada *web service* dan untuk mengintegrasikan dan menerapkan mekanisme keamanan tersebut membuat *web service* kuat terhadap serangan [2]. Penelitian mengenai penyajian suatu metode yang komprehensif untuk suatu jaminan layanan keamanan dalam SOA. Dimana metode yang diusulkan mendefinisikan tiga tahap yaitu *security analysis*,

arsitektur jaminan keamanan, dan identifikasi Standar WS-*Security* [4].

Selain itu penelitian terhadap keamanan *web service* juga pernah dilakukan pada integrasi data laporan kejadian perkara satuan reserse kriminal (satreskrim) yang dilengkapi dengan mekanisme keamanan internal, dimana yang dilakukan pada implementasi mekanisme keamanannya adalah menambahkan fungsi-fungsi keamanan pada *tool* NuSOAP yang mana digunakan sebagai otentikasi serta untuk kerahasiaan pesan SOAP menggunakan kriptografi AES 128 [5]. Selanjutnya untuk implementasi terhadap otentikasi *user* untuk dokumen XML dengan menggunakan *username token* juga pernah dilakukan , melakukan pembuktian terhadap validasi dokumen XML dan melakukan pengujian terhadap dokumen XML [1]. Selanjutnya Untuk mengimplementasikan suatu XML *signature* untuk memperoleh dokumen XML yang *secure* pada kasus transkrip *online*. Dengan cara memperoleh transkrip yang memiliki tipe format XML yang terdapat *digital signature*-nya [6].

Kemudian untuk mengimplementasikan algoritma RSA untuk pembuatan pasangan kunci public dan kunci privat guna proses enkripsi dan dekripsi. Selain itu RSA juga berperan menunjukan jangkauan data yang dapat diproses. Selanjutnya mengimplementasikan *message digest* untuk fungsi hash SHA-1 yang digunakan untuk proses penandatanganan dokumen XML [7]. Penelitian lainya yaitu mengenai data XML yang dienkripsi menggunakan kunci publik dengan algoritma RSA dengan hasil implementasinya berupa dua buah program komputer yaitu findkey.exe dan crypto.exe yang dibuat menggunakan bahasa pemrograman C [8].

3. Metode Penelitian

3.1 Analisa Sistem

Secara umum sistem yang akan dibangun dalam penelitian ini adalah keamanan data pada *web service* dengan memanfaatkan XML *encryption* dan kriptografi RSA. Pada kriptografi RSA memanfaatkan library XMLSEC untuk pembuatan sepasang kunci publik.

Pada penelitian ini, terdapat dua analisa kebutuhan sistem yang akan diterapkan, yaitu:

1. Kebutuhan Fungsional

Dalam implementasinya, *web service* akan dibagi menjadi dua bagian, yaitu:

a. Client menghasilkan *web service* request

Tahap ini berkaintan dengan proses-proses yang dilakukan oleh *client* untuk melakukan request kepada *web service.*

b. Server mengotentikasi client dan mengembalikan response

Tahap ini menjelaskan beberapa proses yang dilakukan oleh *web service* setelah menerima SOAP *request* dari *client*. Proses yang terjadi antara lain memastkan integritas pesan, mengotentikasi pengguna, mengenkripsi data XML, dan mendekripsi data XML.

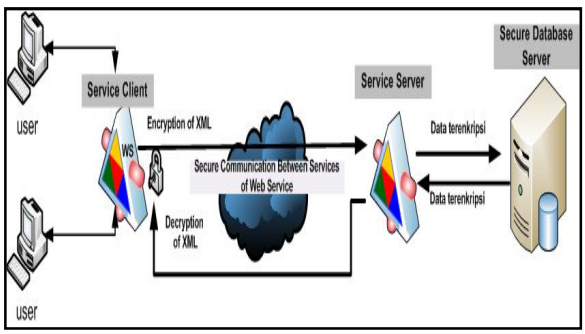
2. Kebutuhan Non Fungsional

Pada analisis kebutuhan non fungsional ini, proses sistem akan diperhatikan melalui dua aspek, yaitu:

a. Faktor waktu respon dari *site* internal maupun eksternal melalui *web service* yang tidak bisa diprediksi.
b. Proses enkripsi dan dekripsi pesan SOAP yang membutukan waktu yang tidak bisa diprediksi.

3.2 Perancangan Sistem

Sistem aplikasi yang akan dibangun memiliki arsitektur keamanan secara umum seperti pada Gambar 1, dimana setiap *request* dari *client* akan dilakukan otentikasi dan konfidensialitas. Otentikasi dilakukan ketika *client* berhasil melakukan *login* dan akan diberikan akses ke sumber daya sesuai dengan hak aksesnya, sedangkan konfidensialitas di gunakan pada proses enkripsi dan dekripsi.



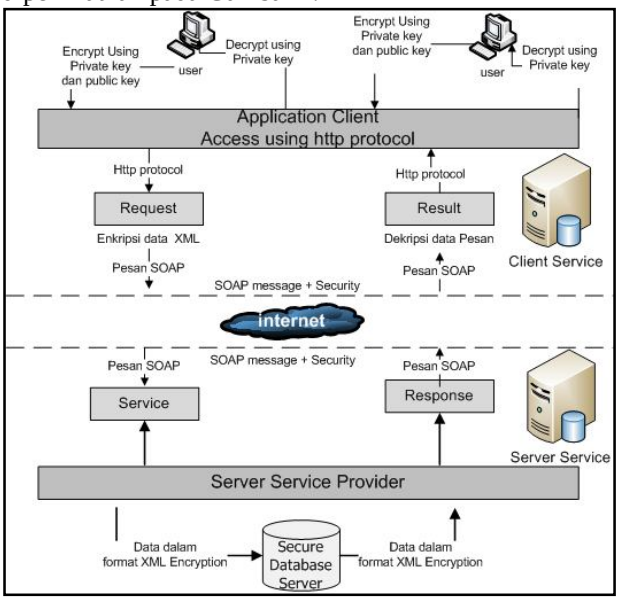
Gambar 1. Model Keamanan Antara *Client service* dan *Service Server* dari *Web Service*

Pada Gambar 1 memperlihatkan model dari keamanan *web service* antara *client service* dan *server service*. Gambaran umum dari keamanan sistem ini dimulai dari pengiriman data dari *user* menggunakan menggunakan protokol http ke *client service*. Pada tahap ini data akan dienkripsi menggunakan *private key* milik masing-masing *user* dan *public key*, kemudian data XML yang mengalir selanjutnya dalam keadaan aman (terenkripsi) pada komunikasi antara *client service* dengan *server service* dari *web service*. Selanjutnya adalah data hasil enkripsi tadi akan disimpan dalam *secure database* dengan format data XML. Proses dekripsi sendiri akan akan dilakukan ketika data diminta oleh *user* lainnya dengan menggunakan *private key* milik *user* masing-masing dan *public key*.

Perancangan mekanisme keamanan data ini bertujuan untuk memberikan gambaran mengenai

ISSN: 2302 -2805

kerahasiaan data dalam proses enkripsi dan proses dekripsi yang melibatkan algortima kunci public RSA. Enkripsi terjadi antara *client service* dan *server service* dimana bertujuan untuk mengamanakan jalur transmisi pada *web service* sendiri. Rancangan ini dapat diperlihatkan pada Gambar 2.



Gambar 2. Rancangan Mekanisme Kerahasiaan Data *User* Pada *Web Service*

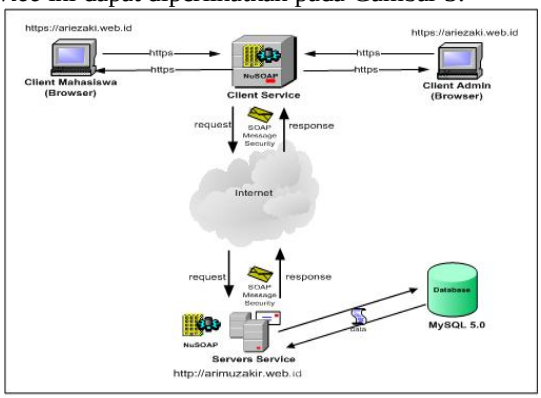
Implementasi terhadap rancangan arsitektur keamanan pesan SOAP akan disesuaikan dengan mekanisme *framework* NuSOAP. Selanjutnya menambahkan suatu *library* yang berisi beberapa fungsi yang dipergunakan dalam menunjang keamanan *web service* pada jalur transport. Selain itu untuk dapat mencapai tujuan dari keamanan tersebut akan dilakukan modifikasi terhadap rutin. Rutin tersebut berasal dari fungsi-fungsi didalam *class library* NuSOAP dan juga penambahan rutin program lainnya untuk keperluan keamanan *web service*.

Penambahan rutin program dan fungsi-fungsi keamanan dimaksudkan untuk pencapaian keamanan pesan. Dimana agar dapat melakukan hal-hal sebagai berikut:

1. Kemampuan untuk dapat mengamankan jalur transmisi data pada *web service* dengan menggunakan *security token* yang disertakan pada *header* SOAP *request*. Tujuannya adalah untuk otentikasi identitas *user* yang meminta layanan serta kendali akses untuk menentukan apakah *user* tersebut dilayani atau tidak.
2. Kemampuan untuk menjaga kerahasiaan serta keaslian data didalam pesan SOAP *request* dan SOAP *response*. Kemampuan ini ditunjang dengan penambahan beberapa *library* dari XMLSEC untuk keperluan enkripsi, dekripsi, serta *digital signature*. Pada penelitian ini memanfaatkan agoritma kriptografi RSA dengan panjang kunci 1024 bit.

3.3 Implementasi Sistem

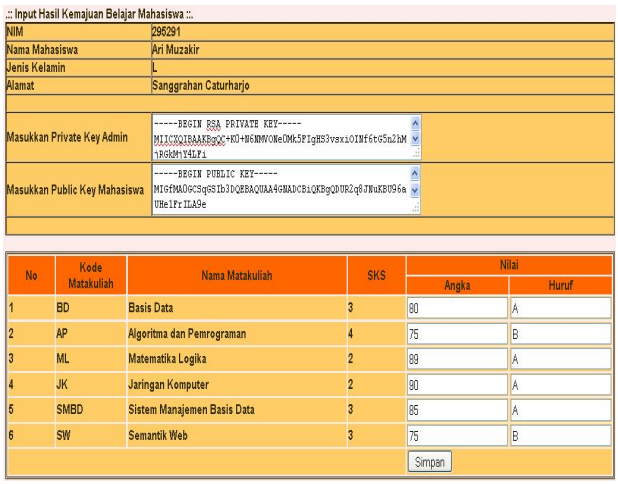
Setelah proses perancangan sistem dilakukan, tahap selanjutnya adalah membuat implementasi sistem keamanan pada *web service*. Sedangkan untuk implementasi dari keamanan *web service* ini, maka dirancang arsitektur dan skenario dalam alur yang akan diterapkan. Arsitektur dan skenario dari keamanan *web service* ini dapat diperlihatkan pada Gambar 3.



Gambar 3. Implementasi Arsitektur Skenario Keamanan *Web Service*

4. Hasil dan Pembahasan

Pengujian sistem merupakan elemen kritis dalam pengembangan sebuah perangkat lunak (*software*) karena akan merepresentasikan hasil akhir dari spesifikasi kebutuhan aplikasi, perancangan dan implementasi. Tujuan utama dari pengujian sistem adalah untuk memastikan bahwa hubungan antarmodul aplikasi telah memenuhi spesifikasi kebutuhan dan berjalan sesuai dengan skenario yang telah dideskripsikan sebelumnya. Pada Gambar 4 diperlihatkan implementasi dalam bentuk aplikasi data nilai dalam sistem keamanan data XML *web service*. Pada Gambar 4 memperlihatkan bahwa *user* perlu untuk memasukkan *private key* miliknya sendiri dan *public key* untuk melakukan enkripsi terhadap data yang akan dikirimkan.



Gambar 4. Ujicoba Keamanan *web service* pada data nilai

Hasil yang diperoleh pada Gambar 4 adalah keamanan data XML pada *web service* yang diujikan

pada data nilai mahasiswa. Pada kasus ini dititikberatkan pada pengujian konfidensialitas dan integritas data. Pada tahap pengujian konfidensialitas ini, *client service* akan mengenkripsi pesan SOAP yang akan dikirimkan yaitu pada data yang akan dikirim dengan memanggil fungsi yang enkripsi yang ada di *server* dan menggunakan *public key* dari *client*, proses enkripsi menggunakan algoritma RSA dengan panjang kunci 1024 bit. Proses dekripsi dilakukan pada *server service* dengan menggunakan kunci privat. Selanjutnya untuk melihat hasil pesan SOAP *request* ini yang berisi data terenkripsi dengan menggunakan metode XML *encryption* dapat diperlihatkan pada Gambar 5 berikut.

```
<SOAP-ENV:Body>
<EncryptedData xmlns="http://www.w3.org/2001/04/xmlenc#"
Type="http://www.w3.org/2001/04/xmlenc#Element">
        <EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#tripledes-
cbc"/>
        <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <EncryptedKey xmlns="http://www.w3.org/2001/04/xmlenc#">
                <EncryptionMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>        <KeyInfo
xmlns="http://www.w3.org/2000/09/xmldsig#">
                <KeyName/>        </KeyInfo>
                <CipherData>
        <CipherValue>ill9Dz2IXlxJuQ9Nrkdqup/4I/npZeywQIfivHO4MTEmXRYIBOVi
SpNROeKKcBNRLthH8ROQsdpTiN/7kO+ppfCBqjX+j9mGEHJOS+U0Tp9KqxFeN4YR3bJ
W4Lt/VOpxvVy+3TnGTv4cZxB0emNcR3H2BUYtjMen65aIflP5wa8=</CipherValue>
                </CipherData>
                </EncryptedKey>
        </KeyInfo>        <CipherData>
        <CipherValue>5BzJItoZ9gMD8pmjNbGg2ynQZR9jYVtmz3YQVHaE6jyMECV1
MXlFNKya13EoO7nCojzq3z03mSXXHC2CFrQkBW3R6pDFHTJqzc8i6he1VLOpcfz+J29A
HlswIndVR50RGW2emplnvGawsAD4zlVO15Uy9n010ZLDYL9sOB/MHe+Pk/hwTn0bu1Cel
7tk5kiBqH1865H386mmVenKJaISjgHulhqZtX</CipherValue>
        </CipherData>
</EncryptedData>
</SOAP-ENV:Body>
```

Gambar 5. Hasil Pesan SOAP *Request* Dengan Model Keamanan Menggunakan XML *encryption*

Hasil yang diperoleh dari Gambar 3 diatas adalah seluruh data tersebut akan dienkripsi oleh *client service*. Langkah ini bertujuan untuk menjamin kerahasiaan data pada jalur transmisi ke *server web service*. Kemudian dapat dilihat bahwa ketika data dikirimkan, maka *client* akan memanggil fungsi keamanan yang ada di *client service* yang bernama *library* class_wss.php. Ketika data dikirimkan dari *client service*, maka data SOAP akan akan dienkripsi.

Kemudian pada pengujian integritas data yang dilakukan yaitu dengan cara mengubah isi data XML *encryption* yang ada didalam database. Hasil menunjukkan bahwa data tidak dapat berubah karena *public key* dan *private key* yang dimasukkan tidak cocok. Jika dilakukan perubahan isi dokumen XML dari struktur <CipherValue> maka proses dekripsi akan *error*, seperti yang ditunjukkan pada Gambar 6 dimana isi dari <CipherValue> ditambah dengan angka "G".

Gambar 6. Perubahan Isi dari <CipherValue> dan *Error* Pada Saat Dekripsi Nilai

5. Kesimpulan dan Saran

5.1 Kesimpulan

1. Desain dan implementasi modul yang telah dilakukan dengan menggunakan *library* keamanan serta dukungan *library* XMLSEC sebagai *library* pendukung dan *library* class_wss. Hal ini mampu mengatasi masalah keamanan pada proses pengiriman yaitu keamanan otentikasi, dan konfidensialitas pesan SOAP *request* yang dihasilkan.
2. Hasil dari implementasi mengindikasikan bahwa konfidensialitas dapat terpecahkan dengan menerapkan konsep keamanan berbasis *library* keamanan yaitu XML *encryption*. Pesan SOAP *request* pada proses pengiriman dapat memenuhi standar keamanan *web service*, dimana data ketika dikirimkan dalam keadaan terenkripsi dengan menggunakan *library* class_wss yang telah dibangun.
3. Pengujian yang dilakukan pada *web service* dengan menerapkan model *library* class_wss. *Library* keamanan *web service* yang dibangun memberikan hasil yang baik, yaitu pesan SOAP *request* pada saat dikirimkan dalam bentuk terenkripsi dan mampu didekripsi.

5.2 Saran

Pertukaran kunci pada penelitian ini hanya sebatas *prototype* sehingga untuk kedepannya perlu pertukaran kunci yang lebih aman. Misalnya kunci *server* disimpan pada suatu repositori *database server* tersendiri agar lebih terjaga keamanannya.

Daftar Pustaka

[1] Rakhim, R, T, 2010, Keamanan *Web Service* Menggunakan Token, Tesis S2 Magister Ilmu Komputer, Universitas Gadjah Mada, Yogyakarta.

ISSN: 2302 -2805

[2] Zhang, W., 2009, Integrated *Security* Framework for Secure *Web Service*s, Research Institute of Applied Computer Technology, China Women's University.
[3] Adriansyah, A, Arifandi,W, dan, Wicaksono, N , 2005 ,*Keamanan Web Service*, Teknik Informatika, Institut Teknologi Bandung, Bandung.
[4] Fareghzadeh, N,(2009), *Web Service Security Method To SOA Development*, World Academy of Science, Engineering and Technology, No.49, 10 hal.
[5]Kenali, E., W., ,2010, Implementasi *Web Service* untuk Integrasi Data Satuan Reserse Kriminal (Studi Kasus Polda Lampung), Tesis S2 Magister Ilmu Komputer, Universitas Gadjah Mada, Yogyakarta.
[6] Suteja, B ,2004, Implementasi XML *Signature* untuk Secure XML Pada Kasus Integritas Transkrip Online, Tesis S2 Magister Ilmu Komputer, Universitas Gadjah Mada, Yogyakarta.
[7] Supriyanto,A., 2007, Otentikasi Dokumen XML menggunakan Algoritma RSA dan Hash SHA-1, Tesis S2 Magister Ilmu Komputer, Universitas Gadjah Mada, Yogyakarta
[8] Hartono, B., 2003, Pemakaian kriptografi kunci publik dengan algoritma RSA untuk keamanan data XML, S2 Ilmu Komputer, Universitas Gadjah Mada, Yogyakarta.

Biodata Penulis

Ari Muzakir, memperoleh gelar Sarjana Ilmu Komputer (S.Kom), Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Bina Darma, lulus tahun 2009. Tahun 2012 memperoleh gelar Magister Computer Science (M.Cs) dari Program Ilmu Komputer UGM. Saat ini sebagai Staf Pengajar Teknik Informatika Universitas Bina Darma Palembang.

ISSN: 2302 -2805