Security Visualization Collaborative Security Framework for Service Oriented Architecture

R. Kamatchi

***Abstract*—The Service oriented architecture is evolving as a new technology, which is currently adopted by many fields like Healthcare, Educational sector etc. It is highly adoptable with web services as an interoperable technology. There are various security mechanisms proposed for the individual security issues which works independently. This article is providing a collaborative security framework for the implementation of SOA with web services.**

***Index Terms*—Service oriented architecture, service provider, security broker.**

I. INTRODUCTION

The seamless interoperability characteristics of web services also introduce security concerns that do not exist in traditional distributed messaging techniques like RMI and CORBA. This is because of the reason that the SOAP based XML messages are not firewall proof and this could lead to intruders gaining access to sensitive systems using the interfaces provided by the WSDL files for service access.

Moreover, traditional websites designed using non-SOA frameworks allow users to interact with an application remotely through a web browser but a web service allows other applications to interact directly with it. This makes the web services more vulnerable to security problems.

To prevent the system from the major part of the security problems, the combined collaborative security mechanisms are required to be implemented. The first section of this paper is discussing about the various security components used in this holistic approach. The second section is describing the detailed flow of operations involved and the final section is providing the algorithm for the implementation.

A. Service Oriented Architecture

SOA is a new computation technology, not only satisfies user's application needs, but also included service broker and service provider interaction. All these are related to standard SOA. The SOA can add flexibility to system architectures, as it is possible to describe a service rather than the set of separate systems that provides the parts required to complete a task.

B. Basic Components and Functionalities of SOA

Basically The SOA contains two parts
1) Service
2) Connections

Service:

A service is a function or some processing logic or business processing that is well-defined, self-contained, and does not depend on the context or state of other services. Example of simple Service is Loan Processing Services, which can be self-contained unit for processing the Loan applications. Other example may be Weather Services, which can be used to get the weather information. Any application on the network can use the Weather Service to get the weather information.

C. Connections

Connections means the link connecting these self-contained distributed services with each other, it enables the client to Services communications.

SOA contains three main components
1) Service client
2) Service provider
3) Service registry

These components must use standard text formats familiar to both the service provider and the service client to enable provision and use of the service.

II. SECURITY PARAMETERS

The security parameters can be defined into the following two categories [1],[2]

A. Functional Aspects of Security

1) Authentication—verifying identity of users.
2) Authorization—deciding whether or not to permit action on a resource.
3) Data confidentiality—protecting secrecy of sensitive data.
4) Data integrity—detecting data tampering and making sure neither the sender nor the receiver can deny the message they sent or received.
5) Protection against attacks—making sure attackers do not gain control over applications.
6) Privacy—making sure that the application does not violate the privacy of the users.

B. Nonfunctional Aspects of Security

These aspects are nonfunctional in the sense that they do not directly relate to security. Instead, they are required to make sure that a security solution works well in an enterprise setting.

These are:

1) Interoperability—this concern is specific to SOA, where different security solutions must not break compatibility of services that are otherwise compatible.
2) Manageability—this concern is bigger for SOA, as a security solution needs to protect many different services.

c) Ease of development—this concern is common for any security solution. Be it SOA or traditional application development, complexity reduces adoption of any security solution.

III. SYSTEM COMPONENTS

The components involved in this model can be separated into two categories

2) Security broker
3) Security service components
4) Federated interface
5) Proxied interface

A. Security Broker

The security broker is similar to the middleware which is placed in between and evaluates the security constraints and bring out the commonality among the services used with the process choreography. [2]

(I) Modeler: The modeler receives a service description as an input to refer to the final web service. It does the following

1) Models the business process
2) It prepares the workflow
3) It identifies the set of services to perform the activities and reads the relationship among them.

(II) Locator: The services required to perform the activity will be spread across the network. That needs to be identified and accumulated to perform the process. The locator takes the description from the modeler and performs the location identification.

(III) Matchmaker: The security constraints will be analyzed and the security matchmaking composition will be prepared by the matchmaker. This will read the compatibility constraints and the security constraints specified with each service. Then it will map the services compatible with each other into a single flow. Each process will be in need of an initial service to accept the input from the consumer and one final service to produce the required output. The initial service will accept the input and perform the activity and forward its output as an input to the next activity in the sequence. While forwarding the output it will also forward the security constraints and compatible constraints specified with the same.

B. Security Service Components

(a) Authentication service: This service is responsible for the authentication of the service requested by the consumers. This authenticates the originality of the services requested.[3],[4]

(b) Authorization service: This service authorizes a particular service requested by the consumer with respect to its originality and the access rights that consumer has about the same service.

(c) Access control service: This service describes the access rights of the user on the requested service.

The service access requested by the consumer will be authorized only when the above mentioned three services return an OK message to the requestor. Then the complete access will be approved.

C. Federated Interface

The federated interface contains the details about the user credentials and the access rights information passed on through the same. This will be useful with the COI (Community of Interest) groups wherein the boundary of the user groups is definite and their service access must be similar with maximum retrievals. [5]-[7]

(d) Proxied interface:

This interface helps the user to hide his identity in the communication and also does the process of filtering as when required.

D. Detailed Process Allocation Algorithm

The following algorithm explains the detailed steps involved with the service modeling process. The process invoked by the user in turn, invokes various services corresponding to each and every requirements specified. As a whole a single process can be invoked and framed together to create a whole application.

S1: Services S_i, $i = 1....n$ will be defined and the process requirements R_i, $I = 1....n$ will be collected wherein the i defines the parameters defined by the user and the service specifications.

S2: Workflow preparation:

a) Requirements gathered from the user will be compared against the service specifications in the following way.

b) S_{ij}, $i, j=1...n$ will be compared with R_i to identify the specific service. Assign a new variable n with 0.

$$S_{ij} = R_i \text{ then } n=n+1;$$

c) The service with max (n) and higher performance will be identified.

S3: Security modeling

S_{ij}, $i, j=1...n$ which are identified by step2 will be modeled to P_i, $i=1...n$. P_i defines the processes involved in a single application. The individual services S_{ij} are the set of activities which will be performed sequentially to achieve the desired outcome.

S4: Matchmaking

SM----$\rightarrow$ SCA

To compare the compatibility constraints of the services with the user requirements. Assign a temporary variable m with 0.

a) $R_{ci} = S_{cij}$ wherein $i, j = 1...n$, (i = security parameters like privacy, confidentiality etc.) Then m=m+1.

b) If the maximum of n is identified and that single service will be identified and assigned to the individual activities of the process list.

c) $P_i=S_{ij}$ $i=1...n$.

 S5: Session ID exchange

C -----$\rightarrow$ AS thru PI, wherein

C = client
AS = Authentication Service
PI = Proxied Interface
Client (session ID)

S6: Key exchange:

a) C $\leftarrow$----$\rightarrow$ AS thru PI
b) PI ---$\rightarrow$ cache the data for filtering
c) FI ----$\rightarrow$ Stored the data for federation.

S7: C--$\rightarrow$ AS (Encrypted credentials)

AS ---$\rightarrow$FDB (Federated Database)

(Client ID, Credentials)

If data verified to be true then go to step8 or display error message.

S8: AS ---$\rightarrow$C

(Service token, OK)

S9: C---$\rightarrow$ SP

(ST, SN, ID)

S10: SP ----→ AS
(ST, SN)
S11: AS --→ AAS
(ST, ID)
If authorized go to S12 else display error message to SP and
go to step 7.
S12: AAS -→ ACS
(ST, SN)
If access rights are yes then go to step3 or display "Access
denied" message to SP.
S13: ACS --→ SP
Access permitted
SP -→ FI
To store the authorized user details
SP ---→ Client
To invoke the services.
S14: Go to step 9 to invoke the next service in the workflow.
Detailed description of the steps involved:

Step1: Process choreography:
a) Each and every process is a collection of services
belonging to various service providers. First the
identification of needed services will be done.
b) The needed services may be belonging to different
service providers with their own security aspects. This
should be verified before the access. The identification may
be carried out with the requirement specification mentioned
by the consumers.

Step2: Workflow preparation:
a) The service specification by the client will be handed
over to the browser.
b) The services satisfying the maximum requirements
specified by the consumers will be identified. The services
satisfying the above mentioned criteria may be more in
number. All services with the specification details will be
identified and accumulated.
c) The services based on the activity will be accumulated
and the tree structure of the same will be prepared. This tree
structure will be containing the service which satisfies the
maximum requirements will in higher priority and followed
by the nodes in decrement order of requirement
specifications.

Step3: Performance analysis
a) The set of services will be analyzed and the services with
the maximum performance attributes will be identified.
Each and every service definition assures a specific
percentage of the performance level. This is specified with
the policies of the same service.
b) The performance parameters can be verified from the
WSDL extensions of the corresponding services.

   Step 4: Security modeling
   a) The services identified by the locator from the
   service registry will be narrowed to the actual service
   after the performance analysis. A single service for a
   particular purpose will be identified. In the same way
   the complete activity list will be replaced with a
   specific service.
   b) The service requirements specified by the consumer
   will be containing the security capabilities and the
   security constraints. At the same time the service
   description also contains its own security constraints
   which will be verified against the requirements
   specified by the consumers.

c) The modeler will prepare the final workflow model
with the services which satisfies the initial and the final
constraints.

Step 5: Security matchmaking
a) The workflow model prepared by the modeler will
be handed over to the matchmaker in order to
verify the compatibility constraints. The services
belonging to a single workflow should work
sequentially to provide a complete solution to a
process defined.
b) The SCA (Secure Capability Authority) is in
charge of evaluating web service security
capabilities. Based on its evaluation, it will issue
signed SAML assertions certifying such
capabilities.
c) The compatibility element present with the WSDL
extensibility element will be verified and the final
workflow tree will be prepared and stored with the
database.

Step 6: Session ID generation:
a) Once the complete workflow is prepared will be
stored with the EIS database in order to execute the
process. All the services involved in the workflow
will be executed sequentially.
b) The process execution will select the first activity
in the workflow and the corresponding service
name will be taken from the server database.
c) Now the client will send an ID for invoking the
session with the server. This ID will be sent to the
Authentication Server (AS) through proxied
interface.

Step 7: Proxied interface:
a) This interface will help the clients to contact the
server with hiding the user details. This interface
will communicate with AAA server on behalf of
the clients and the replies can be forwarded to the
clients back.
b) This interface can have the filters in order to verify
the incoming and outgoing SOAP messages. The
SOAP messages with unauthorized access or
identity can be filtered.

Step 8: Key generation process
a) This process will happen between the consumer
and Authentication Server (AS). The key generated
by this process will be used for the encryption and
decryption of the credentials transferred between
them.
b) The set of algorithms are available with the
security as a service (SAS) component which will
avoid the mismatch among the algorithms used by
the consumer and the authentication server.
c) By using any of the available cryptography
algorithms, the key will be generated and
exchanged between the consumer and the
Authentication server (AS).

Step 9: Token generation process
a) Client requests for the token from the
Authentication service in order to start the session.
But for producing the token, the AS is in need of
the user credentials which describes the details
about the user.

b) The client details are verified from the federated database which contains the details about the registered services and the users.

c) After successful verification of the credentials, the token will be generated and can be used for the complete execution of the activities present with the workflow. But this token will be applicable only for that particular session. For the new session, once again the complete process of key generation will take place.

Step 10: Federated Identity:

a) As soon as the user submits their credentials for verification for the first time with the AS, those details will be saved with the federated database.

b) In the same way the registration of services will keep a cop o the service details from the WSDL file (service name, creator, security parameters) etc with the federated database.

c) The federated database will be used every time whenever the registered user and the service are accessed.

Step 11: Service invoke:

a) Once the token is received by the user, and then the services can be invoked directly based on the workflow details stored by the matchmaker.

b) The consumer will provide the token details, service name to the concerned service provider.

c) The service provider is required to verify the identity and the access rights that the consumer is having with the requested service.

Step 12: Authorization:

a) The service provider will send the details of the user and the service requested to the authorization service.

b) The authorization service will verify the user credentials stored in the federated database against the details provided by the service provider about the consumers.

c) If the details are verified and approved then the user will be authorized to user the specified services.

Step 13: Access control:

a) The details about the service will be forwarded to the access control service to verify the rights to the users on the requested service.

b) If the service is having complete access to the users, then the service access will be allowed to the corresponding user.

c) This access rights acknowledgement will be stored with the proxied interface. So, that the further access can be monitored and filtered based on access rights provided to the same.

S14: Termination

a) Once the complete session gets over then the token will be discarded and the new session will start with the session ID request.

b) Each session will execute a complete process which executes a collection of services sequentially.

c)

IV. PROCESS DIAGRAM:

The above mentioned diagram is clearly indicating the collaborative security model wherein the Authentication, Authorization and Availability parameters are fulfilled by the AAA service. The confidentiality parameters are fulfilled by the proxied interface with the usage of cryptographic techniques. The federated database provides the access control. This collaborative security framework provides the complete security solution to the Service Oriented Architecture.

The services established in the working environment are finely verified and authorized for the originality. The proposed framework contains security in various formats.

Security on demand

Security on compatibility

Security as necessary

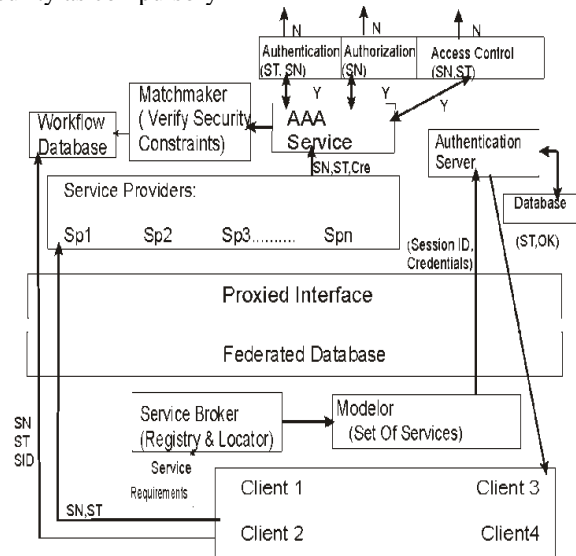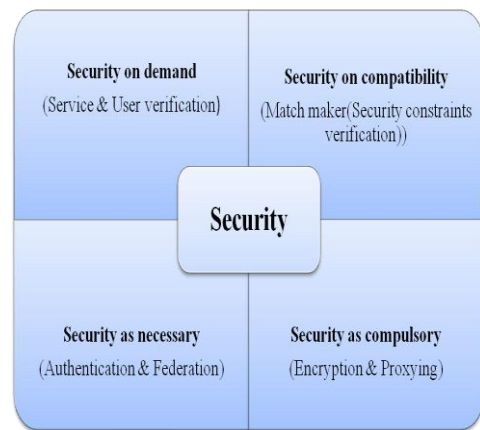Security as compulsory

Fig. 1. Security framework

Fig. 2. Security formats

As mentioned above the services published in the working environment is finely verified under various categories which are assuring the consumers the originality of the data available.

The workflow is prepared dynamically based on the user requirements and the performance guarantee attached with SLA. This workflow may vary for different executions because of the following reasons

- Performance degradation
- Security requirements
- Availability of services.

Based on the above conditions the matchmaker will

generate a new set of services related to workflow every time. The database attached with the matchmaker gets the new service choreography with different set of services during an execution which avoids the hacking of the database. Even though the list of services varies every time the security parameters remain the same with the services which enables the process of federation meaningful and useful.

V. CONCLUSION

The proposed model provides the holistic security approach for the service modeling process which involves various security measures like authorization, authenticity, access control, message transparency etc. This complete process should be handled with each and every service identification and process correlations by which the complete choreography will be secured.

Future work:

Even though the proposed framework contains various security parameters, the same can be analyzed with various security breaches and the functional deficiencies. In the same this architecture can be implemented with any of the specific field and analyzed for its benefits and shortcomings.

REFERENCES

[1] A. Maurizio and J. Sager, "Service Oriented Architecture: Challenges for Business and Academia," in *Proceedings of the 41ˢᵗ Hawaii International Conference on System Sciences*, pp. 1-7–, 2008.

[2] E. Pulier and H. Taylor, "Understanding Enterprise SOA," *Manning Publications CO*.,. pp. 23-25, 139-150, 2006.

[3] L. Cherbakov, R. Harishankar, and S. Kalyana, "Impact of service orientation at the business level," *IBM Journal*, vol. 44, issue 4, pp. 2.

[4] P. BouNassar and Y. Badr, "Towards Integrating Security Services in e-learning Platforms," *IEEE 978-1-4244-3834-1/09,* Pg.no. 573-576, 2009.

[5] "SOA Security," RamaRao Kanneganti, Prasad Chodavarapu, *Manning Publications Co,* ISBN: 1-932394-68-0, pp. 08-09.

[6] K. Gottschalk, S. Graham, and J. Snell, "Introduction to Web service architecture *IBM Journal*, vol. 41, no. 2, pp. 3, 4, 2002,

[7] I. Jørstad and S. Dustdar, "A Service Oriented Architecture Framework for Collaborative Services," in *Proceedings of the 14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise (WETICE'05),* pp. 1-8., 2009.