CHAPTER 5
BROKER BASED SECURE WEB SERVICE COMPOSITION

5.1 INTRODUCTION

In a service composition environment, service consumers, brokers and service providers work together to accomplish service composition. The service providers are responsible for binding a different kind of functionalities with web services, which are collected by brokers and provided to service consumers. The different brokers are responsible for different application domains, where each broker handles similar kind of requests for particular application domain. These kinds of broker based categorization assist us to build the system fast and reduce the load of central repository.

The broker based mechanisms can be associated with the security features to secure business processes and its data over the network from unauthorized users. There exist several broker based dynamic web service composition mechanisms, which have been experimented the various aspects of secure web service composition. It has already been discovered that a multi broker based system can serve as a secure dynamic web service composition system in which its brokers can interact with each other to accomplish their higher goals. But, there is need of similar kind of broker based web service composition mechanism for static composition, which can reduce the network processing load of central repository. The impact of brokers over static service composition through business processes can be analyzed with the help of comparative study, which is performed in the current research.

The necessary literature survey is illustrated in Section 5.2 and the broker based secure framework for business processes is discussed
in Section 5.3. The algorithmic representation of web service composition is also described in the same section. The implementation of the proposed framework with an example is presented in Section 5.4. The experimental evaluation of architecture with broker and without broker is explained in Section 5.5. The discussion on different approaches for web service composition is examined in Section 5.6. The Section 5.7 has summarized the chapter.

5.2 LITERATURE SURVEY

Several researchers have illustrated their research work based on broker based framework in dynamic web service composition [95, 96]. These all research work has demonstrated different broker based ideas. The proxy based dynamic web service composition is developed [95]. In any case, if the web service is unable to serve an incoming request then the request is automatically redirected to the proxy web service. The proxy web service searches and matches the desired web service for particular incoming request. These proxy services majorly discovers and binds the web services based on their etiquette, i.e., static, dynamic or genetic. The broker oriented dynamic web service composition mechanism is explained [96]. In this mechanism, different type of brokers such as, home broker and service broker are introduced. These brokers are responsible to find the best service composition approach available for web service composition based on the incoming request.

The home broker is responsible for two kinds of activities such as, service selection and service equalizing. The service selection activity is a process of selecting one optimum service or a group of services to fulfill incoming requests. The service equalizing is a process in which service-matching operation has to be performed based on service descriptions. The service broker has other activities such as, broker conversation, access control, service invoker and service adaptation. The broker conversation involves two parts of
conversation; firstly the conversation between service broker and composition broker, and secondly, the conversation between service brokers themselves. The access control activity is responsible for controlling broker’s access to particular web service. The service invoker activity is accountable for web service method invocation. In service adaptation, service broker regulates web service methods according to the service context.

The SAML language is used to map business processes and real time applications for performing service composition [97]. The SAML language is also used for the formation of secure dynamic broker based web service composition with the reference vocabulary defined for matching the security constraints [98]. The broker is responsible for receiving the incoming request and then develops the workflow design that models the business processes. The geospatial processing of web environment is carried out with the support of data integrity [99]. The study experimented several strategies to enable the passing of security token to geospatial based BPEL engines. The geospatial processing of web is performed by sensor services, where sensor service enables the live connection between sensor and end users.

The non-functional requirements such as, response time, service cost, availability and reliability are considered into account for web service composition with the help of BPEL language [100]. The service tracking and service selection mechanisms are used for suitable web service matching based on non-functional requirements provided by service consumer. A secure framework for web service composition is developed with the support of encryption and decryption mechanism at web service manager level [101]. The role of web service manager is to manage the web services, which is participating into composition operation for different institutions. The semantic based grid approach has been used to perform superior web service composition [102]. The excellent service will be selected based on their performance, when
large number of web services are handling different application domains.

The dynamic web service composition can be achieved by integrating different approaches such as, runtime reconfiguration, component adaption, composition language overview, workflow mechanism, ontology based composition and declarative composition [103]. The formation of sub web service and presenting an algorithm for automation of web service composition is demonstrated [104]. The sub web service is different from normal web service mechanism in a way that sub web service have only one output interface for communication, whereas web services have more than one output interfaces. The Artificial Intelligence (AI) based approach for automatic service composition is illustrated [105]. The translation of language, generation of composition model, evaluation and execution of composite service are different phases already been experimented using AI approaches.

The graph based Dijkstra algorithm of shortest path in workflow design is experimented based on service composition [106]. The graph vertices are represented as abstract web services and have different cost in multi graph scenario. The abstract web service is denoted as composite web service for different application domains such as, traveling, hotel management and air ticket booking processes etc. The broker based dynamic web service composition with different composition flow models is evaluated [107]. These flow models have focused on web service composition in sequential, parallel, conditional and looping manner. The web service composition is based on some key areas such as, service tracking, dynamic service composition, dynamic service selection and adaption.

The broker based architecture can be used for web service composition based on two different approaches namely the syntactic and semantic verification of web services [108]. It also focuses on the
development of matrices, which compare those verified semantics. The broker is developed as a web service and it focuses on set of quality issues such as, response time, service cost, availability and reliability. These QoS issues help end users to identify the non-functional requirements of dynamic web service composition. The intelligent broker approach for dynamic web service composition based on OWL language is verified [109]. The intelligent broker searches the web services based on their semantics, if the web service is not available then broker searches the services further based on their knowledge base. The knowledge base is considered as database or repository developed based on their past performances. In this research, the web services are registered with their semantic description into the ontologies. The semantic description is a XML based document with defined vocabulary. These ontologies are using few planning techniques such as, the service composition plan, web service searching, service invocation and result delivery to end users.

The existing literature survey explains web service composition and its security aspects such as, encryption and decryption through OWL and SAML languages. The BPEL language in contrast, has a support of authentication mechanism but doesn’t have support of encryption and decryption mechanisms. Additionally, the broker based web service composition mechanisms is demonstrated with the help of SAML and OWL languages but there is no support of broker based framework into BPEL language. Thus, there is need of broker based secure framework for web service composition through BPEL language. The proposed broker based secure framework is discussed in subsequent subsections.

5.3 BROKER BASED SECURE FRAMEWORK

The composite web service as a business process is required to fulfill the business requirements in current scenario. These business processes are need to be secured through cryptographic techniques for
end users. In addition, there is a need of broker’s layer, which can perform management of brokers and reduce the network processing load of central repository. Therefore, a broker based secure framework is proposed for web service composition as illustrated in the Figure 5.1. The security to broker based web service composition can be accomplished by transposition technique of cryptographic mechanism. The main components of this framework are security interfaces, business processes, business process manager, brokers, broker’s layer and UDDI.

Figure 5.1: Broker Based Secure Web Service Composition Framework

The application client or service consumer will request business processes for available web services. The security interface (SI) is developed to secure business processes. The BPEL process manager manages business processes, whereas BPEL process engine is responsible for business process execution. The BPEL process engine is residing inside BPEL process manager. The broker of proposed
broker based web service composition framework is discussed in the subsequent subsections.

5.3.1 Broker and Broker’s Layer

The business processes can communicate with different brokers on broker’s layer. The brokers are responsible for different activities such as, interaction with the web services, sending the incoming request, receiving response from web services etc. The broker performs searching and matching operations for fulfilling incoming requests of different domains. Brokers only who are available on broker’s layer know the location and content of web services. There can be several other brokers to manage web services. The UDDI is responsible for registering different web services or business processes belongs to different domain. The detailed functioning of proposed broker based secure web service composition framework is illustrated in Figure 5.2.

The service consumer will use sender side security interface for the encryption of incoming requests and then sends the encrypted requests to the business process. The business process deals with the BPEL process manager that uses request optimization process to optimize the incoming requests. It also uses selection mechanism to select the BPEL language for business process execution at execution engine. The incoming request then moves to receiver side security interface for decryption process. The original plaintext will be processed at the broker’s layer. The broker as central coordinator forwards the incoming request to all the connected web services with the help of star topology. When the web service receives the request, the web service will receive response for the incoming request and send back to the broker on broker’s layer. The broker sends the response to receiver side security interface to perform encryption on response. The encrypted response will be forwarded to business processes. The business processes send back the encrypted response to the sender side security interface. The sender side security interface performs decryption on encrypted response and then service consumer
receives the plaintext response for their request. The algorithmic description of the proposed broker based secure web service composition is described in subsequent sections.

Figure 5.2: Detailed Functioning of Broker based Secure Web Service Composition
5.3.2 Algorithmic Description

The broker based secure web service composition provides the quality outcomes in a secure manner. It prevents the unauthorized access, malicious attacks and harmful activities over the network. The algorithmic description of broker based web service composition is as follows:

Begin

Application client or service consumer uses sender side security interface to perform encryption on incoming request.

The service consumer sends encrypted request to business processes.

The business process will transmit the incoming encrypted request to receiver side security interface.

The receiver side security interface performs decryption on incoming request and get the original plaintext request.

The original plaintext request will move to broker’s layer.

The broker on broker’s layer will search and match the suitable web service for particular plaintext request.

If web service is found

Broker will send the incoming request to the web services. Web service collects different responses from all the web services. The collective response is send back to the broker on broker’s layer.

Else

“No Web Service found” message will be displayed.

End if
The broker performs encryption operation on response and sends the encrypted response back to the business processes.

The business process sends encrypted response back to the sender side security interface.

The sender side security interface performs decryption on encrypted response and gets the plaintext response.

The plaintext response is send back to the application client or service consumer by sender side security interface.

Broker based web service composition is completed.

End

The algorithm describes the complete step by step procedure to be performed. Algorithm illustrates that how the service consumer sends request and gets the response in a secure manner.

5.4 IMPLEMENTATION

In this chapter, different web services and business processes are considered for the implementation of proposed framework with the example of hotel management system. The proposed system assists end users to search the set of hotels in particular city before any plan for traveling. These web services are responsible for retrieving and responding the address, phone number and email id of all the hotels in particular city. The deployment of web services and business processes for different hotels, management of corresponding WSDL files are performed with the help of Jdeveloper and Intalio Business Process Designer. These web services are categorized and then composed with the help of brokers available at broker's layer. The business processes have different properties such as, partnerLinks, partnerTypes, partners, reply, response etc. They also support different operations such as, invoke, flow, sequence, assign and copy etc. A partnerLink operates between service consumer and business process, between business
process and brokers, between brokers and web services. The invoke operation is responsible for invoking the business process, invocation of corresponding brokers on broker’s layer, invocation of corresponding web services etc.

The implementation phase describes the development of web services, design and development of security interfaces, development and deployment of business processes. It also describes the development of brokers and broker’s layer. These business processes require XSD file and BPD diagram for the interaction between the web services and business processes. The service consumer sends the request for receiving the details of different hotels running in the city such as, name, addresses phone number etc.

The request will be encrypted by security interface at client end. The encrypted input will move to the business process and the business process is developed with the help of business process diagram in Intalio business process designer, as shown in Figure 5.3. Both of the web services are invoked with the help of Invoke1 and Invoke2 operations of business process. Each invoke operation handles different brokers on broker’s layer.

After development of business process, the authentication process is required on Intalio server. The authentication process requires predefined username and password on login page. After successfully login on the login page of Intalio server, a set of business processes will be available on BPEL server for testing purpose. In the current example, the name of hotel in encrypted format is considered as input to the business processes available on Intalio server. These business processes call different web services through different brokers for the same input. These web services are handling different hotels in the city and getting same input from brokers as shown in Figure 5.4.
Figure 5.3: Web Service Composition through Intalio

Figure 5.4: Encrypted Input for Business Process at Intermediate Level
The business process will interact with the brokers on broker layer. The forwarded request of business process will be decrypted at security interface at the service end. The original plaintext will move to brokers and brokers will send it to the web services. At service end, each web services will collect the response for incoming request and then send the response back to the brokers. The brokers will communicate with the various web services of different cities of the state.

The inputs to brokers are illustrated in Figure 5.5 and Figure 5.6 and outputs of brokers are illustrated in the Figure 5.7 and Figure 5.8, respectively. The development of brokers is performed with the help of some additional web services only.

Figure 5.5: Broker’s Input to First Web Service

95
Figure 5.6: Broker’s Input to Second Web Service

Figure 5.7: Broker’s Output of First Web Service
The brokers’ output will be encrypted at receiver side security interface and then it will sent back to the business processes. The encrypted outputs coming to business process from brokers are shown in Figure 5.9 and Figure 5.10, respectively. We have two different outputs corresponding to the different brokers of various cities. This business process output will move to the client in encrypted format only. At the client end, outputs will be decrypted into plaintext in a way that client can read the detail of hotels in the city.

After a successful development of broker based secure framework for web service composition, business processes will be secured over the network from unauthorized access. The business process also has the support of security interfaces, which are developed by transposition technique used in encryption and decryption mechanisms.
Figure 5.9: Encrypted Output 1

Figure 5.10: Encrypted Output 2
The complete evaluation will have set of different web services, which belong to different hotels running in the city. The consumer will request with a query regarding the hotels and the web service will respond with the details of the hotel to the consumer. These web services will participate in web service composition through business processes. The complete experiment is developed with the help of different tools such as, JDeveloper, NetBeans, Eclipse, Tomcat Server, Intalio Server and business process designer. The experimental evaluation will also focus on result computation of web service composition with broker less mechanism and/ or broker based mechanism.

5.5 BROKER BASED MECHANISM VERSUS BROKER LESS MECHANISM

The broker less mechanism includes the composition of web services directly through business processes, whereas the broker based mechanism includes the composition of web services through brokers. In broker based approach, the business process does not know anything about the web services and only brokers deal with the web services. The broker will compose the web services based on star topology approach. The broker itself is considered as web service, which is central coordinator for different web services in the star topology. The response time of different number of web services without broker and with broker is listed in Table 5.1.

The table represents the response time in seconds to the brokers. The tabular analysis states that the broker based mechanism is faster in processing the incoming requests as compared to the broker less mechanism. The graphical representation of response time and number of web services in respect of brokers and without brokers is shown in Figure 5.11. The broker less mechanism consumes more response time to business processes as compared to the broker based mechanism.
Table 5.1: Response Time for Broker Less and Broker Based Mechanism

<table>
<thead>
<tr>
<th>S. No</th>
<th>Number of Web Services</th>
<th>Response Time (in Sec)</th>
<th>Without Broker</th>
<th>With Broker</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td></td>
<td>2.3</td>
<td>0.56</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td></td>
<td>3.6</td>
<td>0.64</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td></td>
<td>4.1</td>
<td>0.81</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td></td>
<td>5.2</td>
<td>0.93</td>
</tr>
</tbody>
</table>

Figure 5.11: Graphical Representation of Broker and Broker less Mechanism
**5.6 DISCUSSION**

The broker based web service composition can be performed with the help of various approaches such as, star topology based approach, graph based approach and tree based approach. Various web service parameters such as, volumes of web services, QoS parameters, tools and technology and type of service composition are considered to evaluate the performance of web service composition in different approaches. The comparison of different approaches is available in Table 5.2 based on various parameters.

<table>
<thead>
<tr>
<th>Web Service Parameters</th>
<th>Star Topology</th>
<th>Graph</th>
<th>Tree</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volume of Web Service</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>QoS Parameter: Response Time</td>
<td>Fast</td>
<td>Slow</td>
<td>Medium</td>
</tr>
<tr>
<td>Tools and Technology Used</td>
<td>Intalio &amp; Java</td>
<td>Microsoft Visual Studio</td>
<td>Netbeans &amp; Java</td>
</tr>
<tr>
<td>Type of Composition</td>
<td>Static</td>
<td>Dynamic</td>
<td>Dynamic</td>
</tr>
</tbody>
</table>

**5.7 SUMMARY**

The broker based secure web service composition has been illustrated and tested in this chapter. This chapter focuses on introducing broker’s layer, which is responsible for managing the set of brokers and reducing the load of central repository. The brokers are classified in a way that each broker is responsible for responding the different kinds of application domain requests. The algorithmic description of complete process has been demonstrated in this chapter also. The algorithmic representation has illustrated the step-by-step
procedure of proposed broker based mechanism for secure web service composition.

The experimental evaluation has been evaluated and tested with the example of business processes. The comparative analysis of broker based and broker less environment and their impacts have been carried out. The impact analysis has illustrated that broker based mechanism is faster in response as compared to broker less mechanisms. The graphical representation and discussion over broker based versus broker less mechanism on different key parameters such as, volume of web services, response time and type of composition mechanisms have been explained additionally in this Chapter.