Security Model to Incorporate Add-On Security for Business Services

M.Thirumaran
Department of Computer Science and Engineering, Pondicherry Engineering College, India

P.Dhavachelvan
Department of Computer Science and Engineering, Pondicherry University, India

S.Abarna
Department of Computer Science and Engineering, Pondicherry Engineering College, India

ABSTRACT
Nowadays, the services offered by the Service Providers are subjected to many risks in terms of privacy agreements and hence they are treated as untrustworthy. Security risk analysis is fundamental to the security of any business and it is considered to be very essential in ensuring that controls and expenditure are fully commensurate with the risks to which the business is exposed. Hence, the Customer requires a set of security services and the model driven security specifications in terms of security policies such as authentication, authorization, confidentiality, integrity and audit in order to overcome such situations. The Customer security requirements should match with the security specifications that are recommended or delivered by the Service Provider and these contracts can be done through Service Level Agreements. In our paper we propose an Add-on security model which provides interoperable security services for the business services according to the security requirements of the business. We also establish the model as schema driven security model which facilitate dynamic integration of security services with the associated business services and finally to provide security assessment and verification mechanism for the Add-on security services along with the business requirements. The security assessment and verification is done automatically using Add-on security service assessment model. This issue plays a main role in verifying where the security model matches with the business requirements and also whether the security agreements are well maintained by both consumer and the service provider. We convert source code to first order logic in reasoning engine to evaluate the policy rules that influence the subject, resource and environment to determine the Access point in the security services and also finally evaluating the QOS like cost, response time, execution time and uptime for business services along with Add-on security features.

General terms: Business logic, XML schema, Finite State Machine, first order logic.

Keywords: Add-on security model, Service Level Agreement, Security assessment and verification.

1. INTRODUCTION
A business is a commercial entity that sells products or services to a consumer. Each business has specific functional requirements, which are executed to create these products and services. Business logic is best defined as a formal functional activity that is performed based on specific operating procedures within a business. Business process modeling is a technique of designing and capturing the business logic of a company. Understanding the internal business procedures is critical in the determination of how an organization can become more efficient. These business models give companies a tool to communicate their current procedures in a standard manner. All processes within an organization consist of multiple layers of business logic. Imagine the process that is required to access a computer. Typically the computer requires the user to enter a name and password. This name and password combination is then verified and validated. Incorrect passwords cause the computer application to prompt the user to re-attempt the required credential business logic process. Business logic is also used in business process re-engineering. This is a technique used by many organizations to streamline and improve existing business processes. Manufacturing is an industry that typically requires specific steps to complete a process. Improving these steps can increase productivity and profits for the organization. The business rules of an organization are typically considered proprietary. This is because they help make a company more profitable and unique within an industry. Proprietary business logic is the secret sauce for many organizations. Business-driven development and management of secure applications and solutions is emerging as a key requirement in the realization of an on demand enterprise. In a given enterprise, individuals acting in various roles contribute to the modelling, development, deployment, and management of the security aspects of a business service. We look at the business service life cycle and propose an Add-on security model for business service. At the time of purchase the consumer buys business service alone. As they use those business services in day to day life they feel some security holes present in it and they need some security essentials to fill those gaps. So the consumers request the service provider to provide the suitable security service and hence the security service acts as an add-on security model to the existing business services. The advantage of add-on security model is it minimizes the cost because purchasing a new business service with security model charges high where as adding a security services to existing business services is economically low.

2. RELATED WORKS
Christian Wolter present security policy and policy constraint models and discuss a translation of security annotated business processes into platform specific target languages, such as XACML or AXIS2 security configurations. To demonstrate the suitability of this approach an example transformation is presented based on an annotated process [1]. Alessandro Armando and Roberto Carbone present an approach to security testing of web-based applications in which test cases are automatically derived from counterexamples found through model checking [2]. Muhammad Qaiser Saleem has highlighted the security problems for SOA based applications and few Model Driven Security Frameworks are presented to develop secure software applications [3]. Wei She proposed an enhanced security

model to facilitate the control of information flow through service chains. It extends the basic security models by introducing the concepts of delegation and pass-on. Based on these concepts, new certificates, certificate chain, delegation and pass-on policies, and how they are used to control the information flow is discussed [5]. Li Jiang designed a WS-Security Evaluation Model, it's provide a valuable way to help user to create the threat modelling and evaluating the safety degree for Web service security. With the case study of SOA system in a certain enterprise, experimental results show that it provides a valuable reference to check out security vulnerabilities of Web service and optimize the system's security design [6]. Michael Menzel introduced the model-driven approach that facilitates the transformation of architecture models annotated with simple security intentions to security policies. This transformation is driven by security configuration patterns that provide expert knowledge on Web Service security. Hence they introduced a formalised pattern structure and a domain-specific language to specify these patterns [7]. Efforts on achieving model-driven development of web services already exist. However, there is currently no complete solution that addresses non-functional aspects of these services as well. Juan P. Silva Gallino presents an ongoing work which seeks to integrate these non-functional aspects in the development of web services, with a clear emphasis on security [8]. L. Boaro and E. Glorio present an integration of model checking and semantic reasoning technologies in an efficient way. This can be considered the first step towards the usage of semantic model checking in problems of security verification. The approach relies on a representation of services at the process level that is based on semantically annotated state transition systems (ASTS) and a representation of specifications that is based on a semantically annotated version of the computation tree logic (AnCTL). The proposed approach permit to perform an efficient and yet useful, semantic reasoning at process-level about web services [9]. Xinwen Zhang proposed group-based RBAC model (GB-RBAC) and applied it for authorization management in collaborations by introducing the concept of virtual group. A virtual group is built for collaboration between multi-groups, where all members build trust relation within the group and are authorized to join and perform operations for the collaborative work [10]. Jian Cao proposed an organizational model and an authorization model for supporting dynamic business processes. More specifically, authorization policies are expressed in an SQL-like language which can be easily rewritten into query sentences for execution. In addition, the framework supports dynamic integration and execution of multiple access control polices from disparate enterprise resources [11]. The security problem of the Web service is becoming the key problem for the Web service development and application. Ke Ma and Chang-xin Song does a deep research on the security communication strategy of SOAP information, at the same time, describes identity validation and access control in detail. After that, they discuss a Web service security frame structure based on WS-Security technology. According to this, a Web service security model is designed, and all the modules and flows in this model are introduced in detail [12]. Meiko Jensen and Sven Feja present an extension to the ARIS SOA Architect that is capable of modelling security requirements as a separate security model view. Further they provide a transformation that automatically derives WS-Security Policy-conformant security policies from the process model, which in conjunction with the generated WS-BPEL processes and WSDL documents provides the ability to deploy and run the complete security-enhanced process based on Web Service technology [13].

3. PROPOSED WORK

3.1 Accessibility in Business Logic

Accessibility is a general term used to describe the degree to which a product, device, service, or environment is available to as many people as possible. Accessibility can be viewed as the "ability to access" and possible benefit of some system or entity. Accessibility is often used to focus on people with disabilities or special needs and their right of access to entities, often through use of assistive technology. Accessibility to business logic Provides regulated access to the business resources which business experts and analyst need to perform their duties i.e. to change the policies and rules in the business logic of the service. In order to make changes in the business logic business experts and analyst must follow some common mechanism that permits management to specify what the business expert and the analyst can do i.e. which resources they can access and what operations they can perform on a system. In any organization planning to implement accessibility should consider the abstractions like policies, mechanisms.

Accessibility policy should be brief which is highly recommended and set at high level. It must also specify how access is managed and who under what circumstances may access what information. The accessibility policies are enforced through accessibility mechanism and they are direct implementations of formal accessibility policy concepts. Finally translate a user's access request in terms of a structure that a system provides for example, a simple table lookup can be performed to grant or deny access.

The three main security goals also pertain to accessibility:

- Confidentiality: only authorized users can gain access to protected information
- Integrity: only authorized user can access and modify protected information
- Availability: the services of the system is accessible to authorized users, when required

3.2 Policy based Access Control

The Policy-Based Access Control pattern decides if a subject is authorized to access an object according to policies defined in a central policy repository. It is also a strategy for managing user access to one or more systems, where business classification of users is combined with Policies to determine what access Privileges a user should have. Theoretical privileges are compared to actual privileges, and differences are automatically applied to Managed Systems. The Policy Based Access Model is a set of rules that determine access and also known as RSBAC (Rule Set Based Access Control). It is a set of rules that determine access rights based on the Object, rather than the Subject.

The policy-based access control provides a powerful and flexible means of protecting data, down to the row level. The main important metrics of PBAC is by having Objects as the Base for Accessibility i.e. Dynamic Change in environment/ System requirements is adapted very easily. A Method of Policy Management independent from business logic can be developed. Policy-based access control makes a strict distinction between the formal statement of the policy and its enforcement. Formal Statement of Policy is usually based on a specification language which should be expressive enough to easily formulate the policy rules. Enforcement is a mechanism capable of intercepting access attempts, evaluating them

against the policy, and accordingly granting or denying the access.

3.3 Attribute based Access Control (policy based)

A new access control approach for Service Oriented Architectures (SOA) is Attribute Based Access Control (ABAC). Attribute-Based Access Control (ABAC) uses attributes as building blocks in a structured language used to define access control rules and to describe access requests. Attributes are sets of labels or properties which can be used to describe all the entities that must be considered for authorization purposes. This model controls access based on properties of subjects or objects. It is used in environments where some subjects may not be pre-registered. Attribute-based systems have desirable properties such as flexibility, privacy and intuitiveness.

The ABAC model brings many advantages over traditional identity or role based models

- Intuitive to model and manage real-world access control policies
- More flexible and more powerful to represent complex, fine-grained access control semantics, which is especially suitable for the dynamic SOA / web services environment
- Management of security information is spread over a number of Attribute and Policy Authorities, possibly across organizational boundaries – suitable for large-scale information sharing
- Reduces overall system complexity, allowing different system components (user directory, service registry, policy server, etc.) to focus on their respective administrative tasks

To utilize the ABAC model to its full potential, many aspects of the entire attribute management "life cycle" needs to be considered, such as attribute provisioning, binding, discovery, and feedback loop.

3.4 Business Logic Model for Schema Generation with Security Attributes

Business Logic is one way of looking at a business to get a feel for what the business is trying to do and how it is trying to do it, as a business. It can also help you see, from a manager's perspective, whether it logically hangs together, or whether some of the business' major premises are at war with each other. The scope of Business Logic Model is to analysis the business logic in all aspects and to generate the xml schema for easy management and better understandable. The author had already published the model [4]. The model produced the business logic schema and then converts into security assessment schema and the invocation of service logic flow execution is done only if it satisfies all the security essentials.

The scope of Business logic:

- models real life business objects (such as accounts, loans, itineraries, and inventories)
- prescribes how business objects interact with one another
- enforces the routes and the methods by which business objects are accessed and updated

Business logic comprises:

- business rules that express business policy (such as channels, location, logistics, prices, and products); and
- workflows that are the ordered tasks of passing documents or data from one participant (a person or a software system) to another.

An organization can be designed as a set of Business Functions and usually the structure of the organization units within an organization is closely based on the business functions. Some of the functions are get, select, store, compare, return, display, catch, etc. The parameter consists of Access control types such as source, resource and environment, these should be associated with level of accessibility it requires.

In fig 1: Service logic is classified into system logic and business logic. The business logic gives detail information flow of the organization so to check the security of the business logic is very important task. The business logic is decomposed into rules, functions and domain variable. A business rule is a statement that gives constrains to some aspect of the business that influence the behaviour of the business. It focuses on the policies of the organization. It is essential for an organization in achieving its goals.

The business rules cannot be broken down further. Business rules express business policy such as channels, location, logistics, prices, and products. A function is a concept used in the Organization Architecture domain and represents what work is done by that organization, or business role. An organization can be designed as a set of Business Functions and usually the structure of the organization units within an organization is closely based on the business functions. Some of the functions are get, select, store, compare, return, display, catch, etc. Domain variable consist of business variable present in the business logic. Hence to evaluate the security of the business logic analysis is done using security analyzer and dependency analyzer process. Entity attribute is associated with a subject such as user, application, process that defines the identity and characteristics of the subject. Resource attribute is associated with a resource such as web service, system function, or data. Environment attribute describes the operational, technical, or situational environment or context in which the information access occurs such as current date time, current threat level. Finally security assessment is done through access point and basic flow. The basic flow gives the detail work flow of the business and from the security assessment the security report is generated. The security report contains issues about the security level of the services and gives varies solution to overcome the security holes and threats in the business services.

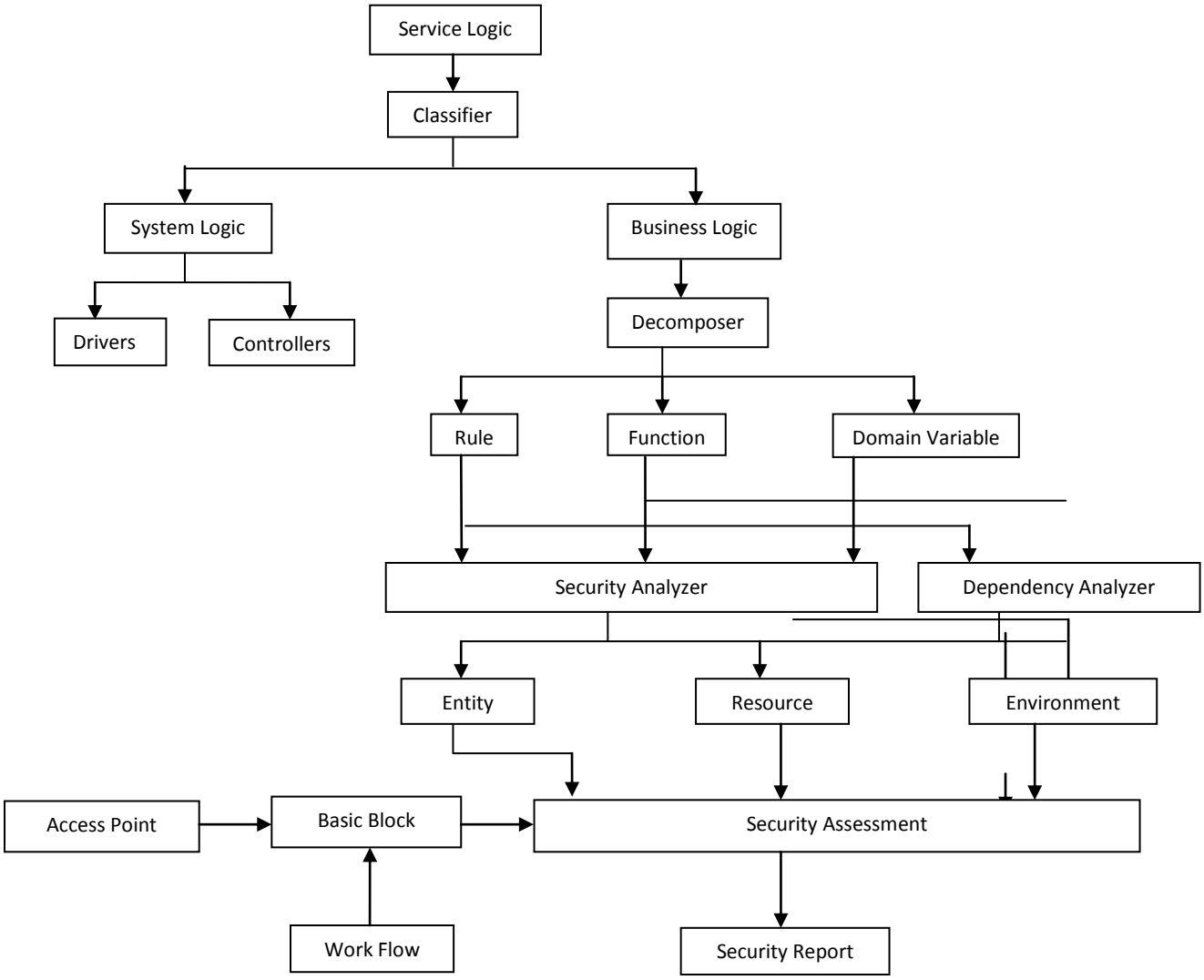


Fig1: Identifying the Access Point in Business Logic

3.5 Policy Rule example for Withdrawal

R1: Check whether the Account No exits
R2: Authentication the user for requested operation
R3: check whether the sufficient amount exit to perform the operation
R4: Check the Minimum balance Rs 500 is maintained by the user
S1: Account no exits
S2: Authenticate the user
S3: Check whether the sufficient amount exits
S4: Check Minimum balance

$\sum$-$r_{1:}$ condition doesn't satisfies so return to S1
$\sum$-$r_{2:}$ condition doesn't satisfies so return to S2
$\sum$-$r_{3:}$ condition doesn't satisfies so return to S3
$\sum$-$r_{4:}$ condition doesn't satisfies so return to S3

The finite state machine consists of four state S1, S2, S3 and S4 each node represent the first order logic for the related rules. The r1, r2, r3 and r4 is the accessibility specification for corresponding rules R1, R2, R3 and R4 that contain Access control attributes like Subject, Resource, Environment and their required Access control.

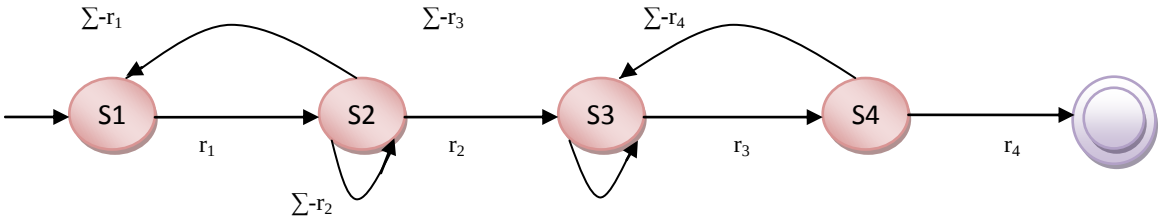


Fig 2: Simulating FSM for Withdrawal Service

Table.1 Mapping from source code to First Order Logic

RULE	CODE	FIRST ORDER LOGIC
R1	get(String account) Select * from withraw where account no like {$Keyword}	*∃a[account(a)] →w[withdrarw(w)] ∧ a[account_no(a)]*
R2	get (user id,pin no) if(user.equals(user id) && pass.equals(pin no)) { out.println("Login Successful"); } else out.println("Login Failed,Please try Again"); }	*∃y[login(y)]∧ z[password(z)] →login success(y,z).* *Or* *∃y[login(y)]∧ ≠z[password(z)] →login success(y,z).*
R3	get amount if (bal < amt) { out.println("Not sufficient balance."); return 1; } if (amt < 0) { out.println("Invalid Amount"); return 1; } bal = bal - amt; return 0; }	*∃a[amt(a)] ∧∀ b[bal (b)< amt(a)] ∧∀ a[amt (a)<0] → insufficient balance(b,a) ∧ invalid amount (b, a)* *Or* $\exists\ b[bal(b)] \rightarrow b[bal(b)] - a[amt(a)]$
R4	get (amount) if (checkamt>=min_bal) { Balamt[accont no]=checkamt; Out.println("Balance Amount:"+balamt[account no+"\n"]); } else { Out.println("\n As per Bank rule you should maintain minimum balance of Rs 500\n") }	*∃a[amt(a)] ∧∀ c[check_amt (c)≥ mb [min_bal (mb)] ∧ ba[bal_amt(ba)]= c[check_amt (c)] →status[balance amount ∧ account no (ba, ac_no)*

Table.2 Basic Block and Flow Graph with Access Point identifier

Block1 [B1]

get(String account) **AP_1:** Select * from withdraw where account no like {$Keyword}	<query type=”Data retrieval” name=”select” table=”account”> <rowset condition= “accno=+accno “> <columns> <column_name>name </column_name> <column_name>id </column_name> </columns> </rowset> </query>

Block2 [B2]

get (user id,pin no) **AP_2:** if(user.equals(user id) && pass.equals(pin no)) { out.println("Login Successful"); return(“Yes”); } else out.println("Login Failed, Please try Again"); return(“No”); }	<if condition=”user id=pin no”> <process> <ouput_statement> <print>“Login successful”</print> </output _statement> </process> <return> Yes </return> </if> <else> <process> <assignment_statement> <print>“ Login Failed, Please try Again”</print> </assignment_statement> </process> <return> No </return> </else>

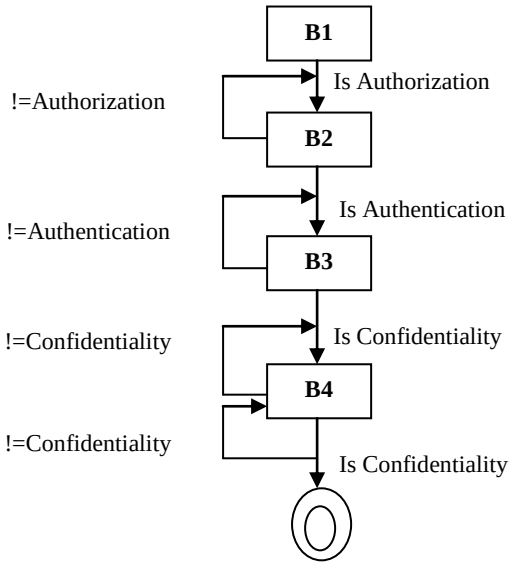
Block3 [B3]

get amount **AP_3:** if (bal < amt) { out.println("Not sufficient balance."); return (“No”); } else if (amt < 0) { out.println("Invalid Amount"); return (“No”); } bal = bal - amt; return (“Yes”); }	<if condition=”bal<amt”> <process> <ouput_statement> <print>“ Not sufficient balance”</print> </output _statement> </process> <return> No </return> </if> <else if condition=”amt<0”> <process> <assignment_statement> <print>“ Invalid Amount”</print> </assignment_statement> </process> <return> No </return> </else if> <else> <process> <assignment_statement> <LHS> bal </LHS> <RHS> bal-amt </RHS> </assignment_statement> </process> <return> Yes </return> </else>

Block4 [B4]

get (amount) **AP_4:** if (checkamt>=min_bal) { Balamt[accont no]=checkamt; Out.println(“Balance Amount:”+balamt[account no+”\n”]); return (“Yes”); } else { Out.println(“\n As per Bank rule you should maintain minimum balance of Rs 500\n”) return (“No”); }	<if condition=”checkamt>=min_bal”> <process> <ouput_statement> <print>“ Balance Amount:”+balamt[account no+”\n”</print> </output _statement> </process> <return> Yes </return> </if> <else> <process> <assignment_statement> <print As per Bank rule you should maintain minimum balance of Rs 500\n”</print> </assignment_statement> </process> <return> No </return> </else>

Fig.3 Flow graph with security holes identification



3.6 Add-on Security Service Assessment Model

The main goal of the Add-on security service assessment model is to provide the assessment report as a solution to the consumer in such a way they can identify the security holes and in what way it can be reduced. Security requirement is the key resource for the Add-on security service assessment model. The security requirement is commonly important for both the service consumer and service provider. The service consumer gives the needed security requirement and the policy for the security services based on the business context (i.e.) security differs for each business context. Hence the service consumer gives their demands to the service provider based on those requirements the service provider lookup into the service repository to find out whether any security services match with the security requirements if so the service provider gives the recommend system from the service registry to the service consumer or else a new system is build with the security requirements. Finally the service level agreement is maintained by both the service provider and the service consumer. The SLA is extremely important to business context to ensure effective engagements for any services or product delivery and it also clearly defines the level of service expected between two parties and also ensures these targets are met within expected standards and budgeted cost. The security assessment and verification is done through security assessment engine using logical analyzer and integration engine. In logical analyzer first is to analyze the access point for the business rules through applying access point analyzer and after determine the access point the main role is to build the basic block and flow graph and finally generating the business logic schema. The security assessment engine contains four life cycle process such as analyzer, verification, FOL mapper and finally reasoning engine. The analyzer evaluates the overall flow of the security assessment engine and also verifies whether the security is in places and covert the rules into first order logic using FOL mapper and finally the reasoning engine plays a important roles in verifying whether the security services match with security requirements. The integration engine is used to add-on the security services along with the business services using plug-in and wrappers and also evaluating both the services using evaluation engine and generating the assessment schema. The Add-on security service assessment model can be used in run time system, dynamic proxy, create audit log and to handle the exception.

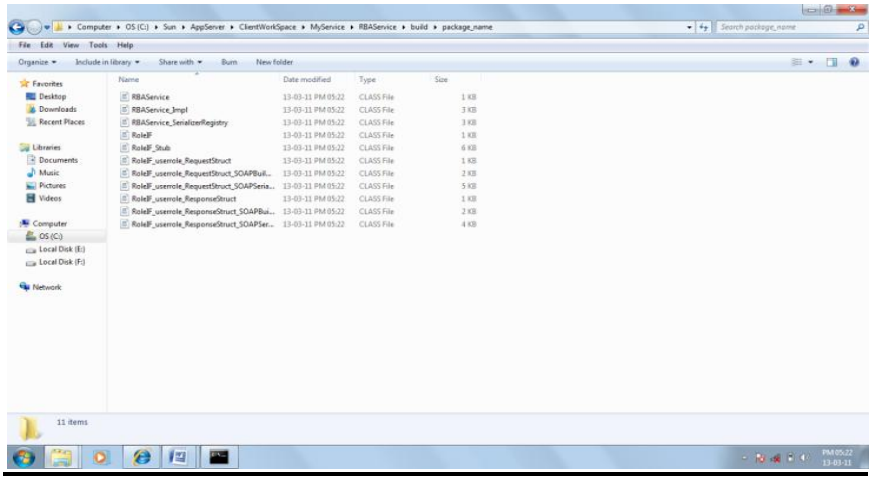


Fig 4: Business Service with Add-on security Deployed in Client Machine

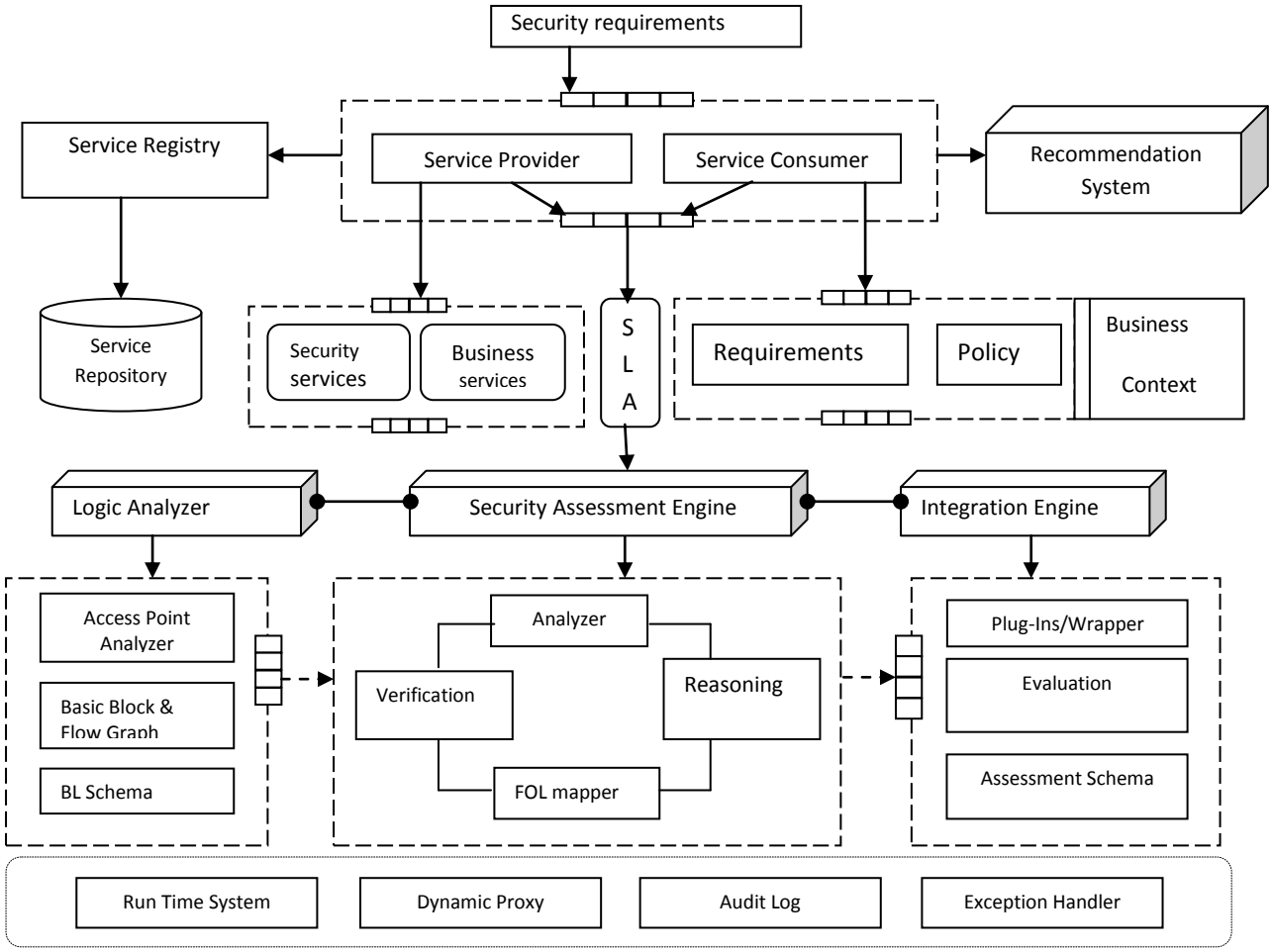


Fig 5: Add-on Security Service Assessment Model

4. PERFORMANCE EVALUATION

Table.3 Performance Evaluation for Business Services with Add-on Security Features

SERVICE NAME	WSDL PATH	COST ($)	UPTIME (Sec)	EXECUTION TIME (Ms)	RESPONSE TIME (Ms)
Airline Service International	http://localhost:8080/AirlineServiceInternational/airintservice?wsdl	7960	24	214	980
Airline Service Domestic	http://localhost:8080/AirlineServiceDomestic/airlineservice?wsdl	6719	21	74	968
Railway Service	http://localhost:8080/RailwayService/railservice?wsdl	5259	18	128	1067
Travel Service	http://localhost:8080/TravelServiceA/reservationservice?wsdl	6701	16	90	826
Search Content Service	http://localhost:8080/SearchContentService/getsearchcontent?wsdl	4850	18	20	634
Advance Search Service	http://localhost:8080/AdvanceSearchService/getadsearch?wsdl	5146	16	5	847
Login Service	http://localhost:8080/LoginService/getlogin?wsdl	4526	14	22	698
RBA Service	http://localhost:8080/RBAService/getrole?wsdl	4649	17	15	716
RSA Service	http://localhost:8080/Myrsaserver/rsaservice?wsdl	7011	10	25	607
User Registration Service	http://localhost:8080/UserRegistration/registrationservice?wsdl	5467	21	86	706
User Registration Service Optimal	http://localhost:8080/OptUserRegistration/userregistration?	5187	18	63	624
Email Registration Service	http://localhost:8080/EmailRegistrationService/emailregistration?wsdl	5390	18	25	956
Email Registration Service Optimal	http://localhost:8080/OptEmailRegistration/emailregistration?wsdl	5259	16	21	803

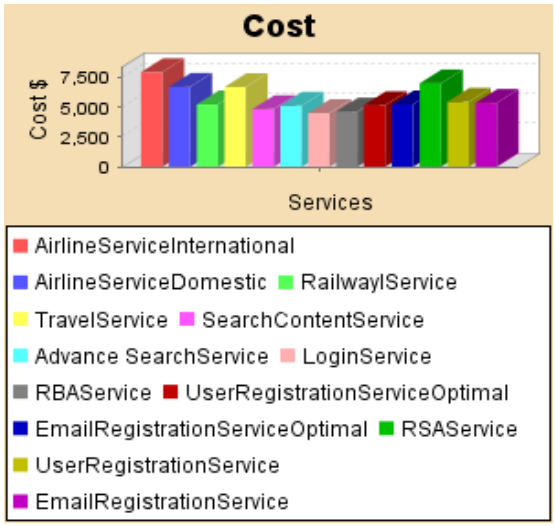


Fig 6: Graph for evaluating Cost

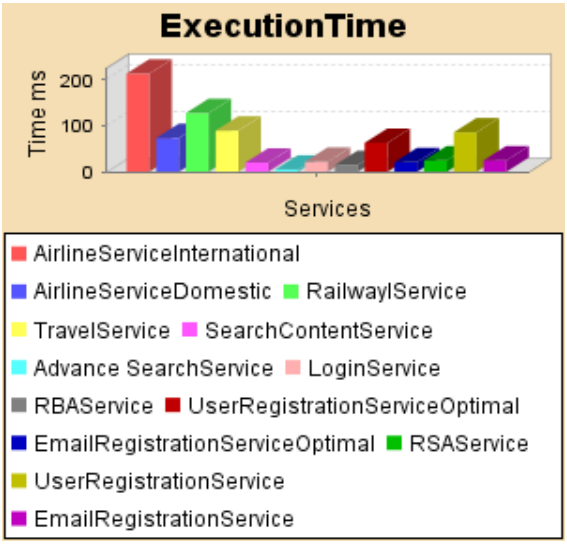


Fig 7: Graph for evaluating Execution time

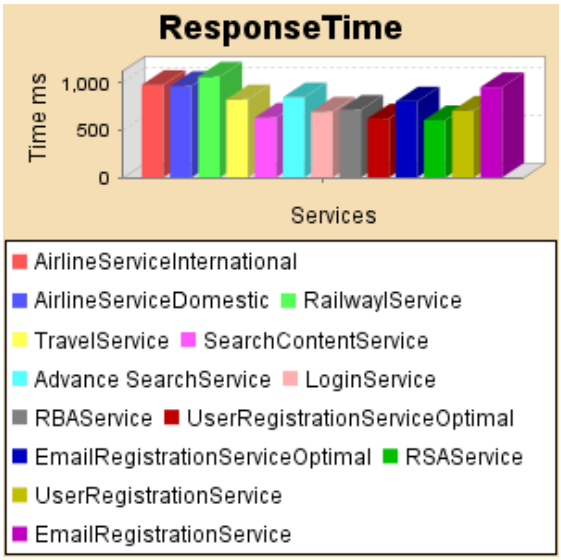


Fig 8: Graph for evaluating Response Time

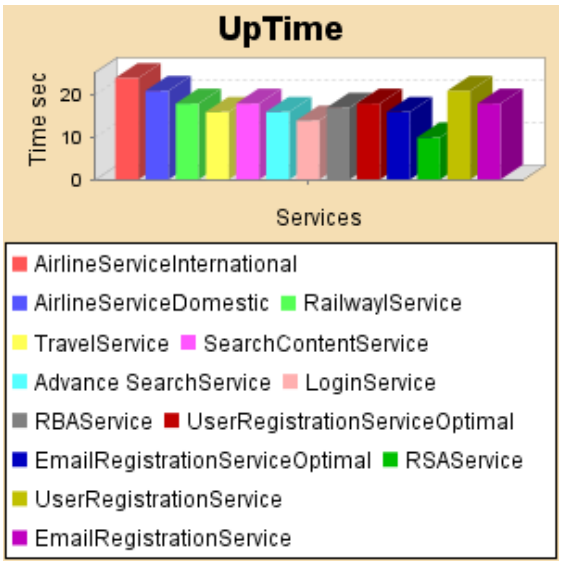


Fig 9: Graph for evaluating the Uptime

5. CONCULION

In this paper, we proposed an Add-on security model which provides interoperable security services for the business services according to the security requirements of the business. We also establish the model as schema driven security model which facilitate dynamic integration of security services with the associated business services and finally to provide security assessment and verification mechanism for the Add-on security services along with the business requirements. The security assessment and verification is done automatically with Add-on security service assessment model. To access the business logic of the service the access point is identified and mapping is done from source code to FOL. Finally the flow graph is drawn with security holes identification. The performance evaluation is done for business services with Add-on security features.

6. REFERENCES

[1] Christian Wolter, Michael Menzel, Andreas Schaad, Philip Miseldine and ChristophMeinel, "Model-driven business process security requirement specification", (ELSEVIER) Journal of Systems Architecture 55, 2009.

[2] Alessandro Armando, Roberto Carbone, Luca Compagna, Keqin Li and Giancarlo Pellegrino, "Model-checking Driven Security Testing of Web-based Applications" Third International Conference on Software Testing, Verification, and Validation Workshops, (ICSTW)2010.

[3] Muhammad Qaiser Saleem1, Jafreezal Jaafar2, Mohd Fadzil Hassan3, "Model Driven Security Frameworks for Addressing Security Problems of Service Oriented Architecture, IEEE 2010.

[4] Thirumaran., Dhavachelvan.P, Asha.T and Lakshmi.P, "Framework for managing Business logic of web services through Schema generation and Property evaluation ", International Journal of Computer Applications,2010.

[5] Wei She, I-Ling Yen, and Bhavani Thuraisingham, "Enhancing Security Modeling for Web Services using Delegation and Pass-on", IEEE International Conference on Web Services, 2008.

[6] Li Jiang, Hao Chen, Fei Deng, "A Security Evaluation Method Based on STRIDE Model for Web Service", IEEE 2010.

[7] Michael Menzel, Robert Warschofsky and Christoph Meinel, "A Pattern-driven Generation of Security Policies for Service-oriented Architectures", IEEE International Conference on Web Services, 2010.

[8] Juan P. Silva Gallino, Miguel A. de Miguel, Javier Fernández Briones, and Alejandro Alonso, "Model-Driven Development of a Web Service-Oriented Architecture and Security Policies", 13th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing, 2010.

[9] L. Boaro, E. Glorio, F. Pagliarecci and L. Spalazzi, "Semantic Model Checking Security Requirements forWeb Services", IEEE 2010.

[10] Qi Li, Xinwen Zhang, MingweiXu and Jianping Wu,"Towards secure dynamic collaborations with group-based RBAC model", Computers & Security 28, 2009.

[11] Jian Cao,JinjunChen, HaiyanZhao and MingluLi, "A policy-based authorization model for workflow-enabled dynamic process management", 32nd Journal of Network and Computer Applications, 2009.

[12] Ke Ma and Chang-xin Song, "Research on a Web Security Service System Structure Model", International Conference on Advanced Computer Theory and Engineering, 2008.

[13] Meiko Jensen and Sven Feja, "A Security Modeling Approach forWeb-Service-based Business Processes", 16th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems, 2009.