Multi-level Security for Integrated Financial Mobile Web Services using Federated ESB

Dr. S. Britto R.Kumar
Department of Computer Applications
Bishop Heber College
Tiruchirappalli, Tamilnadu, India

Dr. S. Albert Rabara
Department of Computer Applications
Bishop Heber College
Tiruchirappalli, Tamilnadu, India

ABSTRACT

The Wireless and mobile devices, and their applications often run on different platforms, which can make integration problematic. Enterprise-service-bus (ESB) approach and service-orchestration platforms through mobile agents might offer good solutions. However, the centralized ESB introduces serious limitations and the federated ESB architectural pattern conquers those limitations by partitioning the infrastructure into separate ESBs.

Security is one of the greatest concerns, especially when integrating cross-organizational applications. When the technology changes the adaptability of mobile devices, network services and service providers, it is vital to design an integrated architecture for mobile web services. This paper proposes an architecture that provides multi-level Security for integrated financial mobile web services using federated ESBs. This architecture enables service integration, network integration and supports access to financial services from any mobile devices using any mobile technologies.

General Terms

Web Services, Security, Enterprise Service Bus.

Keywords

Federated ESB, Mobile Web Services, Multi-level Security, Financial services.

1. INTRODUCTION

Mobile Commerce is an emerging discipline that involves mobile devices, applications, middleware and mobile networks. The rapid growth of wireless networks and services fueled by next generation mobile applications research has ushered in the area of ubiquitous computing. Light weight portable computers, IP based office and home appliances, and the popularity of Internet are strong forces to the service providers to support seamless user mobility.

The Wireless and mobile communication infrastructures have entered an extraordinary era that provides new opportunities for an organization to increase its productivity, efficiency, and effectiveness. The mobile communication creates strategic differentiation in highly enraged and competitive market places and enables better and more customized communications to support the customers and vendors. Mobile devices are now prevailing and are already being successfully deployed by some of the world's most innovative and competitive organizations.

Mobile Web Services has become so important over the past few years that promise to become a global business paradigm of excellence in the near future. Web services provide a standard mechanism for communication between the services. This standardization is independent from the implementation of applications and network infrastructure under which they communicate. However, while considering the development of applications and services, the traditional approach to service orientation (SOA) has constantly failed to deliver the business value due to increase in SOA complexity.

The application development is really a costly process, and software requirements are only increasing. Integration, availability, reliability, scalability, security, and integrity are becoming more complicated issues, even as they become more critical. Most of the solutions today require the use of a collection of technologies, not only one. At the same time, the cost to maintain existing software is rising. With regard to service-oriented architecture (SOA), when an organization has many connected services, the logical network can become extremely difficult to manage. Also, Enterprise Application Integration (EAI) technology failed to address the current complex array of integration issues and also failed to deliver the expected business flexibility[1]. This can be improved by using an enterprise-service-bus (ESB) approach and service-orchestration platforms.

An ESB abstracts a set of fundamental capabilities such as protocol mapping, service choreographies, line-of-business (LOB) adapters, message distribution, transformations, and durability. An enterprise service bus (ESB) enables a business to make use of a comprehensive, flexible and consistent approach for integration while also reducing the complexity of the applications being integrated[2]. ESB unifies message oriented, event driven and service oriented approaches for integrating applications and services. Implementing an ESB supports greater reuse of IT assets by separating application logics and integration tasks that reduce the number, size, and complexity of integration interfaces. While considering integration, Virtual Enterprise Integration (VEI) is effectively accomplished through service-oriented architecture and ESB using web services and business process engines that execute WSDL and WS-BPEL [3].

However, the ESB introduces serious limitations in aspects such as management, performance, and scalability [4,5]. Since the large organizations met several failed attempts to implement centralized ESBs, the industry has moved to a more agile pattern in which the functionality is partitioned across multiple lightweight physical ESBs that are grouped as a federated entity. This pattern is commonly known as federated ESB. The federated-ESB pattern conquers the limitations of the centralized ESB model by partitioning the infrastructure into separate ESB that can be scaled and configured separately.

As the technology changes the adaptability of mobile devices, network services and service providers are major bottlenecks for the common man. While mobile network providers frequently changing their scenarios and operations, it is vital to design an integrated architecture for mobile applications especially for financial services which can be adaptable by any device, any network and any service etc. This paper proposes an architecture that provides multi-level Security for integrated financial mobile web services using federated ESBs. The proposed architecture enables service integration, network integration and supports access to financial services from any mobile devices.

The rest of the paper is organized as follows: Section 2 represents the literature review on service integration using ESB and existing security threats. Section 3 proposes architecture for multi-level Security for integrated financial mobile web services using federated ESB. Section 4 suggests the required level of security for integrating financial mobile web services using federated ESB architectural pattern. Finally, Section 5 concludes the paper.

2. LITERATURE REVIEW

Web services technology has revolutionized the software industry dramatically by the development and integration of a variety of enterprise applications thereby enabling the web users to access them. The service-oriented architecture (SOA) has been successfully applied in enterprise environments for service composition together with Enterprise Service Buses (ESBs) integration pattern.

Kristijan et al. proposed a new approach for cross-domain service integration through an automated federation of Enterprise Service Buses (ESBs). ESBs are the mediation centers within a service domain that enable service interaction across technological boundaries by using service proxies. The authors also presented a configuration engine prototype namely, DISCE (Declarative Inter-ESB Service-Connectivity Configuration Engine) that enables an operator to configure service connectivity in a declarative form, by specifying simple rules. The engine produces a configuration consisting of set of proxies interconnecting clients and services. However, there is no discussion on security in the service proxies [6].

Wen Zhu and Walt Melo identified the challenges faced by organizations regarding agility, integration, security, standards-based technology implementation, and pressure to reduce expenditures. To address these challenges, the authors established Service Oriented Infrastructure (SOI) with emerging SOA technologies including Enterprise Service Business (ESB) and Business Process Management Systems (BPMS). They also examined three open source ESB platforms such as OpenESB, ServiceMix, and JBossESB and provided assessments from both architectural and functional perspectives. In particular, the focus was on the role of Java Business Integration (JBI), the standard basis for Java-based ESB, in such areas as reusability, interoperability, and maintainability. However, ESB provides some serious limitations [7].

Yan Liu presented modeling-based approach to coordinate the process of web service management, configuring parameters or invoking tasks for loan-brokering web service on ESB. This approach uses executable process models to represent diagnosis logic and orchestrate the replacement. The process models can interact with the system and accept administrators' instruction at the process level. However this approach is implemented on an ESB architecture that affects the web services management, performance, and scalability [8].

Security is one of the quality attributes that is of the greatest concern, especially when integrating cross-organizational applications. Enabling security for applications in the context of cross-organizational integration is required to provide a set of runtime security protocols and prebuilt security services.

Most of the security breaches on financial web services used today are based on deceiving the user to steal login data and valid PIN, and for these, the security attacks are phishing and pharming. Phishing is a fraud utilizing social engineering. It is designed to trick users to reveal sensitive personal information such as login IDs, passwords, and credit card details to fraudsters. Cross-site scripting and Trojan horses can also be used to steal login information [9].

The threats posed to web server fall into two categories: threats from an actual attacker and threats accruing from technological failures. The direct threats to payment servers are Malicious Code threats that include viruses, Trojan Horses, Logic Bombs, transmission threats such as Denial of Service (DoS) attacks, Ping of Death Attack, and SYN Flooding. The other threats to payment servers are Data Packet Sniffing, IP Spoofing, and Port Scanning. The network security threats are classified further into interruption, interception, modification, and fabrication which also are in the design and development of mobile payment system [10].

The standard protocol for communication between the browser and the web server is https, which is the http protocol on the top of a Secured Socket Layer(SSL) certificate. The use of SSL server certificate at the web site should allow the user to determine who is communicating with the financial application web server [11].

Fraudulent web sites use a number of different techniques to hide the fact that they are not authenticated including overwriting or disguising the true URL shown in the browser, overlaying the genuine web site with a crafted pop-up window, drawing fake padlock images on top of the browser window to give the impression that SSL is enabled, and registering SSL certificates for domain names similar to the real financial institutions. In practice, these tricks make it extremely difficult for the average user to distinguish a phishing site from a genuine.

Successful Web service security solutions need to leverage both transport and application layer security mechanisms to provide a comprehensive suite of security capabilities. The well known transport security mechanisms, Secure Socket Layer (SSL)/Transport Layer Security (TLS) alone, are not adequate for Web services [12]. Security at the application layer must be added in order to achieve end-to-end protection for mobile payment environments. SSL/TLS are designed for secure communication between two end points namely point-to-point. However, the Web services messages relay on physical network and application infrastructure. The messages are being transferred across various nodes and different protocols. Hence

Web services require end-to-end security rather than point-to-point security as provided by SSL.

XML-based Web services are becoming a standard technology for integrating applications and exchanging data in service-oriented architectures. However, XML-based Web services encompass the number of vulnerabilities by providing access to application APIs and target applications. In recent years, WSDL threats cause a series security breach if the Web services are compromised [13]. The following XML and Web services threats are more serious such as Parameter Tampering in WSDL document, Recursive Payloads, Oversized Payloads, Coercive Parsing, Schema Poisoning, SQL or XQuery Injection, Replay Attacks and XML Morphing.

In order to protect the XML messages, Simple Object Access Protocol (SOAP) is built on XML and HTTP protocols [14]. All type of XML data can be encrypted and decrypted easily by standard cryptographic techniques like RSA, DES, AES and Hash of a message calculated for integrity checker [15]. The web services authenticate to each other by passing security tokens in a standard SOAP message [16]. Tokens can include user name and password credentials, Kerberos tickets, and X.509 certificates etc [17].

3. PROPOSED MODEL

The exorbitant growth in the field of web services has challenged the business users, developers and researchers in the design and development of enterprise applications. In many aspects, enterprise applications have evolved into too complex to be really effective and agile. Most solutions today require the use of a collection of technologies, not only one.

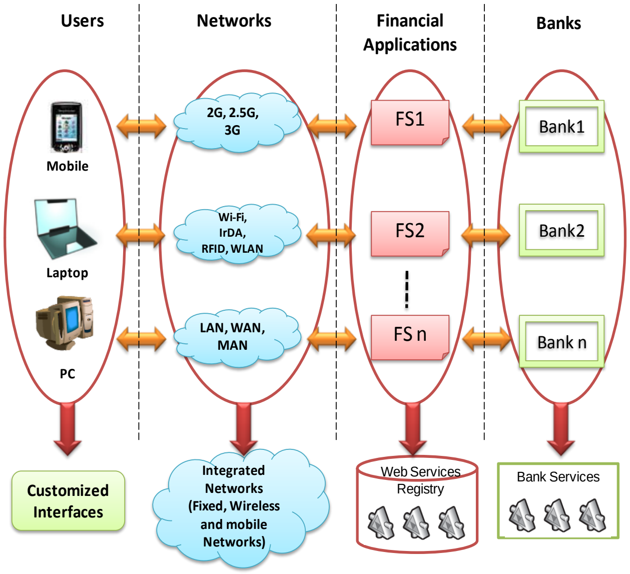

Fig 1: Logical Services Network

The traditional SOA systems share a set of characteristics, such as providing SOAP and WSDL as the fundamental standards for web service interfaces; enabling enterprise services using web service protocols, deploying a centralized ESB for abstracting the different service orchestrations, integrating complex business processes, enable the management of the entire SOA using a SOA-governance tools. However, when enterprise has many connected services, the logical connectivity among them can become extremely difficult to manage with regard to service-oriented architecture (SOA).

In Fig.1, each circle has a number of different services. The problem with this services network is that all of these services are directly interlinked and have too many point-to-point connections between services. However, when the SOA has too many services, the complexity of Service Oriented Infrastructure (SOI) evolves and grows. This approach is unmanageable. Also, traditional SOA causes enormous challenges in areas such as interoperability, performance, scalability and management. These challenges are a direct consequence of the lack of constraints in SOA systems. Architecture styles that do not impose constraints in the underlying domain quite often produce complex and unmanageable systems. In order to simplify the capabilities of SOA and focusing on the important aspects such as interoperability, performance and scalability, a new model has been developed by using an enterprise-service-bus (ESB) approach and service-orchestration platforms.

Although there is no formal definition and standard for ESB, an ESB abstracts a set of fundamental capabilities such as protocol mapping, service choreographies, line-of-business (LOB) adapters, message distribution, transformations and durability. ESB is used to abstract the communication between services and system, thus providing the ESB as a central backbone of the enterprise.

Fig.2 represents an ideal model on which messages are sent to an integrated web service manager and from there distributed to the final services. This model integrates the services that are provided by telecommunication service providers (TSP), application service providers (ASP) such as PayPal and Paymate and financial service providers (FSP) like banks. While considering service requesters, namely the Mobile Clients (MC) must have an account in any bank. The client has made a request for registration to avail the mobile financial services using SMS service with the bank. The bank server also allows the clients for registration over the Internet or directly. Once the registration process is completed, the bank server sends the AcCode, username and password to the client’s personal mail id together with the financial application software. The client can download software using any mobile network services such as GSM, GPRS or through data cable. During the payment process, the client device uses the username and password for authentication.

Mobile communications have entered an extraordinary era and more is expected to be realized in the few years to come. Evolutionary developments are made possible by technology giants such as wireless carriers known as Second-and-a-half generation (2.5G) services which represent a major evolutionary step towards the Third-generation (3G) services that will provide additional capabilities such as streaming, audio and video to meet future evolving communication needs. The new generation of wireless devices being introduced to the market for 2.5G Global System for Mobile Communications (GSM) and General Packet Radio Service (GPRS) services is designed to support mobile applications, with features ranging from small standard keyboards to high resolutions rich colored screens.

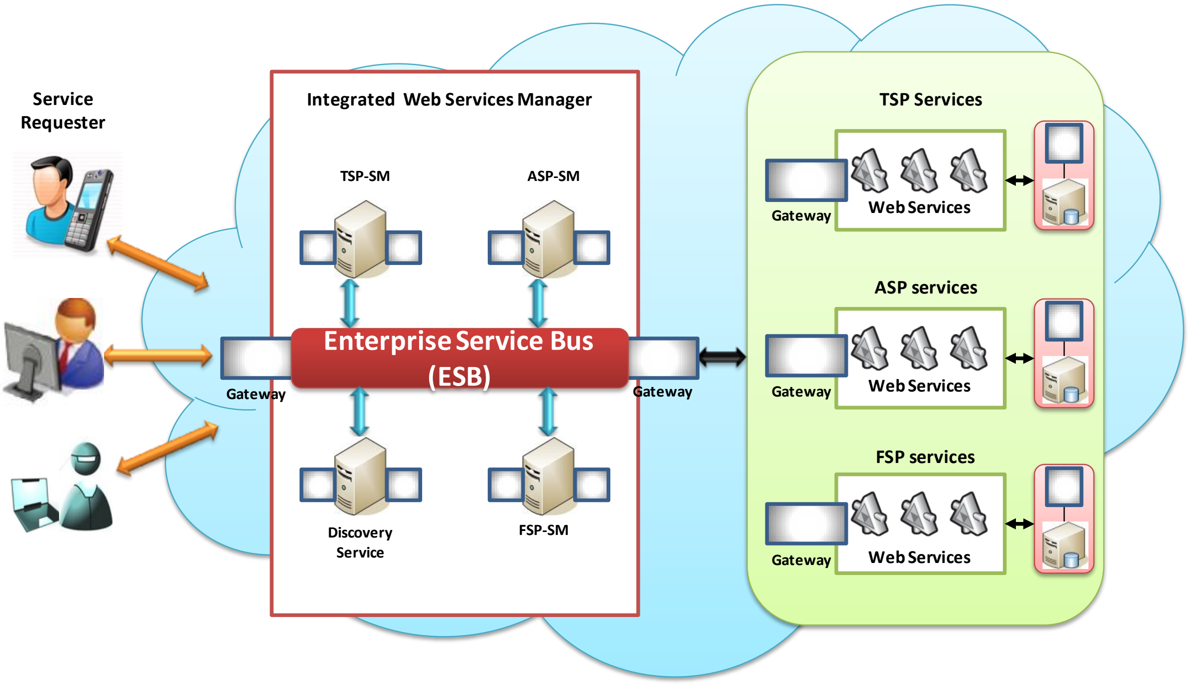

Fig 2: Central ESB for Integrated Financial Mobile Web Services

Global System for Mobile Communications is a common standard issued by European Telecommunication Standards Institute (ETSI) and is the first digital mobile network architecture put into practice. The tremendous market growth of GSM systems indicates the growing importance of mobile communications and an eminent need of security in mobile telephones during international communications.

The General Packet Radio Services (GPRS) of the GSM is designed to provide packet data services over the radio interface. GPRS supports common packet data protocols like IP and X.25. Being a packet data service, GPRS optimizes the use of network and radio resources. GPRS offers the possibility to charge by amount of data transferred rather than the time the mobile phone is attached to the network. It provides a greater benefit to the mobile users economically. In order to provide a cost effective payment system, the proposed architecture makes use of GPRS technology. The mobile clients need to open the GPRS service by applying to the telecom companies. The main advantages of the GPRS are that it can exchange large quantum of data with high speed and the connection is stable. GPRS supports several qualities of data services such as reliability and response time.

The integrated web service manager (IWSM) keeps a record of all registered clients. The authentication of the client is done by the IWSM by verifying the username and password. After authentication, IWSM identifies the respective application service provider together with mobile communication services and financial service providers using service discovery. Once found, the request is sent as service to the relevant ASP and FSP via TSP. The information flow on the network is secure by encrypting and decrypting the message using public key infrastructure. The IWSM web server accepts all the requests that come from the mobile clients and handles device and user authentication. The IWSM database server keeps the student and payment related information.

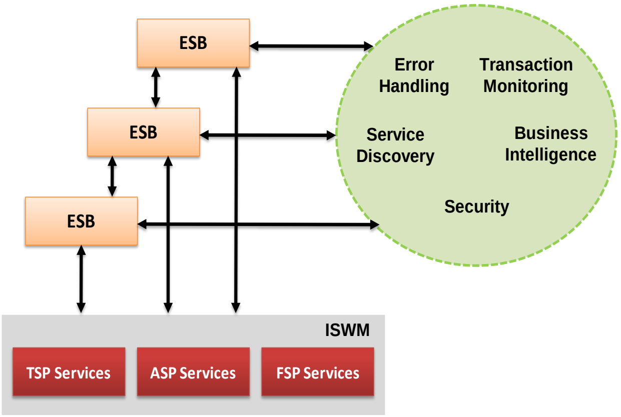

Fig.3: Federated ESB Pattern

While using large SOI and a central ESB architecture introduces serious limitations in aspects such as management, performance, and scalability. The ability to centralize very smart functionalities such as message routing, transformation, and workflows is as appealing as it is unrealistic in medium-to-large-enterprise environments. Essentially, by relying on the central ESB, it constrains the options for scalability and management of the enterprise solutions that leverage SOA infrastructure. After several failed attempts to implement centralized ESBs for integrated financial services, the previous model is improved, in which the functionality is partitioned across multiple lightweight physical ESBs that are grouped as a federated entity. This pattern is commonly known as federated ESB and represents one of the emerging architecture styles for building highly scalable ESB solutions

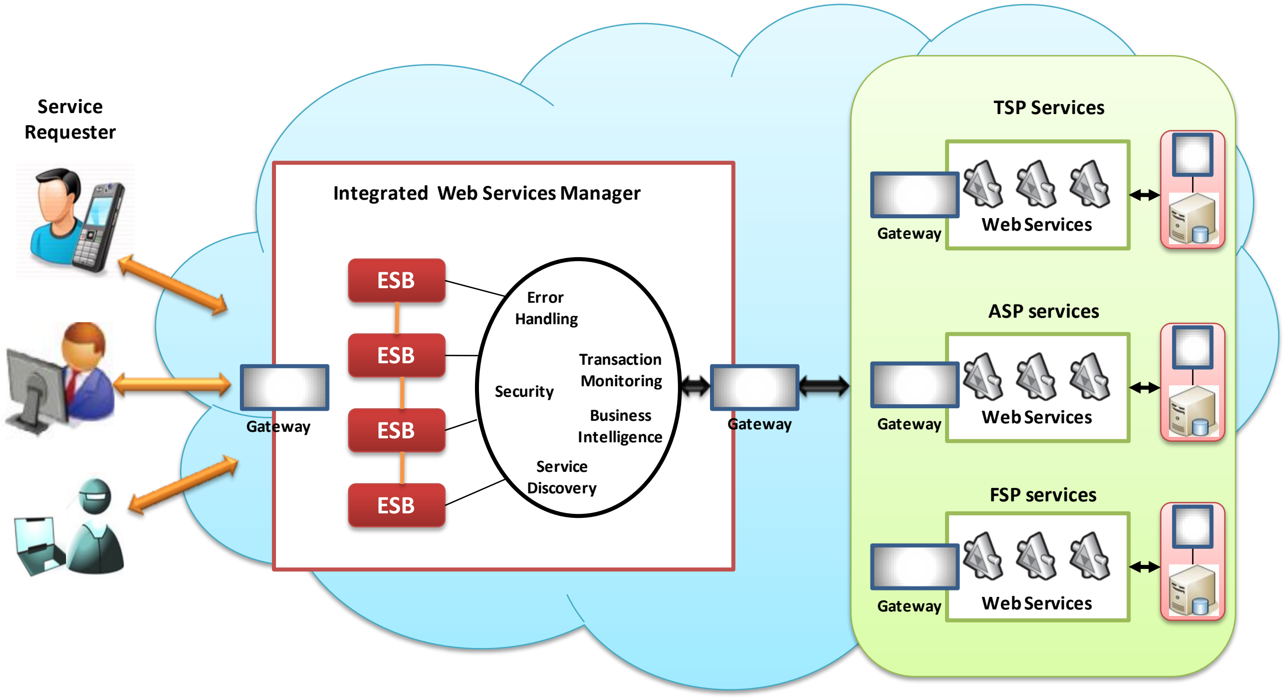

Fig.4: Integrated Financial Mobile Web Services Using Federated ESB

The federated-ESB pattern overcomes the fundamental limitations of the centralized-ESB model by partitioning the infrastructure into separate ESB that can be scaled and configured separately. For instance, in new model (Fig. 3), it has a specific ESB infrastructure to host the B2B interfaces, while another ESB is in charge of the financial-transaction processing. This approach also centralizes certain capabilities such as security that do not impose any scalability limitation on the SOA infrastructure. Fig. 4 represents for integrated financial mobile web services using federated ESB pattern. This model represents a lighter, more flexible approach to both design and runtime management of services.

4. MULTI-LEVEL SECURITY FOR THE PROPOSED MODEL

The most important issue related to the financial Mobile services is security. Nowadays, several security mechanisms are in use offering financial mobile transactions. However, the current security solutions adopted in mobile environments at the communication layer are not adequate. Security at the application layer must be added in order to achieve end-to-end protection for wireless and mobile environments. The security challenges in mobile financial services are related to but not limited to the mobile devices, the radio interface, the network operator infrastructure and the type of transaction. Therefore, the development of multi-level security for integrated financial mobile services has become a major research topic in the field of commerce and trading for the research community and telecommunication industry.

The basic security services for mobile applications environments are authentication, confidentiality, data integrity and non-repudiation. The existing security systems have adopted symmetric and asymmetric cryptographic operations to achieve high level security. Although the mobile devices need more power consumption to perform public key operations, the protocol designers in recent years are providing lightweight security mechanisms using Public Key Infrastructure (PKI). The PKI is a system of digital certificates, certification authorities, and other registration authorities that provides solutions to enable secure mobile web services.

There are number of issues limiting the security in wireless and mobile environments. The major problems in successful mobile web services are lack of adequate security. The encryption techniques such as secret-key and public key infrastructure are the most common methods to ensure transaction privacy, confidentiality and integrity. However, these techniques are depending upon the security of the endpoint systems in terms of protecting the keys from modification or misuse. The public key encryption techniques are the standard mechanism to adopt security like network security, operating system security, application data security and Digital Rights Management.

Fig.5 represents the required level of security while integrating financial mobile web services using federated ESB architectural pattern. This pattern enables the developers to design lighter, interoperable, and scalable SOAs that can enable true business agility in large enterprise scenarios. The level 1 is required secure customized and dynamic user interface to support financial mobile web services. At level 2, it is demanded for the high level security to services rendered by TSP, ASP, and FSP. Also, it is most important to secure the service registry which is placed at IWSM as level 3. In level 4, the secure communication is needed among tele-service providers together with their services. In addition, at level 3 and 4, secure data transmission is entailed with application service providers and with financial service providers respectively.

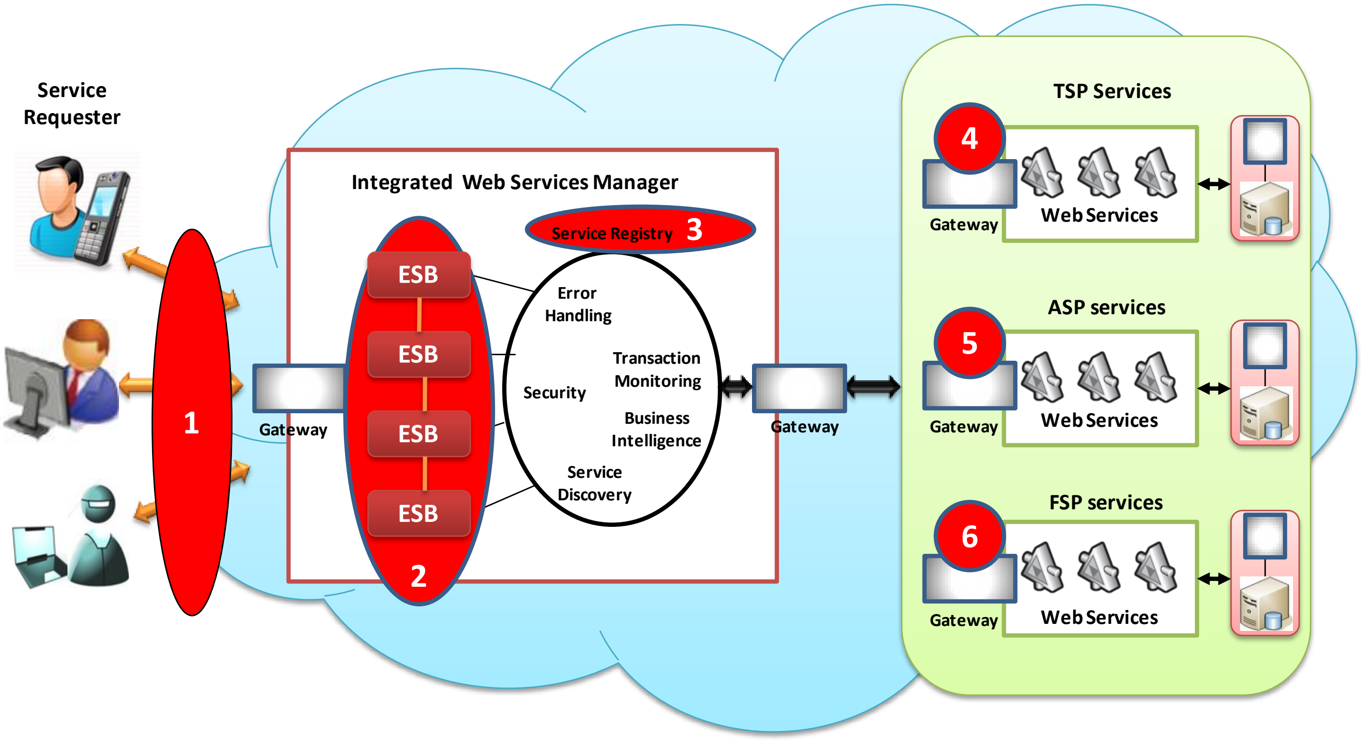

Fig.5: Multi-level Security for the Proposed Model

5. CONCLUSION

SOA has to evolve toward a more agile design, development, and deployment process. Most of all, however, SOA must close the gap between IT and business. While integrating the financial Mobile Web services, the traditional central enterprise-service-bus (ESB) approach and service-orchestration platforms poses serious limitations and the federated ESB architectural pattern conquers those limitations by partitioning the infrastructure into separate ESBs that can be scaled and configured separately.

As the technology changes the adaptability of mobile devices, network services and service providers are major bottlenecks for the common man. Hence, it is vital to design an integrated architecture for mobile applications especially for financial services which can be adaptable by any device, any network and any service etc. This paper proposes an architecture that provides multi-level Security for integrated financial mobile web services using federated ESBs. The proposed architecture enables service integration, network integration and supports access to financial services from any mobile devices. As the future research, the architecture is premeditated to implement with Microsoft .NET language and intended to measure the performance together with security deliberations.

6. REFERENCES

[1] Bieberstein, N., Bose, S., Fiammante, M., Jones, K., and Shah, R. 2006. Service-Oriented Architecture (SOA) Compass Business Value, Planning, and Enterprise Roadmap. Pearson Education. pp.2.

[2] Schmidt, M.T., Hutchison, B., Lambros, P., and Phippen, R. 2005. The Enterprise Service Bus: Making Service-oriented Architecture Real. IBM System Journal. Volume 44, No.4, 781-797.

[3] Sam Chung, Davalos, S., Niiyama, C., Daehee Won, Seung-Ho Baeg, and Sangdeok Park. 2009. A UML model-driven business process development methodology for a Virtual Enterprise using SOA & ESB. In IEEE Proceedings of the Asia-Pacific Services Computing Conference (APSCC). 246-253.

[4] Jesus Rodriguez and Don Demsak. 2009. Lightweight SOAs:Exploring Patterns and Principles of a New Generation of SOA Solutions. The Architecture Journal, Volume, 22, 32-38.

[5] Hérault, C., Thomas, G. and Lalanda, P. 2005. Mediation and enterprise service bus: A position paper. In Proceedings of the First International Workshop on Mediation in Semantic Web Services (MEDIATE). 1-14.

[6] Kristijan Dragicevic, Luis Garcés-Erice, Daniel Bauer. 2010. DISCE: A Declarative Inter-ESB Service-Connectivity Configuration Engine. In IEEE Proceedings of International Conference on Web Services. 489-496.

[7] Wen Zhu and Walt Melo. 2009. Refactoring J2EE Application for JBI-Based ESB: A Case Study. In IEEE Proceedings of the International Enterprise Distributed Object Computing Conference. 213-217.

[8] Yan Liu. 2009. A Process Modeling-Based Approach for Web Service Management. In IEEE Proceedings of the International Conference on Web Services. 928-935.

[9] Joris, C., Valentin, D., De Cock, D., Preneel, B. and Vandewalle, J. 2002. On the Security of Today's Online Electronic Banking Systems. Elseiver Advanced Technology. Computers and Security. Volume 21, No.3. 257-269.

[10] Jung, B., Han, I. and Lee, S. 2001. Security Threats to Internet: a Korean multi-industry investigation. Information and Management. Elsevier Science Publishers. Volume 38, Issue 8. 487-498.

[11] Ezedin, S. B., Mohamed, E. E. and Hayawi, K. 2006. End-To-End Security Solutions for WLAN: A Performance Analysis for the Underlying Encryption Algorithms in the Lightweight Devices. In Proceedings of IWCMC, Canada, ACM Publications. 1295-1299.

[12] Mehul Shah. 2008. Web services Security in SUN™ Java Composite Application Platform Suite (Java CAPS) 6. White Paper. Sun Microsystems Inc. 1-17.

[13] Mirtalebi, A., and Khayyambashi, M.R. 2011. Enhancing security of Web service against WSDL threats. In 2nd IEEE Proceedings of the International Conference on Emergency Management and Management Sciences (ICEMMS). 920-923.

[14] Xiaohong Li and Ke He. A Unified Threat Model for Assessing Threat in Web Applications. In IEEE Proceedings of the International Conference of Information Security and Assurance (ISA). 142-145.

[15] Rahaman, M.A. and Schaad, A. 2007. SOAP-based Secure Conversation and Collaboration. In IEEE Proceedings of the International Conference on Web Services. 471-480.

[16] Nordbotten, N.A. 2009. XML and Web Services Security Standards. IEEE Communications. Surveys and Tutorials. Volume 11, No.3. 4-21.

[17] Duan, Y., Bao, Y., Pan, L., Yan, B., Xun, J. and Shi, N. 2008. A Secure Web Services Model Based on the Combination of SOAP Registration Info and Token Proxy. In IEEE Proceedings of the International Symposium on Computer Science and Computational Technology (ISCSCT), 15 -20.