ABSTRACT

Web services are the new breed of web applications; they are self-contained, self-describing, modular applications that can be published, located, and invoked across the World Wide Web. The use of web services on the Web or Internet is expanding rapidly as the need for application-to-application communication and interoperability grows. Web service based applications can vary in complexity from simple operations, such as checking a bank account balance online, to complex application like customer relationship management, or enterpise resource planning systems.

In the basic web service models, the interactions are between three roles: the service provider, service registry, and service requestor. The main operations of web services are to publish, find and bind. The service provider is the owner of the service and is responsible for the development and deployment of the web services. The provider publishes the deployed services in the service registries. The service requester is the user of the web services, who searches the service registries to find web services that match his desired functionality. The service requester uses the service description available in the registry to bind and execute the web service.

A number of standards have been developed for web services. The functionality of the web services is defined, using the Web Services Description Language (WSDL), which follows the syntax and semantics of the Extensible Mark-up Language (XML). The service specifications are maintained in the service registry known as Universal Description Discovery
and Integration (UDDI) which is an XML-based registry. The interactions in web services are carried out using the standard messaging protocol, called Simple Object Access Protocol (SOAP).

Web services provide many technological and business benefits which include application and data integration, versatility, code reusability and cost savings. For business organizations to make optimal use of web services, various associated services for web service technology such as billing, accounting, security, reliability etc. have to be made available. Flexible billing systems provide a competitive advantage to the service provider and the accounting functionalities need to be adapted to different customer needs. There is a need for accounting solutions that need to be integrated into the web service application systems.

Authentication, Authorization, and Accounting (AAA) refers to the security architecture for distributed systems that is used for controlling access to services, and tracking the resource usage. Authentication involves validating the end user identity prior to permitting that user to access the network. Authorization, defines what rights and services the end user is allowed, once the network access is granted. Accounting is carried out by logging in the session. The statistical information about the usage of the service, such as identity of the user and the nature and quantum of the service delivered, is also provided.

An integrated accounting architecture has been proposed to automate the accounting functions of the web services. The proposed architecture serves to distribute the collection of web service usage and provide for metering, charging, billing, and payment, accounting and auditing. The proposed architecture comprises of an AAAAC
(Authentication, Authorization, Accounting, Auditing and Charging) server and AAAAC clients that would be located at the service providers. Metering of service usage would be carried out by the clients, while the server would be responsible for the management and overall co-ordination of the accounting activities and maintenance of all the accounting records.

Charging and Accounting policies have been proposed to support the variable pricing schemes of service providers. Policies facilitate expressing the change in service behavior, and can be managed independent of the business functions. Accounting of the usage of time, content and functions, is proposed. The service providers would be able to not only cover the cost but also earn revenue from the services consumed. The service users would have to pay only for the services used by them. WS-Policy standard is proposed to define the charging and accounting policies. Standard XML query processors can be used for querying the policy details by service users.

WS-RADIUS protocol has been developed to perform the metering function for web services. This protocol is based on the AAA RADIUS protocol. The protocol components have been developed to be located at the AAAAC server and AAAAC clients. The metering module is designed to record the metered data as Usage Data Records (UDR). The usage data is maintained for every service and for every user. The Internet Protocol Detail Record (IPDR) standard has been proposed to maintain the metered, billed and accounting data. The structure of IPDR allows one or many usage records which are service specific to be maintained in the XML format.

Billing is the process of using the metering information to generate bills on users for their service consumption. It is proposed to maintain
information regarding the billing mode, period and other user information required for raising bills on the service users. Bills would be automatically generated based on the charging and accounting policies selected by the users. Different types of billing can be supported as per service provider/user preferences. For instance, billing could be for a single usage or consolidated monthly, bi-monthly, quarterly, half yearly or yearly.

The accounting module is proposed to store and aggregate the information of resource usage. The module is designed to maintain the bills, update the payments and match the bills with payments to determine the payment outstanding from the users. The Audit module provides an interface for the accounting records to enable auditing, which is the process of verification of the correctness of procedures regarding the service usage. The users, service provider and third party auditors can verify the use of resources and their charging.

The proposed accounting architecture and design has been implemented and tested using a sample E-book service. The web services were developed using SOAP and REpresentational State Transfer (REST). The web application has been designed with multiple web services with provision for E-book search, on-line reading and book download. Different pricing schemes, such as pay-per-use, content based, time based and function based fee structure, have been defined. The RADIUS protocol (RFC 2865) was used as the base to develop the WS-RADIUS metering protocol.

The contribution of the research work is discussed in detail. A critical analysis of the research work and future directions for research have also been outlined.