Decentralized and Diverse Access Control Architecture for Online Purchases

Rajender Nath
Department of Computer Science & Applications
Kurukshetra University, Kurukshetra
Haryana, India.

Gulshan Ahuja
Tilak Raj Chadha Institute of Management & Technology
Yamunanagar
Haryana, India.

ABSTRACT
Open and distributed nature of Internet assists users to use online services for the benefits of costs, time and efficiency. To avail theses services users are required to submit their credentials for the purpose of registration and further verification. The credentials supplied by a user may not be sufficient enough to grant the access to the requested service and a further verification may need to be carried by demanding some confidential and secret credentials from the user. Much has been talked about federated identity management, which makes possible to utilize the existing Identity management systems for realizing authentication and authorization decisions. In a federated system, identity provider plays an important role and issues the certified credentials which can be utilized at the service provider's end. The scalability of such system is limited due to the need of identity provider to act as a central authority and maintain credentials of ever growing large number of requesters. As more and more portals are offering online services, there is a strong need to provide authentication and authorization independent of any central authority. This paper proposes a new architecture which eliminates the role of centralized authority for managing and issuing users' credentials. The proposed architecture allows keeping the right of disclosure of attributes under the sole control of user and also ensures that the user is not able to modify the confidential credentials which have been registered and verified by various trusted authorities. Decentralized diverse attributes based verification architecture can be used as an enabling technology for supporting web based operations.

Keywords
Identity management, authorization, authentication, centralized authority.

1. INTRODUCTION
With the increase in number of portals offering online services like sale-purchase of items, storage services etc., there is an increasing trend towards adoption and use of these services among users. There is an ongoing growth of online portals for e.g. Letsbuy.com, PNet.in, indiaplaza.com which offer sale of products using cash on delivery model. For online shopping, users are given options for making payments via multiple options like credit cards, debit cards, internet banking as well as cash on delivery approach. In cash on delivery model, users are required to register with the service portal by providing basic pieces of information, thereafter the user selects the products he wants to purchase and adds these products to the shopping cart. At the end of shopping, user checks out and selects cash on delivery option for payment. The selling portal assumes that the information provided by the user at registration time is correct and valid. The selling portal processes the request to deliver the products at the specified address. In a defaulting cased a user may submit wrong information like contact address, contact number etc. and this will result in a loss to the selling portal in terms of cost, time and wastage of resources. To avoid this, the selling portal may opt to verify the user's submitted credentials by requesting for certain other attributes few of which may contain confidential and secure information like driving license number, PAN card number, passport number etc. However user may wish to disclose only basic set of credentials in the form of attributes and may decide to refrain from disclosing the confidential and sensitive attributes to service portals for the concerns of safety and privacy. This creates a requirement for trusted agencies which can maintain private and confidential information of users and allow this information to be used by service portals without losing privacy and security of user specific information. This is well known fact that users generally have one or more registrations at various service portals which may contain set or subset of confidential and sensitive attributes and users have a significant degree of trust with these service portals which are generally govt. or semi government registered organizations. There has to be a mechanism whereby privacy and confidentiality of users' attributes is maintained and on the same time the service portal must be able to authenticate users' attributes and provide access to process purchase request after ascertaining that supplied attributes are correct and genuine. As users' attributes are already available with trusted agencies like income tax department, public service department etc. Therefore, the credentials of a user can be verified through involvement of these agencies and thereafter the request can be allowed for processing. The rest of this paper is structured as follows. Section 2 highlights the related work. Section 3 describes the details about architecture and explains its working. Finally Section 4 concludes and briefly describes scope for the future work.

2. RELATED WORK
The traditional mechanism for authenticating a requester is to utilize user's login name and password which is generally stored at the service provider's end. This method of authentication is not suitable for an open distributed domain. The method of authorization is by utilizing client X.509 attributes certificates [1]. Attributes provide information about a requester and can be

coupled with identity of the requester to make authorization decisions. To provide security of the attributes during communication, these are digitally signed and presented as attribute assertions. These assertions associate an issuer, its holder, validity period and other possible conditions with an attribute. A number of research papers based on attributes authorization have been proposed by researchers. Ioannis Mavridis et al. [2] proposed a mechanism for access control based on attribute certificates for medical Internet applications. David Chadwick [3] proposed X.509 privilege management infrastructure. Later David Chadwick et al. [4] proposed Role-Based Access Control with X.509 Attribute Certificates. The proposed approach in paper adopted the standard X.509 PMI to build an efficient role-based trust management system in which role assignments can be widely distributed among organizations, and an XML-based local policy determines which roles to trust and which privileges to grant. Jordi Forne et al. [5] presented an implementation of an authorization system for web based applications based on the ITU-T X509 recommendations, which specifies use of privilege management infrastructure for realizing access control. Shibboleth architecture [6] defined a common XML framework for exchanging authentication, authorization and attribute assertions between entities. The Shibboleth architecture requires federated identity management. Every identity provider (IDP) can define an attribute release policy for each service provider (SP) within the federation. The IDP decides about which attributes can be released to particular SP based on the access control policy. The service requester can't specify about attributes that can be released to a SP. As the number of service requesters increases, there is an increase in the number of federations. This increases the complexity because of each SP has to manage its linkages across multiple federations. Another architecture that has been designed from user's perspective is the Microsoft's CardSpace [7]. A client application named as card selector is used for managing user's identity cards. The card selector allows service requesters to decide about the identity they want to disclose to the particular SP. CardSpace is built to allow exchange of information and attributes using open standards like SAML and OpenId. David W Chadwick [8] presented a model and protocol elements for linking AAs, service providers and user attributes together, under the sole control of the user and allowed merging the attributes from multiple AAs in order to grant the user access to its resources. Frikken K et al. [9] proposed an approach for attribute based access control with hidden policies and hidden credentials. Shen Hai Bo et al. [10] proposed an attribute based access control model for web services. Nirmal Dagdee et al. [11] proposed an access control methodology for sharing of open and Domain confined data using Standard Credentials. The methodology required that various types of standard credentials and related attributes are to be identified and published by some apex authority so that the resource providers can define their access policies in terms of these standard credentials. In real terms, identification of standard credentials is a very difficult task and is not suitable for largely distributed systems having millions of service requesters. Regina N. Hebig et al. [12] described a prototype implementation with an architecture based on the standards XACML, SAML, WSPolicy, WS-SecurityPolicy and WS-Trust, which put the focus on sharing identity and attribute information across independent domains for the purpose of access control. None of the above approaches have considered authorization in an open domain where the use of centralized authority can be eliminated and all entities dynamically collaborate to make decisions about providing access to on line services. A recent framework ADITI[13] for user centric identity federation enhanced the standard federated model with new IdP and SP components operated directly by users. In that approach all the attributes of the user are still kept with IdP and the user has to download all attributes from IdP to the card selector in order to utilize these attributes for authorization decisions. This provides users with full control over their attributes which can be changed at the will of the user. Therefore ADITI framework is not well suited for service portals where users' attributes are required to be verified without control of users over their own attributes and independent of any centralized authority. Rajender Nath et al. [14] presented a fine grained access control model based on diverse attributes. The model required a service provider to make contact with diverse attribute authorities and a redirection of access requests. The working remains dependent on federated access and certificate based management. This paper presents a decentralized diverse attributes based access control architecture that supports users' attributes verification and authorization for supporting on line purchases.

3. THE PROPOSED DECENTRALIZED & DIVERSE ACCESS CONTROL (DDAC) ARCHITECTURE

Figure 1 depicts the detailed view of the overall architecture.The DDAC architecture allows safe integration of users and service providers for attributes disclosure and verification without involvement of a centralized attribute authority. The DDAC architecture mainly comprises of following

A requester side module called as user attribute management (UAM) module that implements the 3 basic operations - attribute creation, attribute-authority mapping, attribute storage. A client side view is shown as figure 2. The UAM module allows users to maintain the set of basic attributes for the purpose of presenting to service portals for availing any service. The user can use UAM module to store attributes registered already with a number of trusted attributes providers (TAPs). Theses attributes are maintained as a list of records in the form of pair combination <TAP, Encrypt{ATS}> where TAP is the trusted attribute provider and ATS is the set of attributes which are encrypted using secret key of concerned TAP. Table 1. shows the set of records maintained at user's end.

A centralized unified resource description search engine (RDSE) which provides both information collection and information retrieval functions for diverse attribute authorities.

A controller module which is located on the service provider's end. The controller module receives the user's request and invokes credibility verification module for verification.

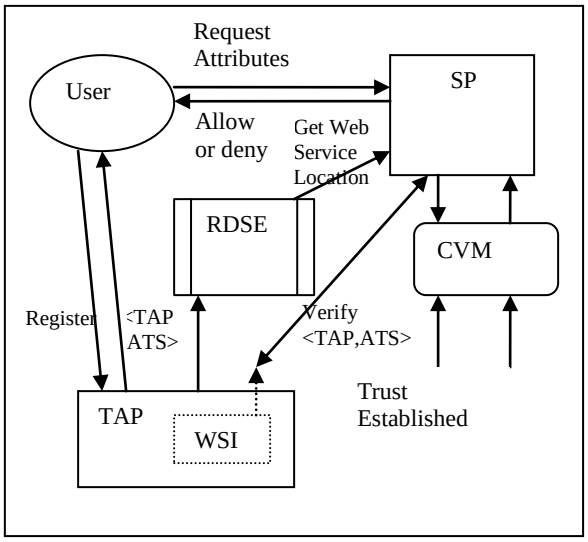

Figure 1: Working of DDAC Architecture

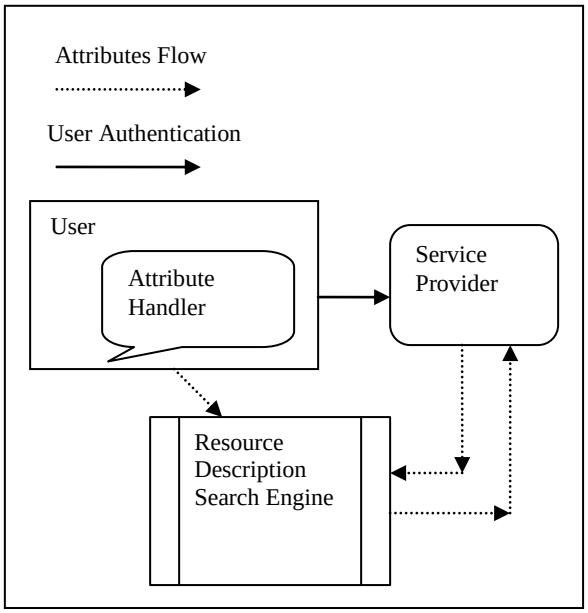

Figure 2: User Attribute Management Module

Table 1. Attributes-Set Records

Trusted Attributes Provider	Attribute Set
TAP1	EKTAP1{ATS1}
TAP2	EKTAP2{ATS2}
TAP3	EKTAP3{ATS3}

A credibility verification module (CVM) verifies the attributes against a data registry and computes credibility profile level (CPL) value for users. Based on the computed CPL value, users' requests may be serviced or denied. Figure 3. depicts the detailed view at service provider's end.

A web service interface, which is to be located at every service provider's end, who wish to provide service for verifying users' attributes. These service providers called as TAPs provide means for communication with the web service for verification of user's attributes.

When a user registers him self at service provider's end by providing set of attributes, controller module on the service provider's end evaluates the supplied credentials against its policy store to check whether its policies permit the request to be serviced without further verification or not. In case policies do not allow access based on presented credentials, controller module invokes credibility verification module. CVM checks against users' registration list, to verify whether the requesting user is already registered or not. If the same user have had already registered, his CPL value is accessed from CPL data store to find out whether request can be serviced or not. In case if user had not already registered or his CPL is found to be less than allowable limits, further verification is carried. All trusted attribute providing agencies who wish to participate for verification register their information in RDSE for search purpose. For e.g three TAPs called as TAP1, TAP2, TAP3 upload the descriptive information about a user notated as follows

(<GUID>, <TAP1>, value< WebServiceID1, WebServiceID2>),
(<GUID>, <TAP2>, value<WebServiceID1, WebServiceID2, WebServiceID3>),
(<GUID>, <TAP3>, value<WebServiceID1, WebServiceID2, WebServiceID3, WebServiceID4>).
The process of verification initiated by SP is carried as per below mentioned steps

a) SP submits the search query to RDSE in the form as (GUID, value<TAP1>).

b) RDSE searches the database to locate the concerned TAP and associated Web Service identifier and sends the information to SP.

c) SP sends the encrypted set of attributes (ATS) to concerned TAP by making a call to the web service for which ID has been located through search in RDSE.

d) Because ATS is submitted in an encrypted manner and therefore can only be decrypted by a TAP using its corresponding decryption key.

Every time a user is serviced, his CPL is computed and is stored in the CPL data store.

The CPL value is computed based on three main parameters 1) Acceptance Rating (AR); 2) Payment Delivery Rating (PDR); 3) Attribute Set Rating (ATSR).

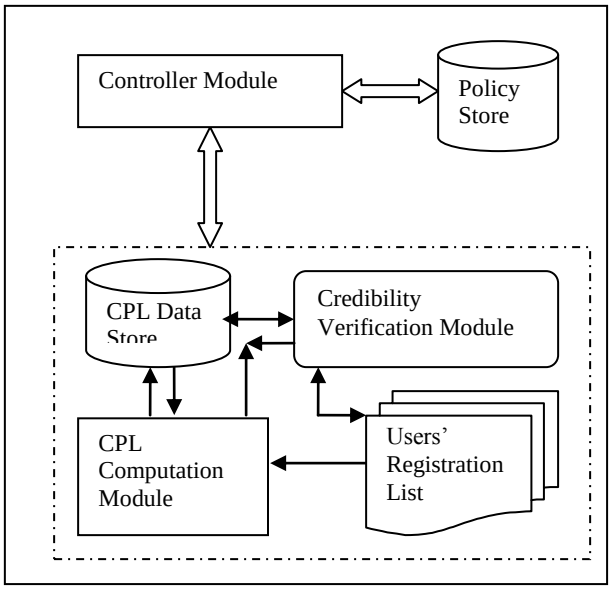

Figure 3: Details of Modules at Service Providers' end

The values for AR, PDR and CPL is computed as

$$AR = QA / QS$$

Where QA is the number of times requested items have been accepted by the user on delivery and QS is the total number of items supplied to the same user.

$$PDR = (N_{PT} + N_{PD}(1 - k * (TD / TG))) / QO$$

Where N_{PT} is the number of times payment made in time, N_{PD} is the number of times payment delayed, TD is the time delay in payment, TG is the time allowed for payment, QO is the number of items ordered and k is set as a rating constant. The CPL value for a user can be computed as

$$CPL = AR * K_1 + PDR * K_2 + ATSR * K_3$$

Where K_1, K_2 and K_3 are weight factors and are set by system administrator at service provider's end. These weights determine which parameter is more important than other parameter and completely dominate access and verification patterns.

If the computed CPL value for the requesting user is within allowable limits, the user's request is successfully accepted and requested items are allowed for delivery at specified location. CPL value of a user keeps on changing as per the past experiences of transactions with users. For new users, the CPL value is totally dependent on set of attributes because in that case AR and PDR values are nil. So for the first time requests, SP may impose more stringent requirements by demanding verification for confidential and sensitive attributes. For subsequent requests, CPL value keeps on changing and SP may choose to assign less weight for attributes set and may relax the further verification. Figure 4. depicts the control flow for handling any service request in the case where access is to be allowed on the basis of CPL.

The architecture proposed in our paper allows for verification of user's credentials and grant of service permission. There is no use of any centralized agency to maintain and provide all user credentials. The proposed system adds an overhead in terms of time required to service any request. This additional time can be measured as per following

Suppose there are N numbers of users and a maximum of M milliseconds are required to serve each user, assuming a constant bandwidth and interruption free connection. Total time (T_1) required to serve N users, in case there is no verification carried, can be computed as

$$T_1 = N * M$$

In the 2nd scenario, where verification is carried based on CPL value, the time required to serve N users can be computed as follows:

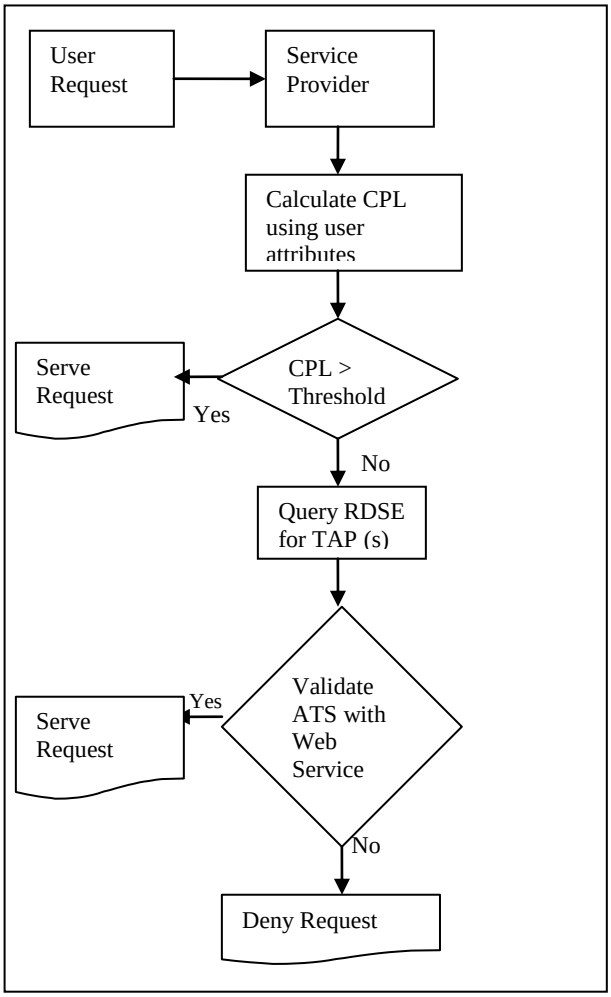

Figure 4: Control Flow for service request

Let TA be the additional time in milliseconds required to verify each request and P be the number of requests for which verification is required. So the total time of servicing requests in this case T_2 can be computed as

$$T_2 = (N - P) * M + P*(M + TA)$$

Where N is total number of requests and M is the maximum time in milliseconds required to serve a user's request without performing any verification. The processing time overhead (PTO) for servicing users' requests, as introduced by additional verification steps in our architecture, can be computed as

$$PTO = T_2 - T_1$$

However, the saving to organization in terms of costs, resource usage and hassle free delivery clearly outscores the extra time required for verification. The method employed in the architecture significantly reduces the processing time overhead based on the credibility profile values of users.

In this paper, we proposed a decentralized and diverse attributes based architecture where role of federated management can be completely eliminated. The proposed architecture also provides a means where time of verification reduces significantly based on the user's credibility and previous experiences. We discussed the issues involved with the use and distribution of users attributes. Federation based systems require a central monitoring authority for storing and utilizing users' attributes. The attributes remain under the sole control of users. The service providers can safely and securely verify users' attributes from trusted attribute authorities. In case of change in the values of the attributes, there is no need for intimation to any other party and trusted attribute partners can easily implement the change at their end. The trusted partners only provide the location and signature of web service in resource descriptive search engine. The information about signature of web service in resource descriptive search engine remains unchanged and do not effect any operation even when there is a change in the value of one or more user's attributes.

4. CONCLUSION

As reported in the literature, so far, no efforts have been made in the existing architectures to consider the use and verification of diverse attributes for supporting online purchases. Most of the existing architectures are based on a centralized authority for distribution of client certificates and user attributes management. This limits the scalability and manageability of such systems in terms of distribution, storage and revocation of user attributes. By analysing the existing architectures for access control, based on verification of attributes, it has been found that there is a strong requirement for a decentralized and diverse attribute based architecture for verification of users' requests. To address this issue, architecture has been proposed in this paper. The proposed architecture has outscored over the existing architectures in terms of decentralized attribute management. It provides complete control over user attributes and requires no revocation in case of any change in values of attributes, thus providing hassle free services to the users. As the proposed architecture introduces a number of additional verification steps, to be performed for granting access to a service request, this leads to additional processing time for servicing users' requests.

5. REFERENCES

[1] S. Farrell, An Internet Attribute Certificate Profile for Authorization, http://www.ietf.org/rfc/rfc3281.txt

[2] Ioannis Mavridis, Christos Georgiadis, George Pangalos, marie Khair, "Access Control based on Attribute certificates for Medical Internet applications", Journal of medical Internet Research, Vol 3, 2001.

[3] David Chadwick, "The X.509 Privilege Management Infrastructure", http://sec.cs.kent.ac.uk/download/X509pmiNATO.pdf, 2002

[4] David W. Chadwick, Alexander Otenko, and Edward Ball," Role-Based Access Control With X.509 Attribute Certificates", IEEE internet computing, march-april 2003, pp. 62 – 69.

[5] J. F d an d M. F. Hanarejos , "Web-based Authorization based on X.509 Privilege Management Infrastructure ",IEEE Pacific Rim Conference on Communications, Computers and signal Processing, 2003.

[6] S. Cantor. "Shibboleth Architecture, Protocols and Profiles", Working Draft 02. 22 September 2004, http://shibboleth.internet2.edu/

[7] D. Chappell, "Introducing Windows CardSpace", Microsoft MSDN website, 2006, http://msdn.microsoft.com/enus/library/aa480189.aspx.

[8] David W Chadwick," Authorisation using Attributes from Multiple Authorities", Proceedings of the 15th IEEE International Workshops on Enabling Technologies Infrastructure for Collaborative Enterprises 2006.

[9] Frikken K, Atallah M, Jiangtao Li, "Attribute-Based Access Control with Hidden Policies and Hidden Credentials", IEEE Transactions on Computers, Volume 55, Issue 10, Page(s): 1259 – 1270, Oct. 2006.

[10] Shen Hai Bo, Hong Fan, "An attribute based access control model for web services", Proceeding of the 7th International Conference on Parallel and Distributed Computing, Applications and Technologies, IEEE 2006

[11] Nirmal Dagdee, Ruchi Vijaywargiya," Access control methodology for sharing of open and Domain confined data using Standard Credentials", International Journal on Computer Science and Engineering Vol.1(3), 2009, 148-155.

[12] Regina N. Hebig et al., "A Web Service Architecture for Decentralized Identity- and Attribute-based Access Control", IEEE International Conference on Web Services, 2009

[13] Michal Prochazka et al., "User Centric Authentication for Web Applications", IEEE, 2010, 67-74.

[14] Rajender Nath, Gulshan Ahuja, "A Fine Grained AccessControl Model Based on Diverse Attributes", Global Journal of Computer Science & Technology, Volume 11 Issue 15 Version 1.0, August-September 2011 USA.