PROMINENT AND SECURED RESOURCE SHARING IN CLOUD ENVIRONMENT WITH SEMI-TRUSTED THIRD PARTY

Mr. Mareboina Kiran Kumar [1], **Mr. Shaik Anjaneyulu Babu**[2*]

1 Final Year MCA Student, QIS College of Engineering and Technology, Ongole

2*Associate Professor, MCA Dept., QIS College of Engineering and Technology, Ongole

Abstract: *Sharing of resources on the cloud can be accomplished on a substantial scale since it is financially savvy and area free. Regardless of the promotion encompassing cloud computing, associations are as yet hesitant to send their organizations in the cloud computing condition because of worries in secure asset sharing. In this paper, we propose a cloud asset intervention administration offered by cloud specialist organizations, which assumes the job of confided in outsider among its distinctive inhabitants. This paper formally determines the resource sharing component between two unique inhabitants within the sight of our proposed cloud resource mediation service. The rightness of resource mediation service among various inhabitants utilizing four particular calculations (Activation, Delegation, Forward Revocation and Backward Revocation) is likewise exhibited utilizing formal confirmation. The execution investigation recommends that sharing of resources can be performed safely and effectively crosswise over various occupants of the cloud.*

Keywords: Cloud Storage, Resource Sharing.

I. INTRODUCTION

There are a number of benefits afforded by the use of cloud computing to facilitate collaboration between users and organizations, security and privacy of cloud services and the user data may deter some users and organizations from using cloud services (on a larger scale) and remain topics of interest to researchers [11], [8], [12], [14]. Typically, a cloud service provider (CSP) provides a web interface where a cloud user can manage resources and settings (e.g. allowing a particular service and/or data to selected users). A CSP then implements these access control features on consumer data and other related resources. However, traditional access control models, such as role based access control [15], are generally unable to adequately deal with cross-tenant resource access requests. In particular, cross-tenant access requests pose three key challenges. Firstly, each tenant must have some prior understanding and knowledge about the external users who will access the resources.

Thus, an administrator of each tenant must have a list of users to whom the access will be allowed. This process is static in nature. In other words, tenants cannot leave and join cloud as they wish, which is a typical setting for a real world deployment. Secondly, each tenant must be allowed to define cross-tenant access for other tenants as and when needed. Finally, as each tenant has its own administration, trust management issue among tenants can be challenging to address, particularly for hundreds or thousands of tenants. To provide a secure cross-tenant resource access service, a fine-grained cross-tenant access control model is required [21].

Thus, in this paper, we propose a cloud resource mediation service (CRMS) to be offered by a CSP, since the CSP plays a pivotal role managing different tenants and a cloud user entrusts the data to the CSP. We posit that a CRMS can provide the CSP competitive advantage, since the CSP can provide users with secure access control services in a cross tenant access environment (hereafter, we referred to as cross tenant access control - CTAC).

II RELATED WORK

DASCE: Data Security for Cloud Environment with Semi-Trusted Third Party

Off-site data storage is an application of cloud that relieves the customers from focusing on data storage system. However, outsourcing data to a third-party administrative control entails serious security concerns. Data leakage may occur due to attacks by other users and machines in the cloud. Wholesale of data by cloud service provider is yet another problem that is faced in the cloud environment. Consequently, high-level of security measures is required. In this paper, we propose Data Security for Cloud Environment with Semi-Trusted Third Party (DaSCE), a data security system that provides (a) key management (b) access control, and (c) file assured deletion. The DaSCE utilizes Shamir's (k, n) threshold scheme to manage the keys, where k out of n shares are required to generate the key. We use multiple key managers, each hosting one share of key. Multiple key managers avoid single point of failure for the cryptographic keys. We (a) implement a working prototype of DaSCE and evaluate its performance based on the time consumed during various operations, (b) formally model and analyze the working of DaSCE using High Level Petri nets (HLPN), and (c) verify the working of DaSCE using Satisfiability Modulo Theories Library (SMT-Lib) and Z3 solver. The results reveal that DaSCE can be effectively used for security of outsourced data by employing key management, access control, and file assured deletion.

Control Cloud Data Access Privilege and Anonymity with Fully Anonymous Attribute-Based Encryption.

Cloud computing is a revolutionary computing paradigm which enables flexible, on-demand and low-cost usage of computing resources, but the data is outsourced to some cloud servers, and various privacy concerns emerge from it. Various schemes based on the Attribute-Based Encryption have been proposed to secure the cloud storage. However, most work focuses on the data contents privacy and the access control, while less attention is paid to the privilege control and the identity privacy. In this paper, we present a semi-anonymous privilege control scheme AnonyControl to address not only the data privacy but also the user identity privacy in existing access control schemes. AnonyControl decentralizes the central authority to limit the identity leakage and thus achieves semi-anonymity. Besides, it also generalizes the file access control to the privilege control, by which privileges of all operations on the cloud data can be managed in a fine-grained manner. Subsequently, we present the AnonyControlF which fully prevents the identity leakage and achieve the full anonymity. Our security analysis shows that both AnonyControl and AnonyControl-F are secure under the DBDH assumption, and our performance evaluation exhibits the feasibility of our schemes.

Fine-Grained Two-Factor Access Control for Web-Based Cloud Computing Services

In this paper, we introduce a new fine-grained two-factor authentication (2FA) access control system for web-based cloud computing services. Specifically, in our proposed 2FA access control system, an attribute-based access control mechanism is implemented with the necessity of both a user secret key and a lightweight security device. As a user cannot access the system if they do not hold both, the mechanism can enhance the security of the system, especially in those scenarios where many users share the same computer for web-based cloud services. In addition, attribute-based control in the system also enables the cloud server to restrict the access to those users with the same set of attributes while preserving user privacy, i.e., the cloud server only knows that the user fulfills the required predicate, but has no idea on the exact identity of the user. Finally, we also carry out a simulation to demonstrate the practicability of our proposed 2FA system.

Role based access control (RBAC) enables fine-grained access control (and generally in a single domain). Different extensions of RBAC have been proposed in the literature to support multi-domain access control. These approaches rely on a single body responsible for maintaining cross-domain policies. However, in a cloud environment, each user (individual or organization) may have one or more tenants and have a separate management infrastructure. Therefore, it is likely that users are not able to agree on a single organization to manage access control on their behalf. With the increased trend of cloud services due to its various benefits (e.g. on-demand self-service model and resources sharing among tenants), it is essential for CSPs to provide mechanisms to segregate the data of the tenants.

An advanced Hierarchical Open Stack Access Control model was proposed in [22], which is designed to facilitate secure and effective management of information sharing in a community cloud for both routine and cyber incident response needs. A cross-tenant trust model and its RBAC extension was proposed in [20] for enabling secure cross-tenant communication. A multi-tenant authorization as a Service (MTAaaS) platform to enforce such cross-tenant trust model is also presented in the paper. In a separate work, an autonomous multi-tenant network security framework "Jobber" was proposed [18]. However, the security of the approaches in these three studies was not demonstrated.

As computing resources are being shared between tenants and used in an on-demand manner, both known and zero-day system security vulnerabilities could be exploited by the attackers (e.g. using side-channel and timing attacks) [1]. In [16], a fine grained data-level access control model (FDACM) designed to provide role-based and data-based access con-trol for multi-tenant applications was presented. Relatively lightweight expressions were used to represent complex policy rules. Again, the security of the approach was not provided. Zhao et al. [23] propose a cross-domain single sign on authentication protocol for cloud users, whose security was also proven mathematically.

In the approach, the CSP is responsible for verifying the user's identity and making access control decisions. Specification level security is difficult to achieve at the user and provider ends.

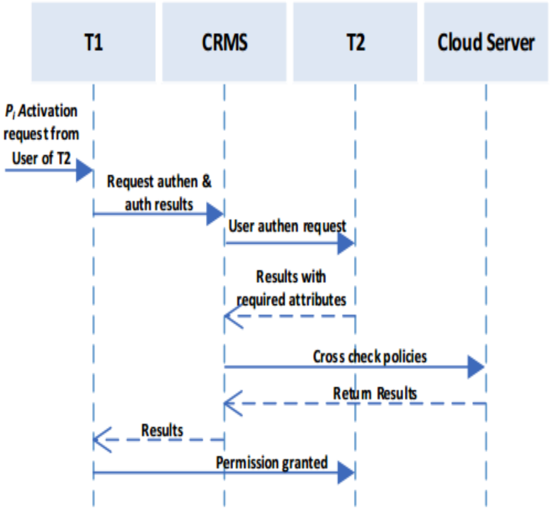

T1
CRMS
T2
Cloud Server
P_i Activation request from User of T2
Request authen & auth results
User authen request
Results with required attributes
Cross check policies
Return Results
Results
Permission granted

Fig.1: Sequence diagram for permission request in the cloud

III PROPOSED SYSTEM

Cloud Resource Mediation Service (CRMS)– A Trusted Third Party For Enabling Cross Tenant Resource Access:

We describe our proposed CRMS designed to facilitate the CSPs in managing cross-tenant esource access requests for cloud users. To explain the service, we use an example ivolving two tenants, *T*1 and *T*2, where *T*1 is the Service Provider (SP) and *T*2 is the Service Requester (SR) (i.e. user). *T*1 must own some permission *pi* for which user of *T*2 can generate a cross-tenant request. The resource request from a user of *T*2 must be submitted to *T*1, which then handovers the request to the CRMS for authentication and authorization decisions. The CRMS evaluates the request based on the security polices provided by *T*1. The steps are represented in Figure 1.

Steps for permission activation request in the cloud

There are three main entities, namely: the SP (*T*1), the SR (*T*2), and the CRMS. The roles of these entities are described as follows:

Tenant T1 responsibilities: *T*1 is responsible for publishing cross tenant policies on the CRMS. *T*1 receives access requests from *T*2 and redirects the request to the CRMS for further processing.

Tenant T2 responsibilities: The CRMS redirects ac-cess requests to *T*2 for authentication. Once the redirected access request is received, the responsibility of *T*2 is to authenticate the identity of particular user. In response, *T*2 sends the user authentication response (valid or invalid) and tenant authentication response to the CRMS.

CRMS responsibilities: The CRMS receives the permission-activation request redirected from *T*1. Once an access request is received, the CRMS evaluates the request on the pre-published policies and responds to *T*1.

The steps for initiating a permission-activation request are as follows:

Step 1: Permission activation request: A user wishing to access a resource at *T*1. The user will be presented a directory where a list of shared services along with their descriptions is present.

Step 2: Request redirection to the CRMS: Upon selection of a shared service the user wishes to access, the user is redirected to the CRMS site. On the site, the user will be asked for the parent tenant. The user selects the parent tenant and the CRMS redirects the user's request to the selected tenant (*T*2 in this case).

*Step 3: Tenant T*2 *authentication:* The user has to authenticate at her parent tenant, *T*2. Upon successful authentication, the user will be redirected again to CRMS with the attributes requested by the CRMS for cross tenant policy execution.

*Step 4: CRMS redirection to tenant T*1 *& permission activation:* The user's attributes are evaluated against the *T*1 policy and if the policy criteria is successfully fulfilled, then the user is provided service access at *T*1; otherwise, the access request is denied. The CRMS also takes into account any conflict of interest policies, such as Chinese Wall Policy

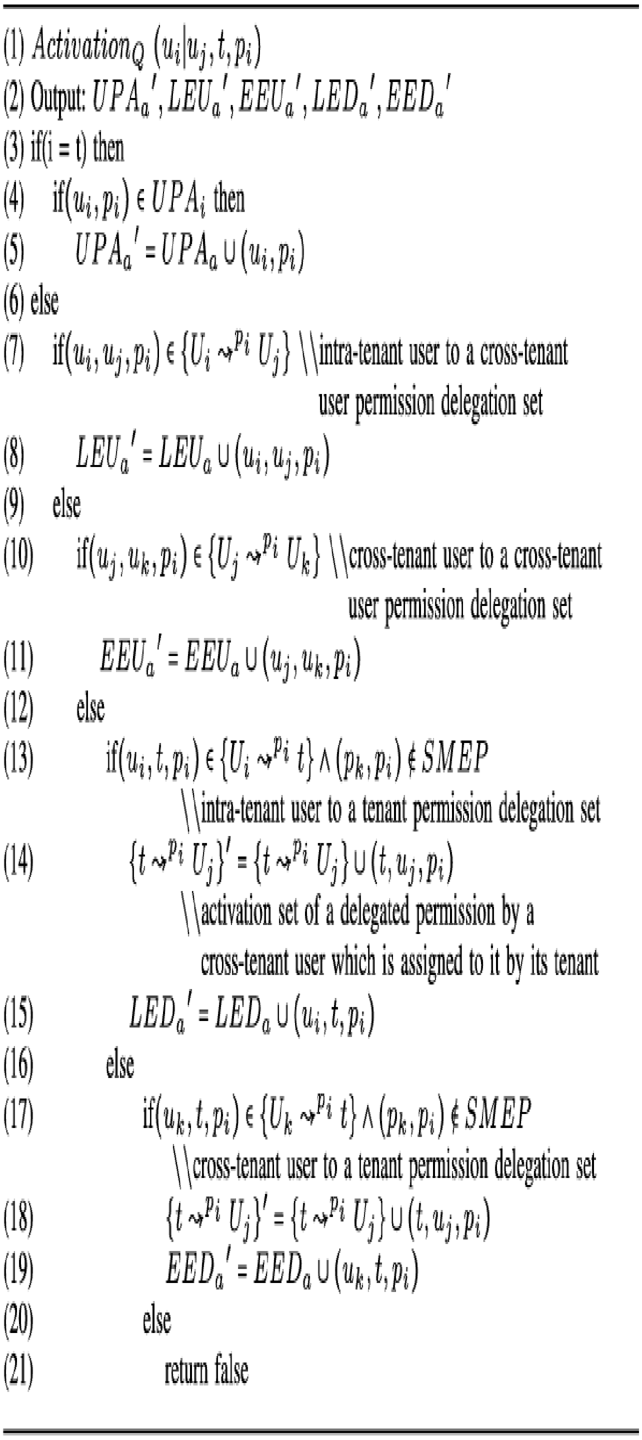
IV METHODOLOGY

Activation Algorithm

The activation algorithm is based on the activation query defined in Section IV. It authenticates a user for the activation of a particular permission. As defined earlier, a permission activation request can be generated by an intra-tenant/cross-tenant user. For a cross-tenant user, a prior delegation of permission to cross-tenant user/tenant must exist according to Definition 2. The algorithm for activation of the permission under the proposed CRMS for CSPs is shown below.

Using HLPN, we model the Activation Algorithm. The HLPN in Figure 3 illustrates the process of activating a permission *pi* in a cross-tenant environment.

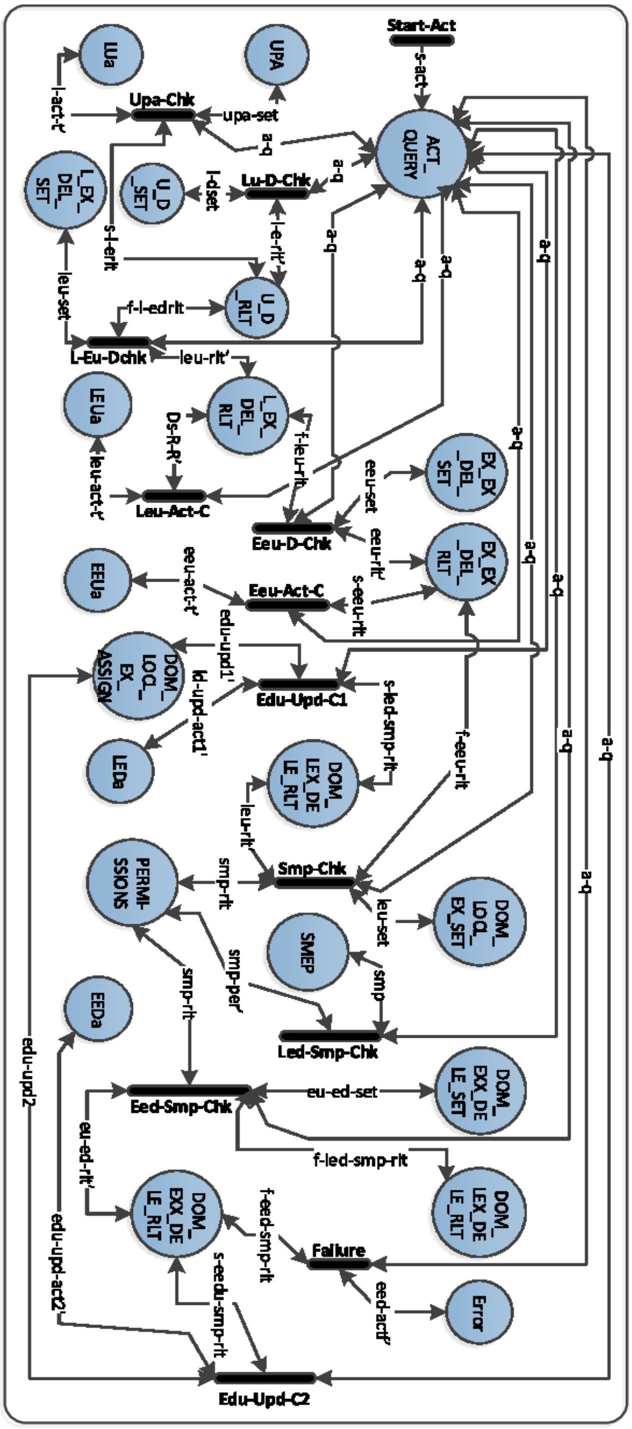
There are four different ways in which the permission *pi* can be activated in a multi-tenant environment. When the condition for any of the cases is satisfied, subsequent permission *pi* for the corresponding user will be activated. If none of the cases are satisfied, the algorithm terminates.



```
$Activation_Q\ (u_i|u_j, t, p_i)$
Output: $UPA_a', LEU_a', EEU_a', LED_a', EED_a'$
if(i = t) then
  if$(u_i, p_i) \in UPA_i$ then
    $UPA_a' = UPA_a \cup (u_i, p_i)$
else
  if$(u_i, u_j, p_i) \in \{U_i \rightsquigarrow^{p_i} U_j\}$ \\intra-tenant user to a cross-tenant
                                   user permission delegation set
    $LEU_a' = LEU_a \cup (u_i, u_j, p_i)$
  else
    if$(u_j, u_k, p_i) \in \{U_j \rightsquigarrow^{p_i} U_k\}$ \\cross-tenant user to a cross-tenant
                                   user permission delegation set
      $EEU_a' = EEU_a \cup (u_j, u_k, p_i)$
    else
      if$(u_i, t, p_i) \in \{U_i \rightsquigarrow^{p_i} t\} \wedge (p_k, p_i) \notin SMEP$
          \\intra-tenant user to a tenant permission delegation set
        $\{t \rightsquigarrow^{p_i} U_j\}' = \{t \rightsquigarrow^{p_i} U_j\} \cup (t, u_j, p_i)$
          \\activation set of a delegated permission by a
            cross-tenant user which is assigned to it by its tenant
        $LED_a' = LED_a \cup (u_i, t, p_i)$
      else
        if$(u_k, t, p_i) \in \{U_k \rightsquigarrow^{p_i} t\} \wedge (p_k, p_i) \notin SMEP$
            \\cross-tenant user to a tenant permission delegation set
          $\{t \rightsquigarrow^{p_i} U_j\}' = \{t \rightsquigarrow^{p_i} U_j\} \cup (t, u_j, p_i)$
          $EED_a' = EED_a \cup (u_k, t, p_i)$
        else
          return false
```

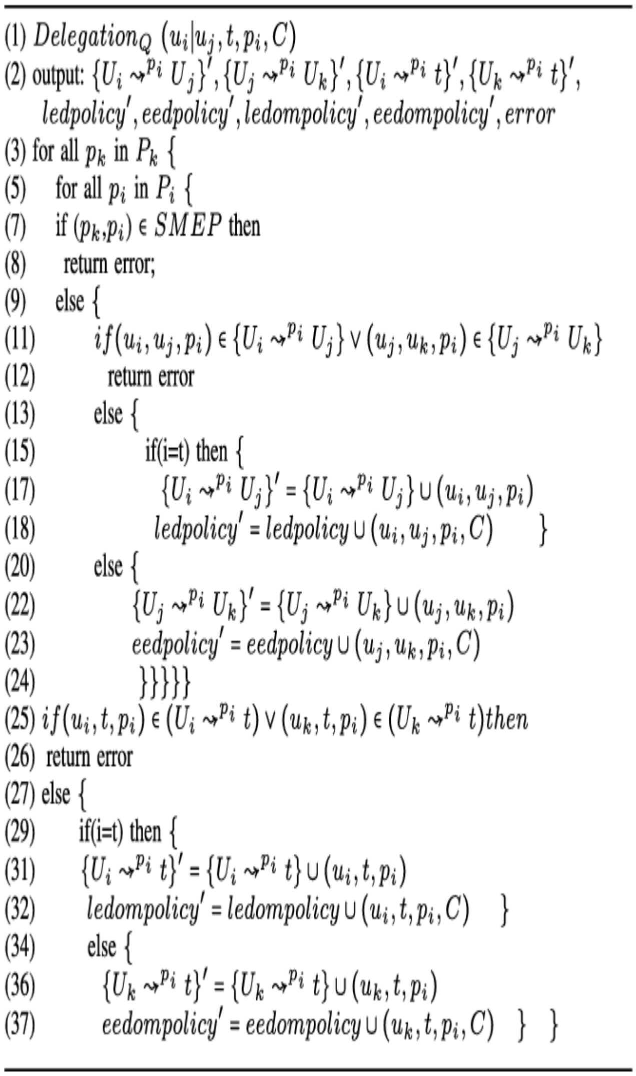
Fig. 2 Activation Algorithm

A HLPN Model for the process of activating a permission *pi* in a cross-tenant environment.



Delegation Algorithm

The delegation algorithm is based on the delegation query defined in Section IV. The delegation algorithm enables an intra-tenant user to generate a delegation request for the permission which the user can activate. The algorithm for delegation of the permission under the proposed CRMS for CSPs is shown in Figure



```
$Delegation_Q\ (u_i|u_j, t, p_i, C)$
output: $\{U_i \rightsquigarrow^{p_i} U_j\}', \{U_j \rightsquigarrow^{p_i} U_k\}', \{U_i \rightsquigarrow^{p_i} t\}', \{U_k \rightsquigarrow^{p_i} t\}',$
    $ledpolicy', eedpolicy', ledompolicy', eedompolicy', error$
for all $p_k$ in $P_k$ {
    for all $p_i$ in $P_i$ {
    if $(p_k, p_i) \in SMEP$ then
     return error;
    else {
        $if (u_i, u_j, p_i) \in \{U_i \rightsquigarrow^{p_i} U_j\} \vee (u_j, u_k, p_i) \in \{U_j \rightsquigarrow^{p_i} U_k\}$
          return error
        else {
            if(i=t) then {
              $\{U_i \rightsquigarrow^{p_i} U_j\}' = \{U_i \rightsquigarrow^{p_i} U_j\} \cup (u_i, u_j, p_i)$
             $ledpolicy' = ledpolicy \cup (u_i, u_j, p_i, C)$      }
        else {
            $\{U_j \rightsquigarrow^{p_i} U_k\}' = \{U_j \rightsquigarrow^{p_i} U_k\} \cup (u_j, u_k, p_i)$
            $eedpolicy' = eedpolicy \cup (u_j, u_k, p_i, C)$
             }}}}}
$if (u_i, t, p_i) \in (U_i \rightsquigarrow^{p_i} t) \vee (u_k, t, p_i) \in (U_k \rightsquigarrow^{p_i} t) then$
 return error
else {
    if(i=t) then {
    $\{U_i \rightsquigarrow^{p_i} t\}' = \{U_i \rightsquigarrow^{p_i} t\} \cup (u_i, t, p_i)$
     $ledompolicy' = ledompolicy \cup (u_i, t, p_i, C)$   }
     else {
       $\{U_k \rightsquigarrow^{p_i} t\}' = \{U_k \rightsquigarrow^{p_i} t\} \cup (u_k, t, p_i)$
       $eedompolicy' = eedompolicy \cup (u_k, t, p_i, C)$   }   }
```

Fig: Delegation algorithm

Using HLPN, we model the Delegation Algorithm. The HLPN in Figure 5 illustrates the process of delegating a permission *pi* in a cross-tenant environment.

DASCE: Data Security for Cloud Environment with Semi-Trusted Third Party possess the following system components

System Components:

- System Framework
- The Responsibilities of Entities
- Steps Involved for Initiating a Permission Activation Request
- Revocation

System Framework:

In this paper formally specifies the resource sharing mechanism between two different tenants in the presence of our proposed cloud resource mediation service.There is three main entities. To explain the service, we usean example involving two tenants, T1 and T2, where T1 isthe Service Provider (SP) and T2 is the Service Requester(SR) (i.e. user) and the CRMS. T1 must own some permission pi for whichuser of T2 can generate a cross-tenant request. The resourcerequest from a user of T2 must be submitted to T1, whichthen handovers the request to the CRMS for authenticationand authorization decisions. The CRMS evaluates the requestbased on the security polices provided by T1.

The Responsibilities of Entities:

a) Tenant T1 responsibilities: T1 is responsible for publishing cross tenant policies on the CRMS. T1 receives access requests from T2 and redirects the request to the CRMS for further processing.

b) Tenant T2 responsibilities: The CRMS redirects access requests to T2 for authentication. Once the redirected access request is received, the responsibility of T2 is to authenticate the identity of particular user. In response, T2 sends the user authentication response (valid or invalid) and tenant authentication response to the CRMS.

c) CRMS responsibilities: The CRMS receives the permission-activation request redirected from T1. Once an access request is received, the CRMS evaluates the request on the pre-published policies and responds to T1.

Steps Involved For Initiating a Permission Activation Request:

Step 1: Permission activation request: A user wishing to access a resource at T1. The user will be presented a directory where a list of shared services along with their descriptions is present.

Step 2: Request redirection to the CRMS: Upon selectionof a shared service the user wishes to access, the user is redirected to the CRMS site. On the site, the user will be asked for the parent tenant. The user selects the parent tenant and the CRMS redirects the user’s request to the selected tenant (T2 in this case).

Step 3: Tenant T2 authentication: The user has to authenticate at her parent tenant, T2. Upon successful authentication,the user will be redirected again to CRMS with the attributes requested by the CRMS for cross tenant policy execution.

Step 4: CRMS redirection to tenant T1 & permission activation: The user's attributes are evaluated against the T1 policy and if the policy criteria are successfully fulfilled, then the user is provided service access at T1; otherwise, the access request is denied. The CRMS also takes into account any conflict of interest policies, such as Chinese Wall Policy.

Revocation:

There are two ways in which we can revoke a previously granted permission from the cross-tenant user/cross-tenant.To achieve the permission revocation, we introduce the Forward Revocation Algorithm and the Backward RevocationAlgorithm.A forward revocation query defines a request in which an intra-tenant userrevoke a permission or a set of permissions from a crosstenant user/tenant along with the deactivation of the delegationpolicy. And A Backwardrevocation query defines a action that is triggered when theattributes of the delegatee mismatch. Thus, an intra-tenant userrevokes a permission or a set of permissions from a crosstenant user/tenant as well as deactivating the delegation policy.

V. RESULTS

We tend to remodel every formula with their properties to the SMT problem solver, so we tend to might verify them. Z3 problem solver considers each the model and its characteristics and checks the properties of the model fulfill the specified level of satisfaction. Within the execution of the CTAC model with the expressed characteristics within the Z3 problem solver, the model of CTAC performed well and made the results of course. F QF AUFLIA logic used for closed quantifier-free linear formulas over the idea of number arrays extended with free sort and performance symbols was conjointly utilized in implementation under SMT-LIB.

VI. CONCLUSION

In this paper, we proposed a cross-tenant cloud resource mediation service (CRMS), which can act as a trusted-third party for fine-grained access control in a cross-tenant environment. For example, users who belong to an intra-tenant cloud can allow other cross-tenant users to activate permission in their tenant via the CRMS.

Future work will propose a formal model CTAC with four algorithms designed to handle the requests for permission activation. We then modeled the algorithms using HLPN, formally analyzed these algorithms in Z language, and verified them using Z3 Theorem Proving Solver. The results obtained after executing the solver demonstrated that the asserted algorithm specific access control properties were satisfied and allows secure execution of permission activation on the cloud via the CRMS. Also include a comparative analysis of the proposed CTAC model with other state-of-the-art cross domain access control protocols using real-world evaluations. For example, one could implement the protocols in a closed or small scale environment, such as a department within a university. This would allow the researchers to evaluate the performance, and potentially (in)security, of the various approaches under different real-world settings.

VII. REFERENCES

[1] Akhunzada, A., Gani, A., Anuar, N. B., Abdelaziz, A., Khan, M. K., Hayat, A., & Khan, S. U. (2016). Secure and dependable software defined networks. Journal of Network and Computer Applications, 61, 199-221.

[2] Alam, Q., Tabbasum, S., Malik, S., Alam, M., Tanveer, T., Akhunzada, A., Khan, S., Vasilakos, A. and Buyya, R., (2016). Formal Verification of the xDAuth Protocol. IEEE Transactions on Information Forensics and Security, 11(9), pp. 1956-1969.

[3] Ali, M., Malik, S. and Khan, S., DaSCE: Data Security for Cloud Environment with Semi-Trusted Third Party.

[4] Barrett, C., Conway, C.L., Deters, M., Hadarean, L., Jovanovi, D., King, T., Reynolds, A. and Tinelli, C., 2011, July. Cvc4. In International Conference on Computer Aided Verification (pp. 171-177). Springer Berlin Heidelberg.

[5] Bofill, M., Nieuwenhuis, R., Oliveras, A., Rodrguez-Carbonell, E. andRubio, A., 2008, July. The barcelogic SMT solver. In International Conference on Computer Aided Verification (pp. 294-298). Springer Berlin Heidelberg.

[6] Bruttomesso, R., Cimatti, A., Franzn, A., Griggio, A. and Sebastiani, R.,2008, July. The mathsat 4 smt solver. In International Conference on Computer Aided Verification (pp. 299-303). Springer Berlin Heidelberg.

[7] Choo, K.K., 2006. Refuting security proofs for tripartite key exchange with model checker in planning problem setting. In 19th IEEE Computer Security Foundations Workshop (CSFW'06) (pp. 12-pp).IEEE.

[8] Choo, K.-K. R., Domingo-Ferrer, J. and Zhang, L., 2016. Cloud Cryptography: Theory, Practice and Future Research Directions. Future Generation Computer Systems, 62, pp. 51-53.

[9] De Moura, L. and Bjørner, N., 2011. Satisfiability modulo theories: introduction and applications. Communications of the ACM, 54(9), pp.69- 77.

[10] Dutertre, B. and De Moura, L., 2006. The yices smt solver.Tool paperat http://yices.csl. sri. com/tool-paper. pdf, 2(2).

[11] Heiser, J., 2009. What you need to know about cloud computing security and compliance. Gartner, Research, ID, (G00168345).

[12] Jung, T., Li, X. Y., Wan, Z. and Wan, M., 2015. Control Cloud Data Access Privilege and Anonymity WithFully Anonymous Attribute-Based Encryption. IEEE Transactions on Information Forensics and Security, 10(1), (pp. 190-199).

[13] Lin, Y., Malik, S.U., Bilal, K., Yang, Q., Wang, Y. and Khan, S.U., 2016. Designing and Modeling of Covert Channels in Operating Systems.IEEE Transactions on Computers, 65(6), pp.1706-1719.

[14] Liu, J. K., Au, M. H., Huang, X., Lu, R., and Li, J., 2016. Fine-Grained Two-Factor Access Control for Web-Based Cloud Computing Services.IEEE Transactions on Information Forensics and Security, 11(3), (pp. 484-497).

[15] Liu, X., Deng, R. H., Choo, K.-K. R. and Weng, J., 2016. An EfficientPrivacy-Preserving Outsourced Calculation Toolkit With Multiple Keys. IEEE Transactions on Information Forensics and Security, 11(11), pp. 2401-2414.

[16] Ma, K., Zhang, W. and Tang, Z., 2014. Toward Fine-grained Data-levelAccess Control Model for Multi-tenant Applications.International Journalof Grid and Distributed Computing, 7(2), pp.79-88.

[17] Murata, T., 1989. Petri nets: Properties, analysis and applications. Proceedings of the IEEE, 77(4), pp.541-580.

[18] Sayler, A., Keller, E. and Grunwald, D., 2013. Jobber: Automating inter-tenant trust in the cloud. In Presented as part of the 5th USENIX Workshop on Hot Topics in Cloud Computing.

[19] SMT-Lib. http://smtlib.cs.uiowa.edu/, 2015.

[20] Tang, B. and Sandhu, R., 2013, August. Cross-tenant trust models in cloud computing. In Information Reuse and Integration (IRI), 2013 IEEE 14th International Conference on (pp. 129-136). IEEE.

[21] Yang, Y., Zhu, H., Lu, H., Weng, J., Zhang, Y. and Choo, K.-K. R., 2016. Cloud based data sharing with fine-grained proxy re-encryption. Pervasive and Mobile Computing, 28, pp. 122-134.

[22] Zhang, Y., Patwa, F., Sandhu, R. and Tang, B., 2015, August. Hierarchical secure

Authors Profile

Mr. **Mareboina Kiran Kumar** pursuing MCA 3rd year in Qis College and Engineering and Technology in Department of Master of Computer Applications, Ongole.

Mr. **Shaik Anjaneyulu Babu** is currently working as an Associate Professor in Department of Master of Computer Applications in QIS College of Engineering & Technology. He is pursuing Ph.D in the domain of data mining.