Community-Based Secure Information and Resource Sharing in Azure Cloud IaaS

Yun Zhang
Institute for Cyber Security
Univ of Texas at San Antonio
San Antonio, TX 78249
amy.u.zhang@gmail.com

Farhan Patwa
Institute for Cyber Security
Univ of Texas at San Antonio
San Antonio, TX 78249
Farhan.Patwa@utsa.edu

Ravi Sandhu
Institute for Cyber Security
Univ of Texas at San Antonio
San Antonio, TX 78249
Ravi.Sandhu@utsa.edu

ABSTRACT

To efficiently collaborate in cyber security defense and response, organizations must be able to securely share information and resources. A community in a cloud IaaS, which refers to a group of organizations with common business interests, will utilize cloud IaaS to realize their infrastructure deployments. Communities establish a mechanism to prevent, detect and respond to cyber attacks, and help member organizations in the community recover expeditiously. In this paper, we present an access control model for secure information and resource sharing between organizations in a community-based isolated environment in Microsoft Azure [2] IaaS cloud platform, one of dominant commercial cloud platforms. The model facilitates organizations to share their IT resources with each other in a controlled and secure manner. We formally specify the administrative model and discuss enforcement techniques in the Azure cloud platform.

CCS Concepts

•**Security and privacy** → **Formal security models;**

Keywords

Formal models; IaaS; Azure; Information Sharing

1. INTRODUCTION

As cyber incidents become more prevalent, secure sharing of cyber incident information is becoming a critical part of cyber incident prevention and response. With cyber attacks becoming more sophisticated, individual organizations are finding it harder to defend themselves. Thus developing a well defined cyber incident response mechanism is an essential part of any cyber defense program. Such a mechanism would help to streamline cyber incident responses, quickly identify potential threats and worst case scenarios, and assist in expediting post cyber incident investigations and responses. Cloud computing technology gives the opportunity to have multiple organizations in a single cloud infrastructure, thus sharing the same underlying security and privacy concerns. While cloud computing technology significantly improves efficiency and flexibility of business systems, it also facilitates cyber collaborations. Current cloud computing setups do not leverage this collaborative aspect and organizations still rely on traditional means of sharing and exchange of cyber related information. In our opinion, there is a lack of a broadly accepted cyber response mechanism through which organizations in a cloud computing setup can securely and reliably share and exchange cyber related information. In our work we seek to develop such a mechanism which allows multiple organizations to actively collaborate in a well defined community in a secure and reliable manner.

In this paper, we will investigate models for secure information and resource sharing in Microsoft Azure [2] public cloud. A public cloud is one of the most prevalent deployment models of current cloud platforms, providing services for public users and organizations. Azure aims to allow rapid deployment of infrastructure and services to meet all needs for businesses. We propose the use of a public cloud model to form a community of organizations that want to collaborate with other community members as part of their cyber defense strategy. The aim here is not just a static information sharing mechanism, but a comprehensive collaborative setup which leverages the full functional stack of the cloud platform with security at the very core of this model. The focus is on security, efficient infrastructure sharing and a clear standardized model for cross-organizational communication.

The community comprises of a cyber security committee and a cyber security forum. The cyber security committee provides the member organizations a platform to easily communicate and coordinate on cyber security and privacy issues and is limited to select individuals from each member organization. The cyber security forum provides a generic mechanism for general users to share security related information across the community and is limited to members of the organizations who can voluntarily join and leave.

In the event of a cyber incident, the relevant organizations within the community can form a cyber incident response team expeditiously and include select external security specialists. This is enabled by a cyber security service in the public cloud, which enables organizations having cross-organization collaborations to communicate and coordinate with other organizations during life cycle of a cyber incident. This collaboration would not only include security information but also public cloud infrastructure that could be used to investigate and analyze this data.

SCC'16, May 30, 2016, Xi'an, China.

DOI: http://dx.doi.org/10.1145/2898445.2898455

In this paper, we present an access control model for cyber security information and resource sharing within a public cloud, for cyber incident prevention and response. This paper proceeds as follows. We present some related work and background knowledge in Section 2. We introduce Azure Access Control (Azure-AC) model in Section 3. In Section 4, we define the Azure-AC model with Secure Isolated Domain extension (Azure-AC-SID), which is our model for cyber collaboration in Azure. We discuss enforcement in Section 5. Finally we conclude our work in Section 6.

2. RELATED WORK

One purpose of cloud computing is for users to conveniently share data with other users, either from the same organization or different organizations. Every cloud product has its own access control policies to share data between users. In cloud SaaS, files like documents, photos and videos can be easily shared in cloud applications like Google Drive, Dropbox, etc. Some other cloud product like Sharepoint [5] and Salesforce [4] provide users platform to share structured data. IaaS cloud platforms like AWS [1], OpenStack [3] and Azure allows users to share computing infrastructure such as virtual machines, networking, storage, etc.

Cloud-based information sharing has been explored in the literature in several different ways. Tang and Sandhu [11] propose trust relationships established between tenants to facilitate sharing. Raykova et al [8] propose fine-grained cryptographic access control to protect not only confidentiality of stored data from unauthorized access and the storage provider, but also of users' access patterns. Thuraisingham et al [12] consider assured information sharing for coalition organizations to share information stored in multiple clouds via policy-based cloud-centric access control. Our model is also about information sharing between coalition of organizations, but not at the level of data access and storage though a cloud platform. It is more concerned about organizations' utilization of cloud platform to share information.

The general topic of information and resources sharing in traditional system has received considerable attention in the literature. Various access control and authorization solutions have been proposed in traditional distributed systems, such as secure virtual enclaves [10], models for coalition-based access control [6], and group-centric secure collaboration [7]. Our focus is on models for sharing information and resources in cloud systems. We have presented several such access control models for collaborative communities of organizations in cloud environment. We developed OpenStack Access Control model with SID extension (OSAC-SID) [13] and Hierarchical OpenStack Access Control model with SID extension (OSAC-HMT-SID) [15], which allow organizations sharing information in a controlled manner in OpenStack cloud platform [3]. We also designed AWS Access Control model with SID extension (AWSAC-SID) [14] in AWS public cloud. This paper is a continuation of this line of work, but in Azure public cloud.

The concept we used comes from Group-Centric Secure Information Sharing (g-SIS) [7], which introduces group-based information and resources sharing, allows sharing among a group of organizations. In this paper, we explore information and resources sharing in Azure public cloud. Azure integrates Active Directory to manage users, which gives a great flexibility and compatibility for organizations to move to cloud. Azure has fixed role set but sufficient for most organizations. Azure also has very powerful federation capabilities, just like AWS. Since our goal is to explore our models across all dominant cloud platforms, it is essential to build a suitable information and resources sharing model for Azure.

In the model we developed in this paper, we confine our attention to information and resource sharing among tenants within a single public cloud. Extending such sharing to multiple/hybrid clouds is an interesting research problem, but out of scope for this paper.

3. AZURE ACCESS CONTROL MODEL

In this section, we introduce Microsoft Azure cloud platform and present Azure Access Control model. Microsoft Azure is one of the dominant cloud IaaS platforms for enterprises. Similar to AWS, as an IaaS provider, Azure's core features include compute, storage, database and networking. Azure divides the basic features into four categories: build infrastructure, develop applications, gain insights from data, and manage identity and access.

In Azure, any user has the capability to create an Azure account. The user who creates an Azure account will be the owner and super administrative user of that account. Unlike AWS, local users created in an Azure Active Directory can create their own Azure account which is completely isolated from the parent account.

Azure has two main components to manage users' access to resources in the cloud: Azure Active Directory (AAD) and Subscriptions (Sub). In order for a user to use resources in Azure, the user has be assigned to a subscription. Azure Active Directory helps to manage users, including both local Azure AD users and other valid Microsoft users.

Azure offers a form of role-based access control, wherein permissions are defined over cloud resources within a role in resource groups. Roles can then be assigned to users. Roles are predefined in Azure.

Figure 1 depicts the Azure Access Control model. In this and other figures, the arrows denote binary relations with the single arrowhead indicating one side and double arrowheads many sides. Azure Access Control (Azure-AC) model has fourteen components, shown as labelled circles or ellipses in the figure. Each component is also referred to by its abbreviated name given in parenthesis below the full name. They are discussed below.

Accounts (A): To have its own public cloud resources, an organization need to open an Azure account. An Azure account allows an organization to own specific (virtual) cloud resources that can be accessed through Azure cloud services.

Azure Active Directory (AAD): Azure Active Directory (Azure AD) is Microsoft's multi-tenant cloud based directory and identity management service. It provides a full suite of identity management capabilities including multi-factor authentication, device registration, password management, privileged account management, role based access control, security monitoring and so on. Azure AD also provides single sign-on (SSO) access to cloud SaaS Applications. It can also integrate with other identity management solutions used in industry.

Subscriptions (Sub): Users have access to cloud resources via subscriptions. Subscriptions are the units of usage and billing for cloud resources. In order to have access to cloud resources, users must be assigned to at least one subscription.

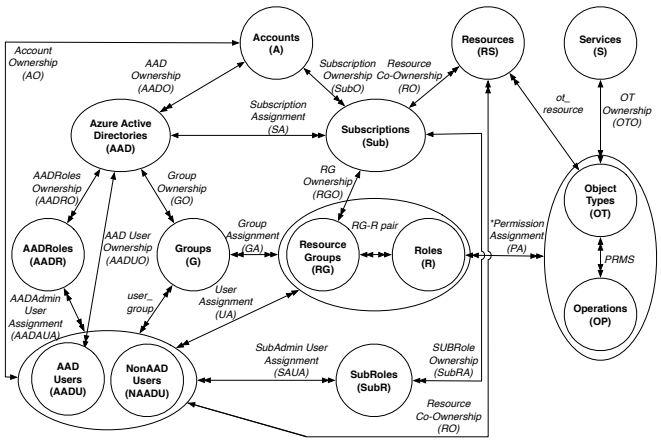

Figure 1: Azure Access Control (Azure-AC) Model

Azure Active Directory Roles (AADR): Azure AD Roles allow one to manage the directory and identity-related features. Azure AD has a set of administrative roles, including billing administrator, global administrator, password administrator, service administrator and user administrator. Each of these administrative roles is designed for a different specific administrative purpose. It also has a normal user role, which has no administrative power.

Subscription Roles (SubR): Subscription Roles are a separate role set from Azure Active Directory Roles. Subscription Roles are administrative roles which give users permissions to manage cloud resources via a subscription. SubR includes service administrator and co-administrators, both of which can give users access to cloud services. The services administrator and co-administrators can be either Microsoft accounts or Azure AD users. A service administrator cannot be a local Azure AD user from the same Azure AD assigned to that subscription.

Azure Active Directory Users (AADU) and Non-Azure Active Directory Users (NAADU): Users represent individuals who can be authenticated by Azure and authorized to access cloud resources through an Azure account. Users from both Microsoft accounts and partner organization accounts are allowed to access cloud resources in Azure. Azure Active Directory Users are users created in Azure AD. They can be administrative users of the directory or normal users. Non-Azure AD users refer to users not from the local Azure AD, but from partner organizations and other Microsoft users.

Groups (G): A group is simply a set of users and it can include both Azure AD users and Non-Azure AD users. Groups belong to an Azure AD account. The existence of groups is to conveniently mange multiple users as a single unit. Each policy attached to a group will apply to all group members.

Resource Groups (RG): Resource Groups is a logical resource container which allows customers to add various cloud resources like database, virtual machine etc. Resource Groups provides a way to monitor and control users' access to collections of cloud resources.

Roles (R): Users are assigned to a resource group with roles to get permissions to access to cloud resources. Roles allows users to have permissions to access cloud resources, for instance virtual machines (VMs), storage, networking and etc. Roles could be different collections of meta permissions like read and write toward a specific piece of resource. Roles are only able to be assigned to users inside a resource group.

Resources (RS): Resources refer to cloud assets which can be owned by users. Cloud assets are cloud resources such as virtual machines, databases, storages, etc. Since the only way for users to access resources is through subscriptions, we also define that the subscription has ownership over the resources.

Services (S): Services refer to cloud services Azure provides to its customers such as compute, storage, networking, administration, and database.

Object Types (OT) and Operations (OP): An Object Type represents a specific type of object. From the CSP's viewpoint, objects are more like services. We define object types as particular service types the cloud provides. For instance, for the compute service, the object type is a virtual machine; for the storage service, the object type is a storage container; etc.

With the concepts described above, we formalize Azure-AC model as follows.

Definition 1. Azure-AC model has the following functions and relations in addition to the components enumerated above.

- Account Ownership (AO) : is a function AO : A $\rightarrow$ U, mapping an account to its owning user.
- AAD Ownership (AADO) : is a function AADO : AAD $\rightarrow$ A, mapping an AAD to its owning account. Equivalently viewed as a many-to-one relation AADO $\subseteq$ AAD $\times$ A.
- Subscription Ownership (SubO) : is a function SubO : Sub $\rightarrow$ A, mapping a Sub to its owning account. Equivalently viewed as a many-to-one relation SubO $\subseteq$ Sub $\times$ A.
- Resource Group Ownership (RGO) : is a function RGO : RG $\rightarrow$ Sub, mapping a RG to its owning subscription. Equivalently viewed as a many-to-one relation GRO $\subseteq$ RG $\times$ Sub.
- AAD User Ownership (AADUO) : is a function AADUO : AADU $\rightarrow$ AAD, mapping a user to its owning Azure AD. Equivalently viewed as a many-to-one relation AADUO $\subseteq$ AADU $\times$ AAD.
- Group Ownership (GO) : is a function GO : G $\rightarrow$ AAD, mapping a group to its owning Azure AD. Equivalently viewed as a many-to-one relation GO $\subseteq$ G $\times$ AAD.
- Azure AD Roles Ownership (AADRO) : is a function AADRO : AADR $\rightarrow$ AAD, mapping a Azure AD role to its owning Azure AD. Equivalently viewed as a many-to-one relation AADRO $\subseteq$ U $\times$ A.
- Resource Co-Ownership (RSO) : is a function RSO : RS $\rightarrow$ Sub $\vee$ RS $\rightarrow$ (AADU $\vee$ NAAUD), mapping a piece of resource to its owning subscription and user. Equivalently viewed as a many-to-one relation RSO $\subseteq$ RS $\times$ Sub $\vee$ RS $\times$ (AADU $\vee$ NAAUD).
- Object Type Owner (OTO) : is a function OTO : OT $\rightarrow$ S, mapping an object type to its owning service. Equivalently viewed as a many-to-one relation OTO $\subseteq$ OT $\times$ S.
- Resource Group Role (RG-R) pair $\subseteq$ GR $\times$ R, is a many-to-many relation mapping resource groups to roles.
- Subscription Assignment (SubA) : is a many-to-one relation SubA $\subseteq$ Sub $\times$ AAD.

- Subscription Roles Assignment (SubRA) : is a many-to-many relation SubRA $\subseteq$ Sub $\times$ SubR.
- AADAdmin User Assignment (AADAUA) : is a many-to-many relation AADAUA $\subseteq$ (AADU $\cup$ NonAADU) $\times$ AADR, mapping a user to a Azure AD.
- SubAdmin User Assignment (SAUA) : is a many-to-many relation SAUA $\subseteq$ (AADU $\cup$ NonAADU) $\times$ SubR. There is one exception of subadmin user assignment relation in assigning a service admin to a subscription. Every subscription has only one service admin user assigned with it, while it can have up to 200 co-admin users assigned with it.
- User Assignment (UA) : is a many-to-many relation UA $\subseteq$ (AADU $\cup$ NonAADU) $\times$ RG-R, mapping a user to a resource group role pair.
- Group Assignment (GA) : is a many-to-many relation GA $\subseteq$ G $\times$ RG.
- Permission Assignment (PA) : is a many-to-many relation PA $\subseteq$ (RG $\times$ R) $\times$ PREM, assigning resource group role pairs to permissions. We mention that Azure has fixed sets of collections of permissions which users can choose, instead of giving users the capability to define their own permission sets.
- user_group $\subseteq$ U $\times$ G, is a many-to-many relation assigning users to groups where the user and group must be owned by the same account.
- ot_resource $\subseteq$ OT $\times$ RS, is a one-to-many relation mapping resources to object types.
- PRMS = OT $\times$ OP, is the set of permissions.

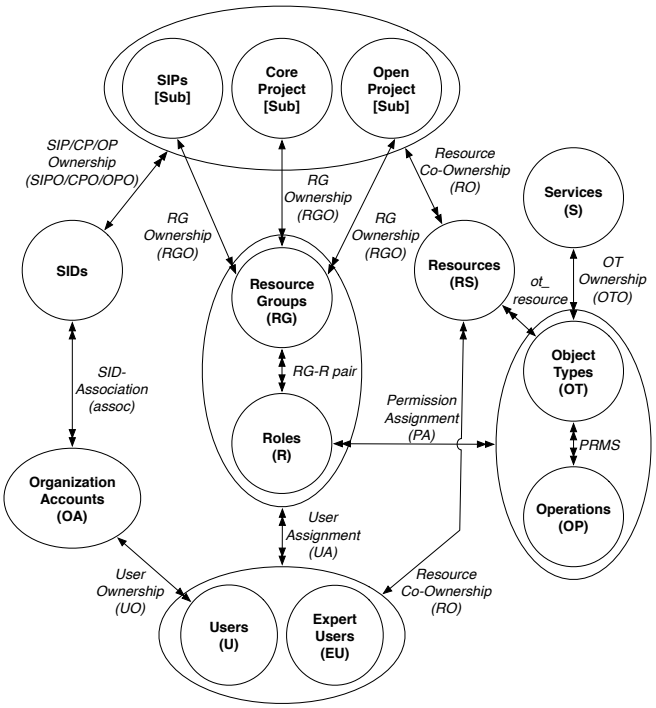

Figure 2: Azure Access Control Model with SID extension (Azure-AC-SID) (ignoring Groups entity)

4. AZURE-AC WITH SID EXTENSION

In this section, we present an access control model for Azure with the Secure Isolated Domain extension (Azure-AC-SID). We extend the Azure-AC-SID model from Azure-AC model to include Secure Isolated Domain (SID) functionality [13]. We present the Azure-AC-SID model so as to cover only the additional components added to the Azure-AC model. Figure 2 shows the Azure-AC-SID model, where we ignore groups for simplicity. In the rest of the paper, group is used to represent a group of organizations, rather than the groups component of Azure-AC model. In our discussion, we assume that a user belongs to only one organization in cloud. For simplicity, we also assume one organization has only one Azure account.

In the following part, we will introduce Azure-AC-SID model. The additional components included in Azure-AC-SID model are described below.

Secure Isolated Domain (SID): Secure Isolated Domain [13] is a special domain, holding security information and resources for cross-organizational security collaborations. SID provides an administrative boundary and a secure isolated environment for cyber security collaborations in a community of organizations. Each SID holds several Secure Isolated Projects (SIPs) designed for cyber incident response and security collaboration among a group of organizations, as well as a Core Project (CP) and an Open Project (OP) for general secure information and resources sharing.

Secure Isolated Project (SIP): Secure Isolated Project [13] is a special project with limited user membership. It is used to collect, store and analyze cyber security information for specific cyber incidents. A SIP provides an isolated controlled environment for a group of organizations within the community to collaborate and coordinate on cyber incidents and other security issues. Subscriptions provide isolated resources containers for different projects to use. Thus, we design projects using subscriptions.

Core Project (CP): Core Project is a shared project holding cyber security committee [9] for the community of organizations. Each organization in the community has representative security users in the committee. Core projects handle the routine security tasks for the community.

Open Project (OP): Open Project is an open shared project where users from the community of organizations share common cyber security information and resources [9]. It is a common forum for all organizational users in the community to share general security information. Information published in Open Project is simply public to every user who is associated with the subscription.

Expert Users (EU): Expert users [9] are external non-organizational professionals. Expert Users don't belong to the group of organizations. They are from other professional security organizations who bring different cyber security skills. They could be from IT consultant companies or from government cyber security law enforcement departments. A SID maintains an Expert Users list which is available to any project inside the SID.

Users (U): Users include both Azure AD users and Non-Azure AD users, which refer to either Microsoft users or partner organization users. We use one entity *Users* to represents all users that are allowed to access cloud resources, since from the stand point of SID functionality, as long as the user is associated to the organization's Azure AD, it does not care where the user comes from.

Organization Accounts (OA): Organization accounts represent organizations in the community. They could be either Azure AD accounts or organizations enterprise accounts which are identified by Azure AD. Organization accounts allows organizations to own specific (virtual) cloud resources.

In the following, we give formalization of concepts introduced above, as well as the relation among them.

Definition 2 Azure-AC-SID model has the following components in addition to AzureAC model.

- SID, SIP, CP, OP, EU, U and O are finite sets of Secure Isolated Domains, Secure Isolated Projects, Core Projects, Open Projects, Expert Users, Users and Objects. Each SID serves one community of organizations. A SID owns a Core Project, an Open Project and a number of Secure Isolated Projects. A SID also maintains resources of its Expert Users.
- Core Project/Open Project/Secure Isolated Project Ownership (CPO/OPO/SIPO) : is a function CPO/OPO/SIPO : CP/OP/SIPO $\rightarrow$ SID, mapping a single core/open/secure isolated project to its owning SID, which equals to map a specific subscription to a SID.
- SID association (assoc): is a function assoc : SID $\rightarrow 2^A$, mapping a SID to all its member organization accounts.
- User Ownership (UO) : is a function UO : U $\rightarrow$ OA, mapping a user to its owning organization account. Equivalently viewed as a many-to-one relation UO $\subseteq$ U $\times$ OA.
- Object Ownership (OO) : is a function OO : O $\rightarrow$ OA, mapping an object to its owning organization account. Equivalently viewed as a many-to-one relation OO $\subseteq$ O $\times$ OA.

4.1 Administrative Azure-AC-SID Model

A SID is an exclusive isolated place for a community of organizations to share their security data and have cyber collaborations. As a public commercial cloud, Microsoft Azure provides full featured API for users to use its functions. Since we can't modify Azure itself, one way to approach SID function in Azure is to build it as a service provided by a third party in the cloud to customers. This requires us to build additional services on the cloud platform.

In general there may be multiple SID providers. Each SID has a Core Project and an Open Project as a security service provided to all organizations in the SID community. Core Project and Open Project are created when the SID is created. Each organization can join different SIDs with different communities of organizations. Each of these SIDs are isolated from each other. We only discuss the model in which the SIDs are manually set up, serving different communities of organizations in Azure public cloud.

We design a SID manager as an automated agent that manages SIDs and their constituent components through their life cycle. It is built using a Python web server as a service to communicate with Azure API. The SID manager processes SID requests from communities of organizations and maintains a separate SID for each community. Within each SID, it facilitates the creation and deletion of SIPs. Each time a cyber collaboration request is sent to the SID manager, it creates a new subscription, assigning the subscription to the group of organizations that made the request. After the collaboration is done, the SIP will be deleted.

Considering that Azure already has its dedicated roles for managing subscriptions and Azure Active Directory, we can use these existing administrative roles from Azure AD roles and subscription Roles to manage SIPs, core project and open project in a SID. Azure provides us five Azure AD admin roles and two subscription admin roles. For simplicity, we are going to constrain our administrative role to include only Azure AD global admin role and subscription co-admin role. Azure also provides a rich set of operative roles in resource groups, which allows users to have permission to access cloud resources.

To make role assignment simple and clear, we constrain roles in two types: administrative roles and member roles, which separately denotes the permission of being able to manage users and permissions only for accessing cloud resources. We use one admin role *SIDAdmin* to represent all admin permissions a user can get from Azure AD and subscriptions. We use one member role *SIDmember* to represent all normal roles a user can get in a resource group. Admin users have the capability to add and remove other users from their home organizations to Core Project subscription or a SIP subscription. Member users can be added/removed from/to a project subscription inside a SID. Member users are the those who actually have access to the cloud services and resources, like creating or deleting a virtual machine.

The administrative aspects of AzureAC-SID model are discussed informally below. A formal specification is given in Table 1.

Initially setup the SID: We design SID as a service in Azure cloud, which is provided by a third trusted party. For every community of organizations who are going to have cyber collaborations, we offer one SID associated with the community. The number of organizations associated with the SID is fixed. Let *uSet* denotes the fixed group of security admin users, each of which represent one and only one organization in the community. Each organization in the community has equal limited administrative power in the SID, which is carried through *uSet*. SID maintains *uSet* as a core group [9] of SID admin users. Only users from *uSet* later can dynamically create SIPs in the SID.

Inside the SID, organizations can request multiple SIPs for convenience of different cyber collaborations. The number of SIPs depends on how many collaborations are initialized by the group of organizations. A SID is initially set up with a Core Project and an Open Project, while organizations can then automatically request to create and delete SIPs, as well as add or remove users from/to SIPs. With the initialization of a SID, admin users from *uSet* automatically get limited administrative permission in a Core Project in a SID, which is represented by role *SIDadmin*. A normal user from the community automatically get permissions to be able to add themselves to Open Project with role *SIDmember*.

Create a SIP: A set of security admin users *uSet* together create a SIP for an cyber collaboration among the community of organizations. The creation of a SIP succeeds based on agreement among the community of organizations. Each organization in the SIP has equally limited administrative power, which is represented by role *SIDadmin*.

Delete a SIP: After the collaboration is finished, a SIP needs to be securely deleted. The delete command is issued by the same subset of the security admin users (*uSet*) who issue the SIP creation. All information data and resources are securely deleted from the SIP. All users assigned to the SIP are removed from it.

Add/remove a user to/from a Core Project or SIPs: Core Project and SIPs admin users are the set of security administrative users (*uSet*) from the community of organizations. These limited administrative users can

Table 1: Azure-AC-SID Administrative model

Operation	Authorization Requirement	Update
SipCreate(uSet, sip, sid) /* *A set of organization security admin users together create a sip* */	$\forall$ u $\in$ uSet.(u $\in$ *uSet*) $\wedge$ sip $\notin$ SIP	assoc(sid) = $\bigcup_{u \in uSet}$UO(u) SIPO(sip) = sid SIP$'$ = SIP $\cup$ {sip}
SipDelete(subuSet, sip, sid) /* *The same subset of security admin users together delete a sip* */	$\forall$ u $\in$ subuSet.(u $\in$ *uSet*) $\wedge$ sip $\in$ SIP $\wedge$ assoc(sid) = $\bigcup_{u \in subuSet}$UO(u) $\wedge$ SIPO(sip) = sid	assoc(sid) = NULL SIPO(sip) = NULL SIP$'$ = SIP - {sip}
UserAdd(adminu, u, p, sid) /* *Admin users add a user from his home account to a Cp/Sip* */	adminu $\in$ *uSet* $\wedge$ u $\in$ U $\wedge$ UO(u) = UO(adminu) $\wedge$ p $\in$ (CP $\cup$ SIP) $\wedge$ (CPO(p) = sid $\cup$ SIP(p) = sid)	UA$'$ = $\exists$ rg $\in$ p.(UA $\cup$ {(u, (rg, *SIDmember*))})
UserRemove(adminu, u, p, sid) /* *Admin users remove a user from a Cp/Sip* */	adminu $\in$ *uSet* $\wedge$ u $\in$ U $\wedge$ UO(u) = UO(adminu) $\wedge$ p $\in$ (CP $\cup$ SIP) $\wedge$ (CPO(p) = sid $\cup$ SIP(p) = sid) $\wedge$ $\exists$ rg $\in$ p.(UA $\cup$ {(u, (rg, *SIDmember*))})	UA$'$ = UA - {(u, (rg, *SIDmember*))}
OpenUserAdd(u, op, sid) /* *Users add themselves to a Op* */	u $\in$ U $\wedge$ UO(u) $\in$ UO(*uSet*) $\wedge$ op $\in$ OP $\wedge$ OPO(op) = sid	UA$'$ = $\exists$ rg $\in$ op.(UA $\cup$ {(u, (rg, *SIDmember*))})
OpenUserRemove(u, op, sid) /* *Users remove themselves from a Op* */	u $\in$ U $\wedge$ UO(u) $\in$ UO(*uSet*) $\wedge$ op $\in$ OP $\wedge$ OPO(op) = sid $\wedge$ $\exists$ rg $\in$ op.(UA $\cup$ {(u, (rg, *SIDmember*))})	UA$'$ = UA - {(u, (rg, *SIDmember*))}
ExpertUserAdd(adminu, eu, p, sid) /* *Admin users add an expert user to a Cp/Sip* */	adminu $\in$ *uSet* $\wedge$ eu $\in$ EU $\wedge$ p $\in$ (CP $\cup$ SIP) $\wedge$ (CPO(p) = sid $\cup$ SIPO(p) = sid)	UA$'$ = $\exists$ rg $\in$ p.(UA $\cup$ {(eu, (rg, *SIDmember*))})
ExpertUserRemove(adminu, eu, p, sid) /* *Admin users remove an expert user from a Cp/Sip* */	adminu $\in$ *uSet* $\wedge$ eu $\in$ EU $\wedge$ p $\in$ (CP $\cup$ SIP) $\wedge$ (CPO(p) = sid $\cup$ SIPO(p) = sid) $\wedge$ $\exists$ rg $\in$ p.(UA $\cup$ {(eu, (rg, *SIDmember*))})	UA$'$ = UA - {(eu, (rg, *SIDmember*))}
CopyObject(u, o1, o2, p) /* *Users copy object from organization accounts to a Cp/Sip* */	o1 $\in$ O $\wedge$ o2 $\notin$ O $\wedge$ UO(u)=OO(o1) $\wedge$ u $\in$ U $\wedge$ p $\in$ (CP $\cup$ SIP) $\wedge$ $\exists$ rg.((u, (rg, *SIDmember*)) $\in$ UA)	O$'$ = O $\cup$ {o2} OO(o2) = p
ExportObject(adminu, o1, o2, p) /* *Admin users export object from a Cp/Sip to organizations accounts* */	adminu $\in$ *uSet* $\wedge$ o1 $\in$ O $\wedge$ o2 $\notin$ O $\wedge$ OO(o1)=p $\wedge$ p $\in$ (CP $\cup$ SIP) $\wedge$ $\exists$ rg.((adminu, (rg, *SIDadmin*)) $\in$ UA)	O$'$ = O $\cup$ {o2} OO(o2) = UO(adminu)

add/remove users of their organizations to/from the Core Project and SIPs. All the users added to the Core Project or SIPs are existing users from an organization's account. The limited administrative users don't have the permission to create new users or delete a existing user. They can only add existing users to the Core Project or SIPs. When users are removed from the Core Project or a SIP, they will lose the access to corresponding information and resources in the Core Project or the SIP, regardless of the ownership of the piece of information in the past. Admin users in Core Project or a SIP can see all users added from the community of organizations, as well as information and resources they bring in, which means there are no hidden users, information and resources in a Core Project or a SIP.

Add/remove a user to an Open Project: Every user in the collaborative community of organizations is allowed to join the Open Project. Users in Open Project have equal but limited permissions. They can share cyber data, but have no control over other users. We use role *SIDmember* to represent this limited permission. Users add/remove themselves from their organizations to/from the Open Project. Users will not be able to access and share any data once they leave the Open Project.

Add/remove an expert user to/from a Core Project or SIPs: Expert Users are needed when external cyber expertise need to be involved. For instance, a cyber incident needs experts from security consultant companies, government cyber experts, cyber polices, etc. SID services maintain a relation with external experts. Expert users can be added/remove to/from a Core Projects and SIPs as a member. Users from *uSet* can request to add/remove expert users to/from the Core Project or a SIP. An existing Expert User in the Core Project or a SIP can also be removed. For instance, at the end of a cyber collaboration, a unneeded expert user will be securely deleted. After the Expert User is deleted, the user will lose all access to any information and resources in the Core Project or a SIP.

Copy data between organization accounts and a Core Project or SIPs: Users can copy data from their home accounts to the Core Project or a SIP. The administrative users from sets *uSet* can export data from the Core Project or a SIP to their home accounts.

5. ENFORCEMENT

Microsoft Azure has been updated to new releases from time to time. We discuss the enforcement of Azure-AC-SID model on the current Azure release. We design the SID service as a third-party-provided secure information sharing service in Azure cloud, for the reason that Azure is a mature commercial cloud platform. In the following part of this section, we first give the basic knowledge of concepts of users and account in Azure cloud. Then we further give the design of how to enforce SID service in context of a Azure public cloud.

Azure accounts form the basic resource boundary in the cloud. To sign in to the Azure cloud, a user has to have either a Microsoft account or a Azure AD account which stores the organization accounts information. In our paper, we uniformly call both of these types of accounts Azure accounts.

For a user to use resources in the Azure cloud, the user has to be assigned to a subscription. A user can be assigned to multiple subscriptions. Each subscription has a trust relationship with one and only one Azure AD, which gives the Azure AD power to authenticate users, services and devices for that subscription. Users have one Azure AD as their home directory to authenticate them, while they can be guest users in other Azure ADs. One Azure AD can

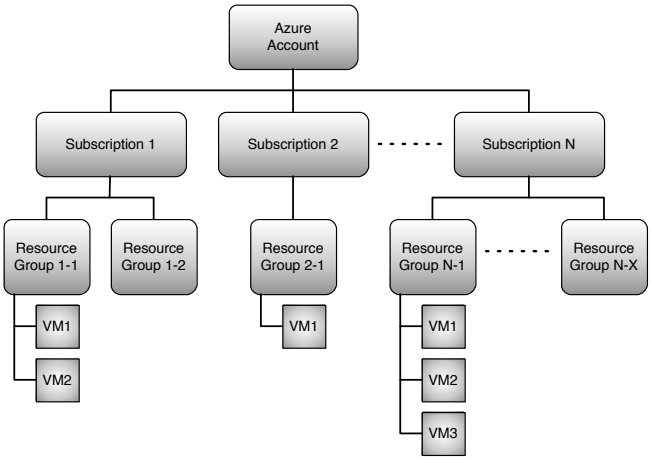

Figure 3: Azure Account Resource Division

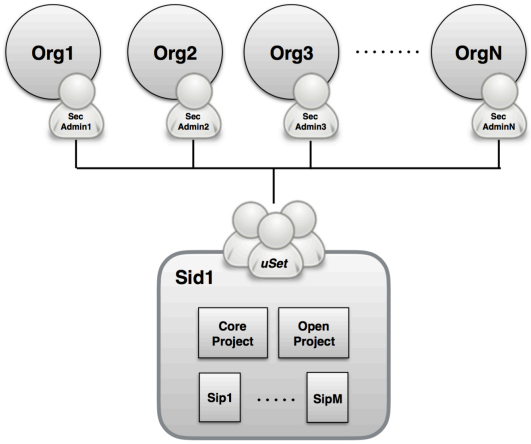

Figure 4: Setup SID Service

be associated with multiple subscriptions. Subscriptions can change their associations with Azure AD at any time. When a subscription de-associates with an Azure AD, users will lose their access to resources through the subscription but still exist in that Azure AD.

Inside an Azure account, subscriptions are sub-divisions of cloud resources while they are separated with each other. Resource groups are further sub-divisions of cloud resources under a subscription. Figure 3 shows the perspective of resources divisions within an Azure account. Azure account owners can create different subscriptions for different management and billing purpose. Azure account owners also assign a service administrator for a subscription, who has full permissions to assign any cloud services and resources to users though the subscription. Each subscription only has one service administrator. Service administrators can further assign co-administrators to the subscription to help assigning cloud services and resources to users.

Azure AD has its own set of administrative roles, including global administrator, billing administrator, password administrator, service administrator and user administrator. For simplicity, we only use global administrator role in our model. Azure AD global administrator can create, edit and delete users and manage user licenses. An admin user has to have both Azure AD global administrator role and subscription administrator to be able to grant a user the access to cloud resources in an Azure account.

In the enforcement, we use a web server to provide SID service. We design SID service to consist of a SID manager account with subscriptions for all the SIDs created in the cloud. SID service web server is the interface which respond to organizations' requests. SID manager account manages all subscriptions in response to the requests.

Azure offers subscription roles and Azure AD roles for administration purpose, which gives a user full administrative power in a subscription associated with a Azure AD. However, administrators in Azure-AC-SID model have constrained administrative power, thus we cannot directly use subscription roles and Azure AD roles. We enforce these constraints through SID manager. The SID manager checks users' account identity information and enforces constraints on users request to a SID.

SID manager maintains a list of security administrative users (*uSet*) from each community of organizations. Each organization in the community has one and only one security administrative user in *uSet*, which represents the organization in the SID, as shown in Figure 4. SID manager also maintains the associations for each SID with its member organizations in the community.

Setting up SID service for organizations: We manually initialize a SID for each community of organizations in the cloud. A Core Project and an Open Project will be created with the creation of a SID, as shown in Figure 4. The security admin users group *uSet* will be added to Core Project as administrators. Every user from the group of organizations will have the permission to add themselves to the Open Project. A security admin users list is created and associated with the SID ID.

SIP request handling When a set of security administrative users from *uSet* send a SIP request to SID manager, SID manager creates a SIP by creating a subscription in the SID manager account, and grant the group of users from *uSet* with role *SIDadmin* to be admin users of the SIP. Role *SIDadmin* gives admin users the permission to add other users from their home organizations to the Sip subscription. Figure 5 shows the process of SIP request.

Delete a SIP: After a collaboration is completed, organizations can request to delete the SIP. The SIP subscription will be deleted with all the information and resources that is created during the collaboration will be cleaned up. All users who are granted access to the SIP will be removed from the SIP subscription.

6. CONCLUSION AND FUTURE WORK

In this paper we presented an access control model for secure information and resource sharing in Microsoft Azure IaaS public cloud platform. We defined the model, gave an administrative model and also discussed the enforcement from practical perspective. We have explored access control models for secure information and resource sharing in the dominant IaaS cloud platform: OpenStack, AWS and Azure. For future work, we would like to compare our models in these three different cloud platforms regarding to the difference of the cloud platforms themselves. We would also like to investigate and compare different concept of roles in

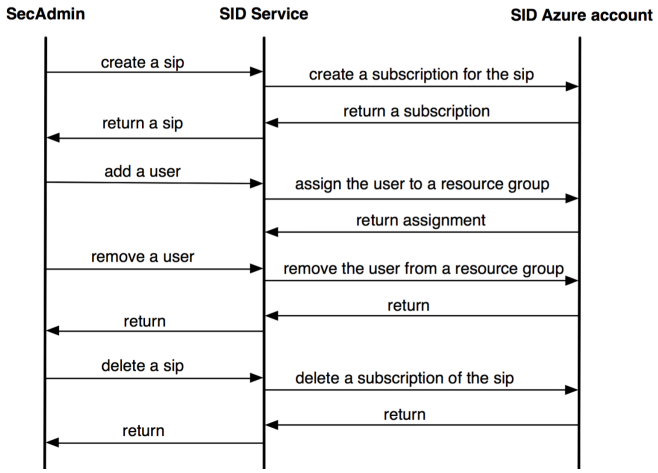

Figure 5: SIP Request

these models, since different cloud platform have different ways to approach permissions to access resources and cloud services.

7. ACKNOWLEDGMENTS

This work is partially supported by NSF CNS-1111925 and CNS-1423481.

8. REFERENCES

[1] http://aws.amazon.com/.

[2] https://azure.microsoft.com/.

[3] https://www.openstack.org/.

[4] http://www.salesforce.com/.

[5] What is sharepoint? https://support.office.com/en-us/article/what-is-sharepoint-97b915e6-651b-43b2-827d-fb25777f446f.

[6] E. Cohen, R. K. Thomas, W. Winsborough, and D. Shands. Models for coalition-based access control (CBAC). In *Proc. 7th ACM SACMAT*, 2002.

[7] R. Krishnan, R. Sandhu, J. Niu, and W. Winsborough. Towards a framework for group-centric secure collaboration. In *5th IEEE CollaborateCom*, pages 1–10, 2009.

[8] M. Raykova, H. Zhao, and S. M. Bellovin. Privacy enhanced access control for outsourced data sharing. In *Financial cryptography and data security*, pages 223–238. Springer, 2012.

[9] R. Sandhu, K. Z. Bijon, X. Jin, and R. Krishnan. RT-based administrative models for community cyber security information sharing. In *7th IEEE CollaborateCom*, 2011.

[10] D. Shands, R. Yee, J. Jacobs, and E. J. Sebes. Secure virtual enclaves: Supporting coalition use of distributed application technologies. In *IEEE DARPA Information Survivability Conference and Exposition*, volume 1, pages 335–350, 2000.

[11] B. Tang and R. Sandhu. Exteding openstack access control with domain trust. In *In Proceedings 8th International Conference on Network and System Security (NSS 2014)*, October 15-17 2014.

[12] B. Thuraisingham, V. Khadilkar, J. Rachapalli, T. Cadenhead, M. Kantarcioglu, K. Hamlen, L. Khan, and F. Husain. Cloud-centric assured information sharing. In *Intelligence and Security Informatics*, pages 1–26. Springer, 2012.

[13] Y. Zhang, R. Krishnan, and R. Sandhu. Secure information and resource sharing in cloud infrastructure as a service. In *ACM WISCS*, pages 81–90, 2014.

[14] Y. Zhang, F. Patwa, and R. Sandhu. Community-based secure information and resource sharing in aws public cloud. In *1st IEEE International Conference on Collaboration and Internet Computing (CIC)*, 2015.

[15] Y. Zhang, F. Patwa, R. Sandhu, and B. Tang. Hierarchical secure information and resource sharing in openstack community cloud. In *IEEE Conference on Information Reuse and Integration (IRI)*, 2015.