iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 **Issue:** VII **Month of publication:** July 2014

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

Identity and Access Management in Cloud Computing

Sonam Sudha, Ms.Vasudha Arora

Student, Assistant Professor

MRIU, Faridabad, India

Abstract: ***In traditional identity management systems user authentication is usually carried out on the basis of management list, previously defined in a system. But with the increasing number of users in such environment as a cloud the management of this list becomes more difficult. For this purpose, in this paper the model supplying the dynamic management of the user's identity federation was introduced. With the implementation of multi-agent systems and the decision making solutions the identity federation dynamization was ensured.Entities (e.g., users, services) have to authenticate themselves to service providers (SPs) in order to use their services. An entity provides personally identifiable information (PII) that uniquely identifies it to an SP. In the traditional application-centric Identity Management (IDM) model, each application keeps trace of identities of the entities that use it. In cloud computing, entities may have multiple accounts associated with different SPs, or one SP. Sharing PIIs of the same entity across services along with associated attributes can lead to mapping of PIIs to the entity. Proper Identity Management may be seen as the first step towards accessing any kind of service from the clouds. It's a person's identity which authorizes him and gives rights to access some data or application in the cloud. Hence Identity theft is perceived as a severe problem and may have disastrous consequences. This paper reviews the trends and approachestaken towards a better Identity management in Cloud computing for a more secure cloud environment.***

I. INTRODUCTION

A. Identity Management

An *identity* is a set of unique characteristics of an entity: an individual, a subject, or an object. An identity used foridentification purposes is called an *identifier* [1].Entity identifiers are used for authentication to service providers (SPs). Identifiers provide assurance to an SP about the entity's identity, which helps the SP to decide whether topermit the entity to use a service or not. Entities may have multiple digital identities. An *Identity Management System* (*IDM*) supports the management of these multiple digital identities. It also decides how to best disclose PII to obtain a particular service. IDM performs the following tasks [2]:

1) *Establish identities:* Associate PII with an entity.
2) *Describe identities:* Assign attributes identifying an entity.
3) *Record the use of identity data:* Log identity activity in a system and/or provides access to the logs.
4) *Destroy an identity:* Assign expiration date to PII. PII become unusable after the expiration date.

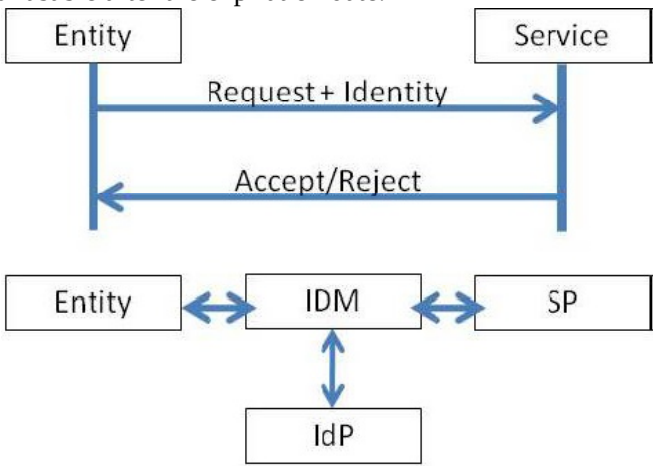


Figure 1. Authentication using Third Party Identity Management

Fig. 1 shows an example of authentication that uses PII.
A user wants to use a service, for which she needs to authenticate to the SP but does not want to disclose her identity data. She has to disclose PII to uniquely identify herself to the SP. The main problem is to decide which information she should disclose and how to disclose it.

Identity Management can involve three perspectives [5]:
1. The pure identity paradigm: creation, management and deletion of identities without regard to access or entitlements.
2. The user access (log on) paradigm: A traditional method say for ex a user uses the smart card to log on to a service.
3. The service paradigm: A system that delivers personalized role based, online, on-demand, presence based services to users and their devices.
A set of parties use IdM and collaborate to identify an entity. These parties are [6]:
1. Identity Provider (IdP): It issues digital identities. For example debit card providers issue identities enabling payment, government issues PAN card or SSN to citizens.
2. Service Provider (SP): It provides access to services to the identities that have the right required identities. For example- a user needs to provide identity information to be able to do transactions via net banking.
3. Entity: Entities are the ones about who claims are made.
4. Identity Verifier: Service Providers send them the request for verifying claims about an identity.
An Identity management system uses one of these three identifiers [4]:

1. That are known by both the entity as well as the service provider

2. That an entity knows and can be verified by the service provider via the identity providers
3. Identifiers like biometric information

II Related Work

B. The Identity Life Cycle:
The Identity Life Cycle is closely related to the concept of digital identities. It comprisesthree main steps[7]:

- Provisioning
- Maintenance
- Deprovisioning

1 Provisioning

The term provisioning is often explained with an example of a new employee. When people join a new company, they often need physical objects like an office, a desk, a phone, a key card, etc. Likewise, many collections of digital information need to be created for the new employee describing who s/he is and what his/her roles and entitlements (access rights and privileges) are within the organization. The allocation of these digital objects and the creation of the digital identity information that enables the necessary services for a user is called provisioning [7].This idea can however be expanded to more than just people joining the company. Many individuals from outside the company might also need provisioning, for example customers, vendors and business partners. Basically, everything that can have and identity that might use provisioning. While provisioning is often limited to people-related identity information, it might hence also include the information of other company assets.
Provisioning often happens when a new identity is created at the beginning of the Identity Life Cycle. Typically, information that describes the object corresponding with the identity is provisioned into Human Resources (HR) systems, operating system directories, application directories, etc. From these newly created Persona's, additional information describing the identity's roles and entitlements within the organization is thencreated. For example, upon joining the company every new employee might have a user account created on a certain server. This Persona might contain identity information like the person's job title. This information can in turn be used to define the roles this person might have. A new employee with "administrator" as its job title might for example automatically be added to the administrators group on said server.

II. MAINTENANCE

Identity information is prone to changes over time. During the Identity Life Cycle,modifications of it will therefore most likely be necessary. Synchronization plays a substantial role here. As the information is updated in one data store, it is often desired that this will be distributed automatically to other data stores using certain synchronization processes that are in place. An example of this would be the change inhome address of a certain employee. This piece of identity information might be modified in the HR system of the company and then synchronized to a server belonging to a department that sends out a monthly magazine to all employees. Maintenance however should not be confused with (re)provisioning. As maintenance is solely about updating identity information it does not cover the creation of

new information that describes persons, groups, devices or services. For example, if an employee of a certain company is promoted, this might cause him/her to acquire new roles and responsibilities. To reflect these new entitlements, the employee's identity information is said to be reprovisioned, not synchronized. As the promotion might cause changes in various relationships, it is possible that news accounts and other data objects must be created in various data stores.

3Deprovisioning

Previously, we learned that when an employee joins a company, through provisioning, itis digital identity is created. In time, while the employee keeps on working at the same company, the identity information will be modified and synchronized. On major changes, like a promotion, the information might even be reprovisioned. When an employee leaves the company however, its identity reaches the end of its life cycle and it is time for the last phase, deprovisioning.

Deprovisioning corresponds with the removal or disabling of Persona's when an identity leaves a domain. For every system that stores one of its Persona (by hand or automatically) a choice has to be made from the following actions:

- Delete the Persona
- Disable the Persona but keep it
- Disable the Persona but delete it later

Deleting the Persona simply means that the Persona will be removed from the data store. This action could have at least two caveats. Firstly, one has to realize that thismight also remove valuable history information, which might still be necessary for auditing later. Secondly, one has to keep in mind that groups or distribution lists the Persona formally belonged too might not update automatically. There are however countless reasons why in specific situations one might prefer disabling a Persona to deleting it. As already mentioned, the account might still be needed for auditing later. Another practical example would be an e-mail account, which might still be useful to forward e-mails to a new address. When a Persona is disabled, basically it will still physically exist but it will not be possible anymore to log in with it, nor to get services previously associated with it.

During an identity's life cycle[7] it might be possible that a Persona mechanism that disables a Persona first but deletes it only after a certain time interval has passed. This is a good way to protect against accidental deleting (for example because of an erroneous edit in an HR application) and might combine the benefits of both previously mentioned actions.

From a security point of view, deprovisioning is a very important step that is often neglected but should be done in a timely and accurate manner. Frequently, identityinformation of former employees can still be found on the systems of organizations and sometimes they even still have working accounts. This not only presents a grave security risk, but also potentially makes the company liable for regulatory compliance, create legal exposure or disturb efficient business processes. We will expand upon this later in this chapter.

During an identity's life cycle it might be possible that a Persona temporary becomes disabled. This might happen for example when a customer does not pay its invoice in time. This process however is not part of deprovisioning but is generally considered maintenance.

B.1 Access Management Systems

According to Access Management systems are systems that support the definition and enforcement of policies and rules that govern access to network resources. To accomplish this, the design of Access Management systems is often remarkably similar to the Identity Life Cycle Management systems previously discussed. Access Management systems consist of one or more repositories, in which access policies are stored and maintained. Furthermore, centralized management of these policies must be enabled and distributed enforcement supported. Due to the similarities in how they work, in practice, Access Management systems can take advantage of the Identity Life Cycle architecture that is present. To distribute policies and rules for example, existing provisioning systems can be employed.

An access policy is a set of rules that determines who is allowed to access certain resources. Most of the time, companies need to be able to identify its users and authorize their access to resources. Sometimes however, due to privacy considerations, the possibility of anonymous or pseudonymous access to company resources might also be needed. In order for the Access Management systems in an IAM infrastructure to be able to offer these kind of services, they should be capable of both identity based and anonymous (or pseudonymous) access control.

Usually, an Access Management system can be categorized as either general-purpose or special-purpose. The special-purpose Access Management systems, which are the most common ones, apply to only one particular system, application or resource. They are also called "resource managers" and include the access-control features embedded in server operating systems, database managers, transaction processing systems or other application environments. Special-purpose Access Management

systems are used to control access to a specific set of files, records, transactions or other resources and services.
General-purpose Access Management systems on the other hand are utilized to controlaccess to a whole range of heterogeneous systems, applications and other resources. Web based access control products are a good example of these kind of systems. They are used to control access to services behind websites and web portals and nowadays typically combine authentication and authorization functions to support multiple authentication systems and use roles-, group- and rule-based systems for scalability. In practice, web-based access control systems often support interoperability with identity data stores and include user management functions, like delegated administration, facilitating their integration into Identity Management Systems.

III. PURPOSED APPROACHES FOR IAM

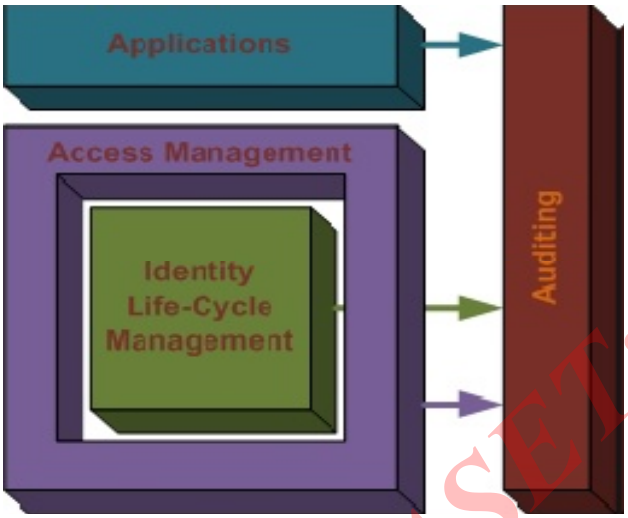


Figure 2: Overview of an IAM system

Figure 2.offers a schematic overview of the complete functional definition of an Identity and Access Management System. As auditing information can also be provided by applications running on top of an IAM solution, an additional component, Applications, is also shown. This component however is strictly seen not part of IAM. In the next chapter of this report, a project will be described in which such an application is developed.
A few times it has already been hinted why a company might choose to develop and implement an IAM solution. As already mentioned however, the impact and complexity of IAM projects is so heavy, that the amount of resources needed and the costs are factors that carry an enormous weight. For this reason, it is critical to answer the question on why one would want to start an IAM project in the first place. To answer this question business challenges often faced by companies nowadays and for which IAM might offersolutions will be discussed in the next section.
Having a keen understanding of the involved business challenges yields great benefits to the consistency and effectiveness of an IAM solution. Another factor that might offer advantages is a sound understanding of the different approaches and technologies that areavailable. The latter will be discussed it

A. IAM Architecture

As we will now talk about technologies however, this functional approach is not very practical anymore. We will hence look at IAM from a more architectural point of view now, starting with the following diagram inspired by [8].

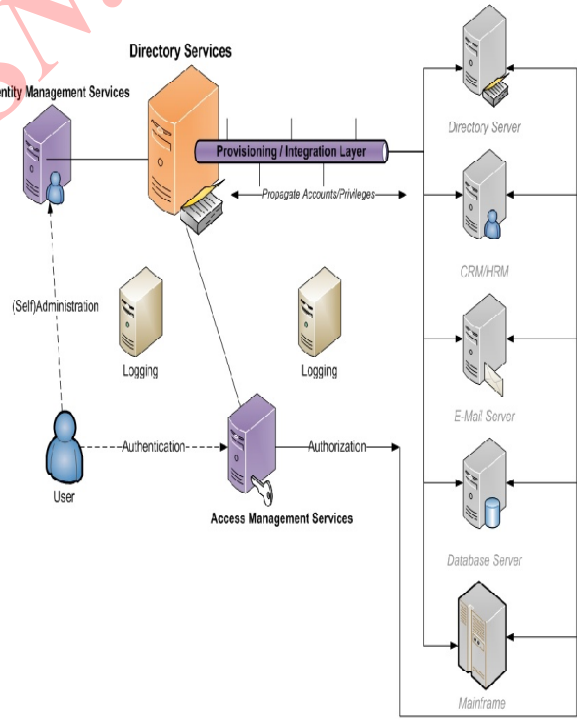


Figure 3: Architectural Overview of IAM

ISSN: 2321-9653

Before we will focus on the different components of IAM systems as depicted in thisoverview, some initial considerations are deemed necessary. In the center of the diagram, we see the Directory Services component, which can be considered the core of IAM. While all IAM components can in fact be deployed independent from any IAM architecture.

The use of Directory Services as a central building block other components can leverageand integrate tightly with[10], could be considered the characteristic feature which distinguishes an IAM solution from any other IT solution offering similar functionality.

On the right side of the overview, some example servers are depicted that could be part of a company infrastructure. In most cases, this would be legacy systems that are not specifically placed there as part of the IAM solution. The (Identity/Access) management of these systems will eventually become the task of the IAM system. Most of the identity data the IAM system works with will come from these systems. Some of them will wholly or mainly provide data, others will receive it. The first group will be called Authoritative Sources, a term which is also used to determine which data source should be leading in case of conflict. If on two connected servers in a company the unique e-mail address of a certain customer differs for example, the e-mail address on the server that is considered to be the Authoritative Source will be taken as the leading one. Possibly, the IAM solution might consequently overwrite the other e-mail address.

B. FEDERATION OF IDENTITIES

On the internet, it is likely that each user ends up with multiple credentials and multiple access permissions across different applications provided by different service providers. These fragmented logins present a challenge to the users and service providers, in forms of synchronization of shared identities, security, etc. There is a strong need for an intrinsic identity system that is trusted across the web and within enterprises and unambiguously identifying users.

Federation of identities maintained by the multiple service providers on the cloud is very critical to cloud based service composition and application integration. An expected issue in this regard is the naming heterogeneity. Different SPs use different factors for authentication like account number, email ID, PayPal ID, etc. Also, when transactions traverse multiple tiers of service hosted in clouds, the semantics of the context of identity information has to be properly maintained, constrained and relaxed as per specific needs. Consider a complete transaction cycle for an e-bay purchase, based on PayPal account .It traverses from e-bay to supplier, through various tiers in supplier's domain to get approvals, release and shipping. Then it goes through PayPal to approve, validate, release the pay, bill the amount to the customer, etc. For each step, the federation [7] authority decides the essential attribute of the customer to be shared with each department.

The user identity mapping in the previous environments have been one-to-one, or in other words, user ID to single user profile. In cloud architectures the mapping challenge is many-to-one, one-to-many and pseudonyms. Pseudonyms are for privacy protection details, when a user does not want his identity to be tracked as he crusades various domains.

Another issue is the trust relation setup between the service providers of the federated world. Currently it is based on policy files framed by the local authority, depending on various factors like the domain trust information automatically fed in by the trust authorities. This is not a scalable or flexible model that can meet cloud computing demands. Cloud scenarios require dynamic trust propagation and dynamic authorization.

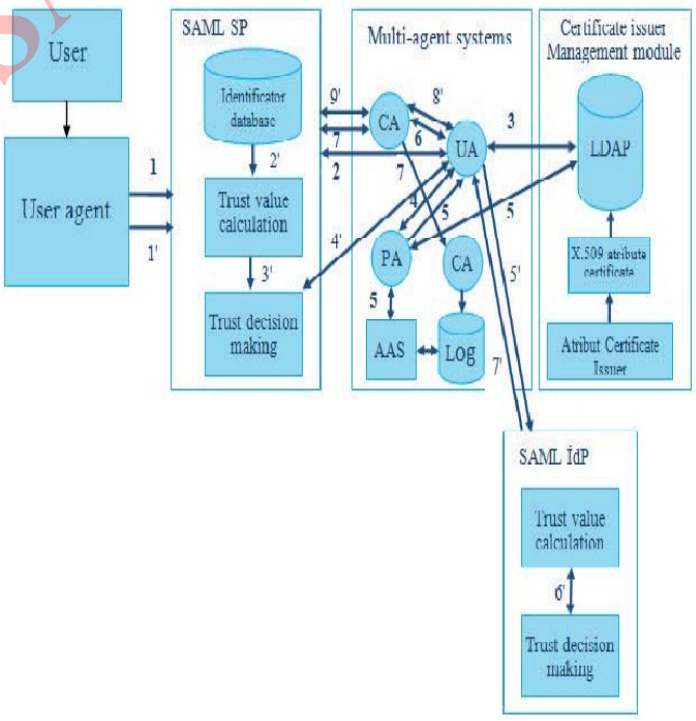


Figure 4 Identity federation dynamic management Architecture

ISSN: 2321-9653

C. Single Sign-on

Providing a username and password is the most common way of authentication. This authentication mechanism however comes with a couple of disadvantages. In order for it to be secure, the password chosen by the user has to be long and complex enough to avoidpeople guessing or brute-forcing it. While this often already makes it hard to remember a single password, the situation gets even worse when multiple resources need to be accessed with different passwords.

The concept of Single Sign-On (SSO) tries to solve this issue. The idea is to make it possible for users to only authenticate once, after which the SSO requested at the beginning of the session automatically authenticates this user for every other resource that might be accessed, given the user has the appropriate access rights of course.

Single Sign-On might greatly increase the efficiency and productivity of users, as no longer they have to remember many difficult passwords and no longer they need to go through tedious login prompts to access the resources they require. The former also has a positive effect on the amount of calls made to the helpdesk, also resulting in reducedcosts. In the next chapter, we will see a real-life example where there were actually so many customers calling the helpdesk because they forgot their password, that the company needed a solution for this to reduce growing costsassociated with this. Finally, as people will be less inclined to use very easy passwords, SSO might also lead to security improvements.

SSO is often implemented by installing Single Sign-On agents on each workstation. This SSO client stores the login information of users when they log in for the first time and passes it to all applications that are accessed and might need authentication. For this to work, all access controls on the resources that are accessed need to have the necessary information to validate whether the information send by the SSO client is correct. As we will see later, the provisioning mechanisms used by the Access Control system in an IAM solution can be used for this.

Sometimes SSO is considered to be one of the building blocks for creating IAM solutions. Whether SSO is a part of IAM or merely a typical application that runs on top of an existing IAM infrastructure can however be disputed.

IV. RELATED WORK

A.Initial Work:

When a company that wishes to implement an IAM solution maintains legacy systemsas those described above, they basically have two choices. One way is to develop their own provisioning, deprovisioning and maintenance solution, which might involve the use of custom made virtual directories and extra metadirectories. Another option is simply to replace the system by a newer one. In the latter case, it is clear that a database migration will be necessary.

The main challenge here is how to make both identity stores cooperate with each other.Ways need to be found to integrate and synchronize the data between the systems. Later, it will also be desired that services can be offered consistently on both sides, something that should already be accounted for in advance.

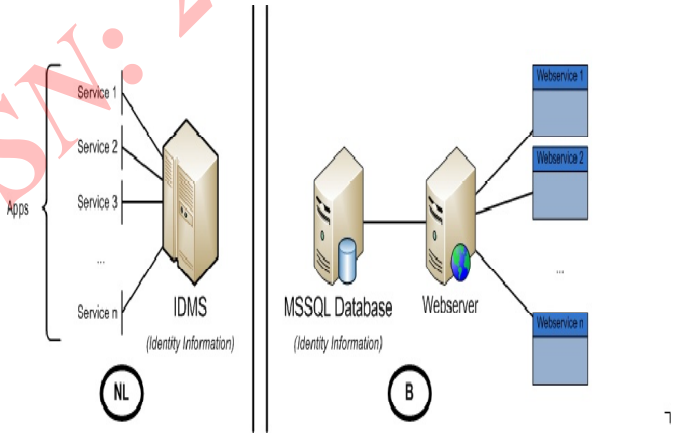


Figure 5 Initial Situation

B. Proposed Solution

After counseling with the customer, the solution presented was proposed solution. The first radical change here is the gradual removal of the IDMS mainframe and its services. The greyed out part of the image depicts this process. This is quite a huge operation as it requires both the data in the mainframe and the services it offers to be transferred to other platforms.

Central in the solution was the addition of a new IBM DB2 relational database at the Dutch site. The primary task of this system is to act as a bridge between the mainframe and the MSSQL database. In order to do so, a data model needed to be defined on this system which can contain data from both

ISSN: 2321-9653

systems. Secondly, ways needed to be found to make this intermediate database able to communicate with the mainframe as well as the MSSQL serverdatabase.

Finally, this database server also needed to act as an Authoritative Source for theIdentity and Access Management system that was developed at the site. This made it possible to provision both the Belgian and Dutch identity information to all other systems connected. As a consequence, consistent services could hence be build on top of the IAM infrastructure, making use of all the advantages this offers.

When we analyze this solution, we can distinguish four mayor tasks that have to be accomplished. Each part will be discussed in detail in the next section, while we will already take a brief look at these mechanisms here.

Firstly, a way needs to be found to bring the data in the IDMS mainframe to the new intermediate DB2 database. This task consists of two parts. The first step is to bring all the current data to the database and later also new data entering the mainframe needs to be synchronized. This is a result from the fact that the services on the mainframe will be migrated gradually, as already mentioned before. Part of this task also consists of developing a data model on DB2 which can hold the data provided by the mainframe. This data model will depend on the solution chosen for the migration and synchronization however. Secondly, we also need to be able to migrate and synchronize data between our intermediate and Belgian MSSQL database. As it is also desired that the MSSQL database would get all Dutch data, this migration and synchronization needs to be bidirectional.

The good news however is that both databases use very recent and popular technologies, making this task rather common and well documented. As a footnote, it should be noted that the MSSQL also contains all historical data. This means that on every modification of the data, the old data will still be stored in the history such that rollbacks to any situation in the past are possible. As, partially because at the Dutch site no such history exists, migrating this data would make things very complicated, it was decided to keep the historical data out of the scope of this project. Again, this step requires the development of a data model on the DB2 database.

Up to now, we saw that the two mechanism discussed above require the development of two separate data models on the intermediate DB2 database. The next logical step is to synchronize these two data models. This task requires the search for a technique on DB2 that could detect inserts, updates and deletes in one data model and translate them into respectively inserts, updates and deletes on a second data model. This step needs quite some logic and as a result needs a lot of custom development. For this reason, the development of this particular mechanism can be considered the main part of the project.

The final step consists of making the DB2 database part of the IAM architecture. .

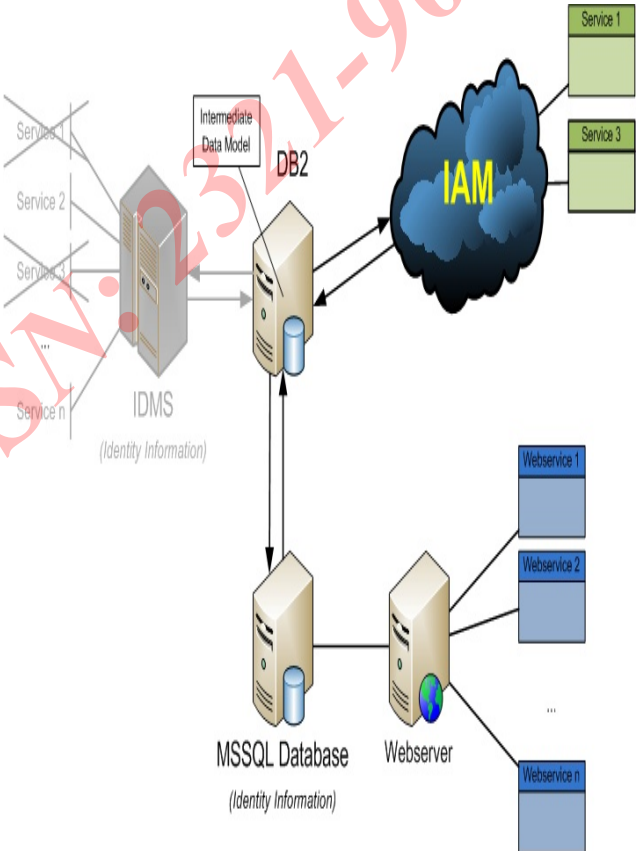


Figure 6. Proposed Solution

V CONCLUSION

A.1IAM defined and implemented in large organizations that use cloud services

The most used and referenced definition is the model Gartner presented in the paper 'Identity and Access Management Defined' (Witty, Allan, Enck& Wagner, 2003). It describes Identity and Access management as a set of capabilities in the domains of administration of identities and rights and real time enforcement of these rights. Enterprises need to ensure that users are properly identified and that these identities are validated to IT resources -this is *authentication.* They need to know that users can only access what their job function allows them to access within the enterprise -this is *authorization.* They need to have a consolidated, enterprise wide view and way to manage user access — this is *administration.* Finally, they need to ensure that the activities associated with user access (administration and real-time enforcement) are logged for day-to-day monitoring, regulatory and investigative purposes — this is *audit.*

Identity & Access management in the cloud domain can be divided into 3 types of implementations.

- IAM for SaaS that allows users to access and manage cloud-based business applications.
- IAM as SaaS that moves provisioning into the cloud.
- IAM to SaaS that integrates on-premises IAM solutions with SaaS business applications

Based on the empirical research it can be concluded that IAM for and to SaaS types are predominantly implemented in large organizations mostly to provide seamless access andcontrol of SaaS and cloud based solutions .

VI. REFERENCES

[1] A. Josang and S. Pope. User Centric Identity Management, In Proc. AusCERT, Gold Coast, May 2005.

[2]Wikipedia.IdentityManagement Systems. July 2010. http://en.wikipedia.org/wiki/Identity_management_systems.

[3] M. Linares. Identity and Access Management Solution (Version 1.4c). GIAC Security Essentials Certification, Practical Assignment, 2005.

[4]P.Angin,B.Bhargava,R.Ranchal,N.Singh,M.Linderman,L.B. Othmane,L.Lilien;"An Entity centric Approach for Privacy and Identity Management in Cloud Computing",29th IEEE International Symposium on Reliable Distributed Systems;2010.

[5] S.Subashini, V. Kavitha;"A survey on security issues in service delivery models of cloud computing";Journal of Network and Com

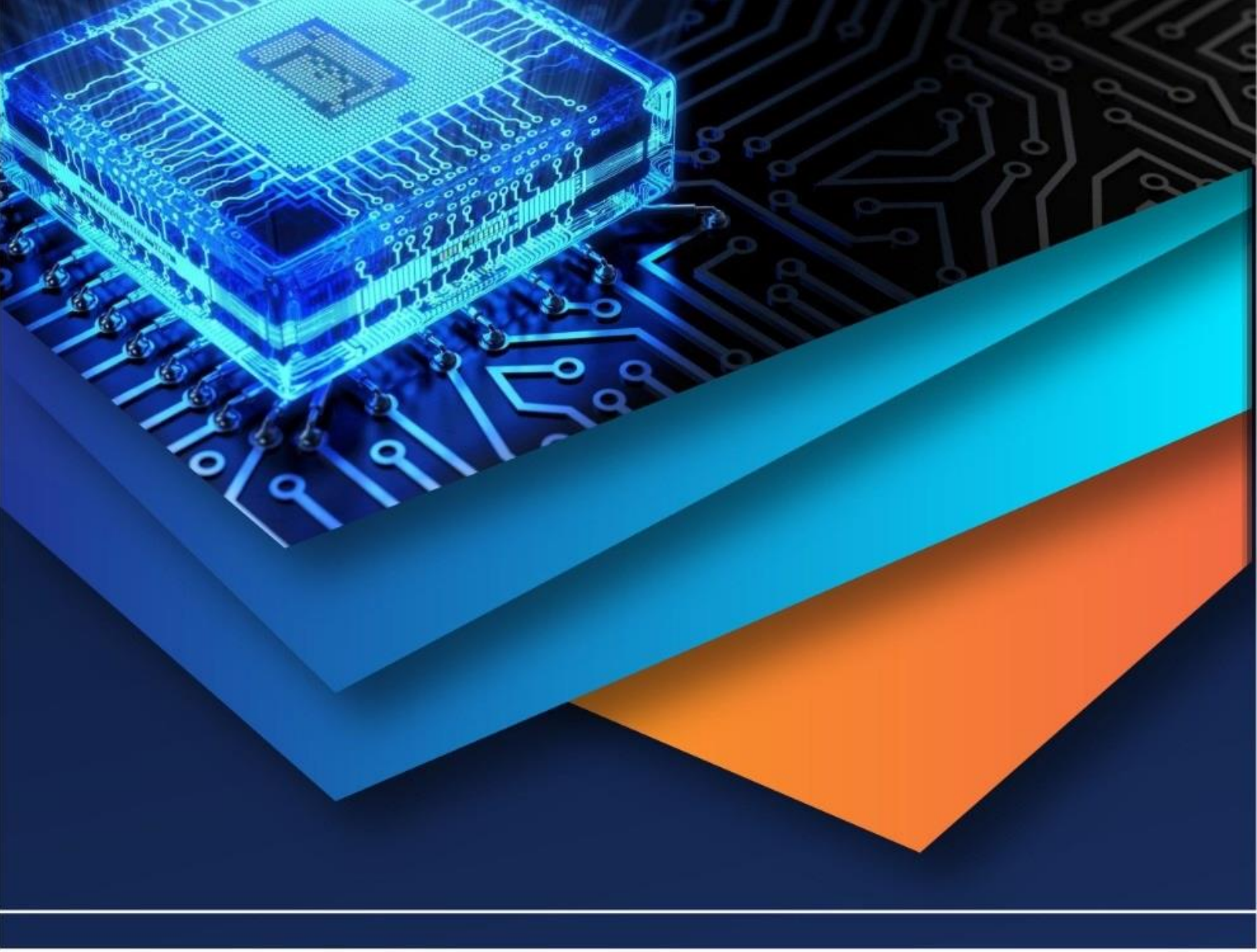
[6] K.Cameronand M.B.Jones;"Design Rationale behind the Identity Meta system Architecture";Jan2006.http://research.microsoft.com/enus/um/people/mbj/papers/Identity_Metasystem_Design_Rationale.pdf

[7] OWASP homepage. http://www.owasp.org. Accessed 17 Mar 2007.

[8] ISO 27001. http://www.iso.org. Accessed 17 Mar 2007.

[9] DirX Solutions - Totally Integrated Identity Management. Siemens, 2003.

[10] IT Governance Institute. Enterprisewide Identity Management: Managing Secure and Controllable Access in the Extended Enterprise Environment. ISACA, 2004.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY