Designing dynamic federated identity management framework for reduction of management overhead in cloud computing

Siboniso C. Makhaye, Paul Tarwireyi, Mathew O. Adigun and Anass Bayaga
Email: BayagaA@unizulu.ac.za, Department of Computer Science, University of Zululand
Private Bag X1001, KwaDlangezwa, 3886
LL Lekena, Tshwane University of Technology, Quality Advisor: Surveys and Institutional Research
Quality, Planning and Risk Management

***Abstract*-- Open and dynamic environments pose many challenges for current identity management (IdM) solutions. This is owing to the fact that IdM systems are not designed for open and dynamic environments. The disadvantage with current federated identity management solutions is that; trust between service providers (SP) and identity provider (IdP) in a federation is pre-configured or established at design time, which limits the cooperation with potential business partners. There have been numerous attempts to develop a dynamic federated framework that will establish trust relationship dynamically between entities from different security domains. The framework should also support an independent (trusted) entity that will facilitate dynamic trust negotiation between non trusting parties, thereby reducing management overhead. In addressing the issue of management overhead, we propose a dynamic approach for the federation of identities. This paper proposes a Dynamic Federation Identity Management (DFIM) framework which will enable dynamic trust negotiation between non-trusting entities in open environments. Our framework uses dynamic trust engine that evaluates user claims using security assertion mark-up language (SAML v2.0) between communicating entities. We discuss component interaction and the design of DFIM framework aimed at reducing overhead in the management of identities. Lastly we also present a usage scenario of our proposed framework to show its applicability in open environment.**

Index Terms— federated identity management, management overhead, dynamic trust, trust negotiation

I. INTRODUCTION

Cloud computing is a new paradigm that enables ubiquitous, convenient, on-demand network access to shared pool of configurable computing resources based on an off premise pay-as-you-go operational model [1]. Enterprises have expressed concerns about adopting the cloud environment for critical workloads. One of the most cited reasons for not moving to the cloud is the concern about cloud security, particularly identity management (IdM) [1] [2]. The disadvantage with current federated identity management solutions is that, the trust between service providers (SP's) and identity providers (IdP's) in a dynamic environment is pre-configured or established at design time, which limits the cooperation with potential business partners. This means that the willingness of the SP to relay information on the IDP depends on a pre-established business agreement. Therefore, there is a necessity to develop a dynamic federated identity management framework that will establish trust relationships dynamically, between entities from different security domains. The framework should also support an independent (trusted) entity that will facilitate dynamic trust establishment between non trusting entities, thereby reducing management overhead. Overhead can be taken as the cost of general processes of an application that cannot be traced to any task performed by the user or other third-party entities. This process shows an increase in tasks without an increase in productivity [3].

The main aim of this investigation is to improve upon existing identity management solutions for the purpose of reducing IdM overhead through establishing trust dynamically in open environments which includes cloud computing.

There have been several approaches towards a federated identity management framework, but less effort has focused on the dynamic trust establishment in cloud computing environments. There are many federated identity protocols which are based on open standards. The most popular ones are Liberty Identity Federation Framework (ID-FF 1.1), Security Assertion Mark-up Language (SAML v2.0) and WS-Federation, as remarked by Zuo et al [5]. In addressing the issue of management overhead in open and dynamic environments, the works in [5] and [6] have become the inspiration and the basis of our approach. Both Gao and Zuo have contributed immensely towards improving dynamic trust approaches in different computing paradigms by providing dynamic techniques towards establishing trust between non trusting entities [5] [6].

Work in [6] has demonstrated that, in order to realize business relations with efficiency between strangers, advanced IdM techniques should be the pillar of business interactions through the introduction of automatic processes for the evaluation of trust. But this paper further proposes a Dynamic Federation Identity Management (DFIM) that will enable trust relationships between two or more non-trusting entities to be established dynamically at runtime. The main distinguishing factor between current federated identity management techniques and the one presented in this paper is that, the approach of this paper is designed to suit open and dynamic environments, for the purpose of establishing trust dynamically between non-trusting entities.

The rest of the paper is organized as follows; Sections (II) and (III) indicate the background on trust negotiation and work

related to this study. Section (IV) we present our framework, component interaction diagram and a scenario. In Section (V) we present the conclusion of this effort along with future challenges to be addressed.

II. RELATED WORK

Several initiatives towards a more comprehensive IdM reference framework are underway in the industries and also in academia. Both have stressed the importance of a dynamic identity management solution, which could reduce management complexities in open and dynamic environment. Identity management schemes that have been developed are comprised of three major components [7]. The following are the three major components in current IdM solutions.

- **Service Consumer (SC)** – Is a client whose legal identity and attributes are stored in IdP to access the desired service. Users (SC) interact with the browser to request a service.
- **Service Provider (SP)** - Is an entity that provider's services to requesting service consumers over a network.
- **Identity Provider (IdP)** - Is the heart of the IdM system which provides the authentication and authorization of the users to legally access a service.

Models that are very common in the development of IdM Solutions include isolated, centralized and federated models.

Isolated identity management -In such a framework there is no co-operation between participating parties to support user authentication. The Service Provider trusts only, the identities that are stored in it repository. This system is used a lot in online applications and resources, because it is relatively simple for service providers to manage, but it is rapidly becoming unmanageable for users [7]. The exponential growth in online services has led to users being overloaded with identifiers and credentials (different logins and passwords) that they need to remember and manage and for this reason most users end up creating simple and weak passwords or alternatively use the same passwords for different accounts [8]. Hence, new innovative identity management models are being proposed and implemented [9].

Centralized identity management- A framework of this type has a single IdP that provides identity services to the participating SPs within a closed domain or Circle of Trust (CoT). One of the first approaches has been Microsoft Passport, which is based on the so-called centralized model. Under such an approach users make use of single sign on (SSO) authentication mechanism, therefore they can authenticate once and gain access to protected resources across multiple systems [10]. Here a central Identity Provider (IdP) is responsible for collecting and provisioning an individual's identity information. A major drawback of such approach is that the IdP is potentially a single point of failure and often, is not trusted by all participating parties.

In federated identity management- (FIM) systems, this technique is usually confused at times, in some instances it is referred to as the collaboration of several SPs to use a single IdP, all enclosed in one CoT [10]. The FIM is viewed differently in this paper, this paper views FIM as a setup where identities are being shared with different security domains, and thus this approach incurs lot of complexities. Identity providers (IdPs) needs to be up and running at all costs in order to provide authentication service to users who are requesting a protected service [7]. FIM require a prior trust between identity provider and service provider in order for a user to be successfully authenticated to access a protected service. FIM allows for easy revocation of attributes, which is done by the identity provider refusing to issue security token to revoked attribute [7]. This approach also incurs a lot of management overhead in dealing with requests from different administrative (security) domains, for which the effort of this paper is trying to overcome.

The diagram below (Figure 1) shows the evolution of federated identity management techniques that are based on open source and standardizing organizations in open environment. This figure also shows how SAML versions have been released in relation to other FIM techniques up to the current version SAML 2.0, which is the base of our approach.

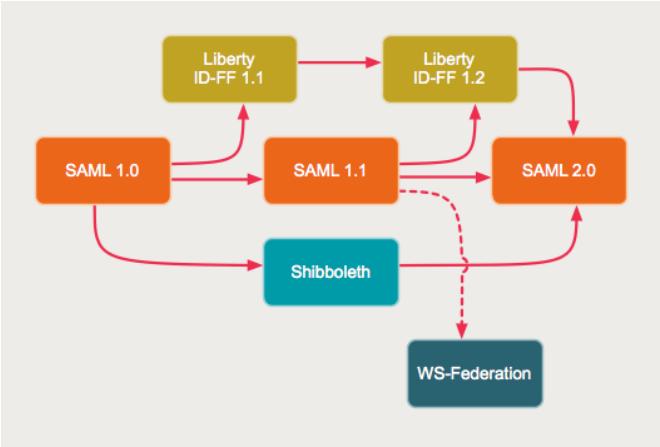

Fig. 1. Evolution of federated identity management [11]

III. TRUST NEGOTIATION

Trust negotiation is the approach that has been proposed to address the limitations of existing access control solutions when used in the context of open systems. Trust negotiation technique tries to make a decision about whether the identity provided by the user allows that user to carry out the requested operation [7].

In traditional open environments, SP's and SC's are parties that usually know each other before any request could be initiated. Information that is shared in the environment usually determines which parties can provide what kind of services and which parties are entitled to access those services [12]. Hence, trust between parties is not a troublesome issue. Even if trust issues can arise, as it does in

Some occasions, But in traditional client-server systems, the

question is whether the server should trust the client, and not vice versa. In this case, trust establishment is often handled by bilateral access control methods, such as having the client log in as a pre-registered user [7]. One major notable difference between the dynamic environments such as cloud computing and traditional environments is that; cloud computing provides an environment where parties establish connections and interact without previously having knowledge of one another.

Several identity management systems and techniques towards a more comprehensive dynamic trust negotiation have been developed. These techniques address the issue of management overhead by establishing trust dynamically in open and dynamic environment.

Automated trust negotiation (ATN) is the approach to gradually regulate the exchange of secret digital credentials between non-trusting parties. Various ATN systems have been implemented and deployed; which include TrustBuilder [13] and Trust-X [14]. TrustBuilder is an architecture that focuses on trust negotiation techniques. The architecture varies credentials and checks policy compliance. Other systems like Traust [13] have been developed to enhance TrustBuilder to provide interaction between applications or systems that offer trust negotiation services. Trust-X is a technique that provides an XML-based language that makes use of policies and certificates for trust negotiations. It also support a peer-to-peer architecture used for trust negotiation management.

Dynamic trust negotiation (DTN) is the process of establishing trust between strangers or non-trusting parties, through which they rely on an intermediary to govern their communication. Whereas Automated trust negotiation (ATN) approach gradually regulate the exchange of sensitive digital credentials between strangers.

DTN can be differentiated from ATN in that [7] negotiations occur between trusted parties and not between strangers even though the goal is to establish trust between strangers. One of the most challenging research issues in identity management has been on how to establish trust between non-trusting entities (total strangers) from different security domains.

IV. RESEARCH MOTIVATION

Until now there is no single coherent approach to IdM framework that exists for dynamic and open environments, which makes interconnection of custom solutions a problem, thus reducing the wide acceptance of IdM schemes (ITU, 2007). Moreover, existing solutions that have been implemented are not prepared for dynamic environments, since they do not address the issue of dynamic trust negotiation and establishment capabilities. At the root of the problem there is a lack of a universal model and a lack of IdM reference framework architecture which could serve as a guideline when designing and developing IdM systems [15].

Current research has shown and proven that in order for IdM framework to be widely adopted in the security research community, the framework should be based on profound standards and specifications. Standards should allow easy future advancements of the framework in terms of its interoperability with other components which are yet to be developed in the future.

During the course of our investigation, we identified a number of design criteria, principles and laws that needs to be taken into consideration, if one wants to design federated identity management framework for dynamic environment. These are four key design criteria which are not strongly supported in current identity management solutions and significantly reduce management overhead [16].

C1. Dynamic bilateral trust establishment- In dynamic environments like cloud computing environment services are autonomous, therefore an identity management solution cater for dynamic environments and give rise to solutions that can establish trust relationships dynamically between entities from different security domains.

C2. Communication security- the interaction amongst all entities in the federation should be secure in order to provide confidentiality and integrity for the session.

C3. Claim based Authorization- the identity management solutions should have a Claim-based authorization, responsible for facilitating negotiation based on user claims. This approach addresses more scenarios than the popular role based security and is also useful when an application requires complex and fine grained control on expressing access control decisions. Role based security model may not be powerful or flexible enough and is often too coarse when we reach complex and dynamic scenarios.

C4. Trusted third party reliance – the framework should support certificate authority that is trusted by the communicating parties for the purpose of facilitating integrity for certificates in circulation.

After a thorough investigation on the four design criteria that we have been described in this section, evaluation shows that federated identity management can serve as the basis in developing a framework for the reduction of management overhead. While considering the design of IdM, we classified the current three IdM models – isolated model, centralized model and federated model based on the design criteria we have formulated for the dynamic trust establishment.

Table I Comparison of identity Management Models

	Design Criteria			
Models	**C1**	**C2**	**C3**	**C4**
Isolated	**X**	**✓**	**X**	**X**
Centralized	**X**	**✓**	**X**	**X**
Federated	**X**	**✓**	**✓**	**X**

Among these three models, the federated identity management model has the most potential with the high

flexibility and support some of our design criteria as indicated in Table 1. That is, federated identity management model forms the basis of solving management overhead problem noted in Section 1. It is noted that identity management models above are not effective in open environments, which implies a significant issue that needs to addressed, hence there is a need to redesign and develop an identity management framework that supports all mentioned criteria in order to achieve the ultimate goal of reducing management overhead in dynamic environments

VI.THE PROPOSED SOLUTION APPROACH

This paper presents an identity management framework that, would allow non-trusting entities to be able to interact and establish trust relations dynamically in an open environment. Before any meaningful interactions between trusting and non-trusting entities can starts, a certain level of trust must be established [4]. Generally, trust is established through disclosure of secret information between non-trusting entities that want to exchange critical resources. Since neither party is known to the other, this trust establishment process should be bi-directional. Both parties may have secret information that they are reluctant to disclose until a trusted entity is provided, which will evaluate the validity of the secret information.

The proposed framework is based on Security Assertion Mark-up Language (SAML V 2.0) standard in order to provide communication security between interacting entities, which is anticipated to provide confidentiality and integrity throughout the session. Our framework is more concerned about reduction of management overhead which emanate from manual procedures before a non-trusted entity could be given access to requested resources. Our approach uses universal policy logic to maintain consistency and facilitate smooth trust negotiations. To fulfil the design criteria, our proposed framework employs a fourth player in its dynamic trust management framework which brings about dynamicity in our framework. It makes use make use of a Trusted Third Party (TTP) – which is entity that facilitates the communication between two entities that both trust the entity also acts as a signer of certificates possessed by two entities. TTP act as a pillar of trust, for which each entity holds accountable should the information given by an entity defaults.

A. Proposed Framework

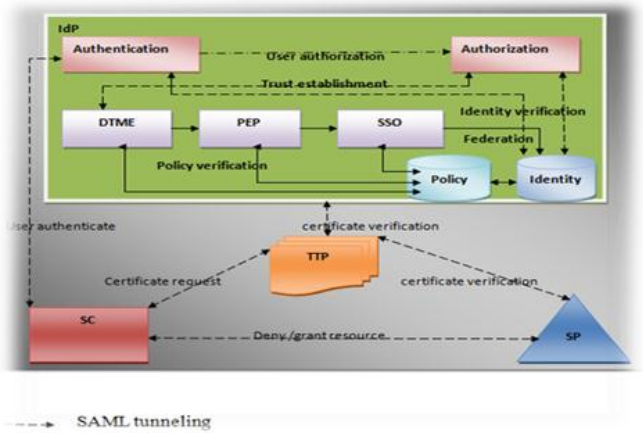

Fig. 1. Dynamic Identity Management Framework

Dynamic trust management engine (DTME) - this component is responsible for processing internal and external trust relations evaluating user claims against the service policy access and then relay the result of the evaluation to the service. This logical component (DTME) is also responsible for making trust decision between strangers at runtime.

Policy Enforcement Point (PEP) - this component helps service providers to make better decisions whether to trust the service consumer or not. Also facilitates the exchange of secret information (certificates) using SAML as a protocol for the transmission of data. Trust policies should be carefully designed and all possible risks should be taken into consideration while doing so. Will also implement single sign on principle that will allows service consumers to access services over the Internet using only one user credentials over a distributed network to access different resources. Thereby reducing time it takes to authenticate and facilitate the management of identities.

B. Protocol for interaction

The sequence diagram below figure 3 shows the interaction between four components of our framework.

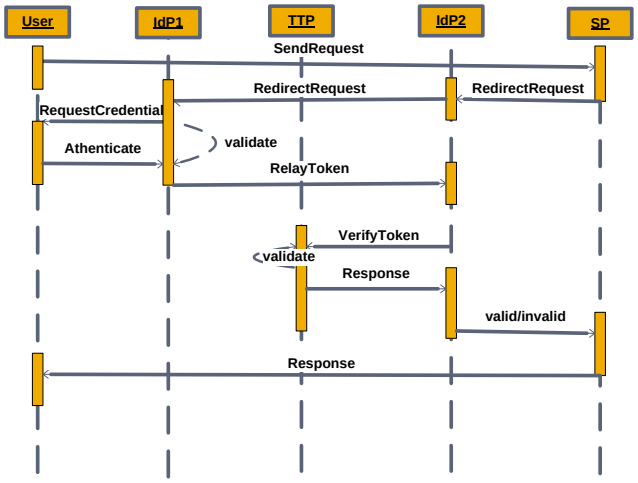

Fig. 3. Protocol for dynamic trust

Interaction Sequence

Step 1: The user sends a request to an external resource.

Step 2: The service redirect the request to its IdP.
Step 3: The IdP sends the authentication request to the IdP of the user and the IdP of the user relays the request to the user.
Step 4:.The user uses logon details that are authorized to its IDP, to request access to the service that is outside the domain of trust.
Step 5: The internal IdP responds by sending the Credentials (Token), also the trust negotiation request and other policy requirements (claims) to access the service.
Step 6: The external IdP validate the credentials (token) with the trusted third party (TTP) and then sends the claims to the service as parameters.
Step 7: The service accept or denies the request based on claims given.

C. Implementation

We have used open source development tools in order to implement the prototype of our framework. We used CXF, which is open source framework that supports the development of APIs such as JAX-WS and JAX-RS. CXF is easier to use, support spring framework and WS-extensions such as WS Security, WS Trust, WS Federation and WS Secure conversation. Service providers (SPs) are deployed in apache tomcat server and they expose, service name, service type. These SPs use parameters as input fed to evaluate the incoming request of acquisition of a resource or service.

For the service to be acquired by the requesting user, the DTME checks the user certificate status (valid/invalid) if certificate is invalid then access is denied, else the DTME further checks the user claims against the service policy, if the user claims comply with service policy then the service is given to the requesting user .

DTME is implemented using claim based trust negotiation to establish trust dynamically between non-trusting entities. User claims represent attributes about an identity and request and responses in SAML for communication security. Identity provider (IdP) implementation is based on WS Federation which is an open source package for web SSO across or within organizational boundaries. Its scalability and flexibility makes it suitable for our framework. We have used the openLap to provide directory, which is an open source implementation of the lightweight directory access protocol to store user and service provider information.

Trusted Third Party (TTP) component is developed in an independent machine. It responsibility is to create, sign and verify x509 certificates. For which in our case certificates are self-signed for simplicity purposes of our implementation only. But in true business environment well established TTP's are used to sign and verify certificates for production purposes, which include VeriSign, Comodo and DigiNota.

Trust is brokered through TTP which accept a token (X 509 certificates) and then check for the signature value, signed by the same TTP. This trust evaluation can also be established between complex permissions and divergent services, through the evaluation of the signature value in the certificate. TTP tag <ds: *SignatureValue*> encapsulate the private key of the TTP, which is calculated from the basic formula digital signature **s** is generated on a message m according to the equation: $\mathbf{s = md \bmod n}$. Where (n, d) is the signer's RSA private key. The signature is verified by recovering the message m with the signer's RSA public key (**n, e**): $\mathbf{m = s^e \bmod n}$ [17], this is also stated in our usage scenario.

D. Implementation Scenario

The demonstration of our dynamic framework presented in a usage scenario. We have created two security domains (companies) namely booking domain and the hiring domain. Security domain involves users, data, systems and devices that adhere to the security policies of a certain domain.

From the booking domain we have implemented a protected web service called airline booking and in the hiring domain we have also implemented a protected web service called car hire service. The user Bob is a registered member of booking domain and can easily access the services provided the Booking domain. At this time He would like to hire a car from another domain which is Hiring domain. Bob clicks the link of car hiring service but the service (car hire) presents logon option web page to Bob. Since Bob has not registered with hiring service and He is very reluctant to comply with the frustrating registration process with Hiring domain, Bob then chooses to use booking services (ID) during logon process.

Hiring domain is a stranger to Booking domain (non-trusting domains), hence Hiring domain request Bob's certificate signed by the TTP. Also Booking domain request the car hire service to provide service description, this includes service name, service domain, and service type for the purpose of proving the endorsement of the service by the trusted third party as shown in figure 4 below. Hiring domain validates the particulars of the user through the session between the hiring domain's IdP and trusted third party (TTP).

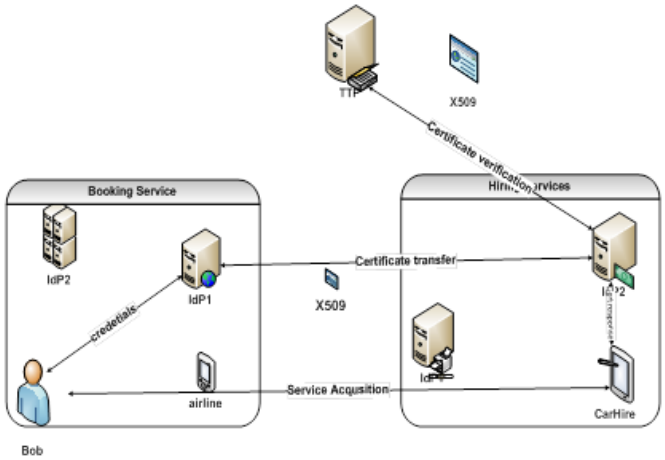

Fig. 4. Implementation scenario diagram

After the certificate validation has been successfully completed, then car hire service connects to Bob and request Bob's attributes (Claims) to satisfy the resource access policy. These attributes includes Bob's full name, driver's license, ID,

age, sex and cell phone number. Car hire initiate a trust negotiation session for user attributes exchange. Booking service sends an SAML attribute assertion of Bob to hiring service. Hiring services allows Bob to access car hire service.

VII. CONCLUSION AND FUTURE WORK

This paper presented a dynamic federated identity management (DFIM) framework for the reduction of management overhead in dynamic environments. Our approach has portrayed the real world situation, where there is more than one IdP even in the same trust domain. Showing untrusted parties penetrating different trust domains to access restricted or protected services. The proposed approach is envisaged to significantly reduce management overhead, as depicted in our usage scenario. This work is an extension of our previous research effort [18] which puts more emphasis on the reduction of management overhead in identity management solutions deployed in cloud environment.

In future, we intend evaluate two performance metrics response time and the throughput to prove that management overhead is drastically reduced through the intervention of our IdM solution. Furthermore, we intend to investigate possible security vulnerabilities which includes, Malicious service provider, dishonest identity provider, man in the middle, active eavesdropper and denial of service adversary in dynamic open environment. There are still a number of issues that hinders the effectiveness of current IdM solutions, Hence the introduction of our approach.

REFERENCES

[1] Juniper Networks, "Identity federation in a hybrid cloud computing environment solution guide," .2009

[2] Oracle, "Service-Oriented Security an Application-Centric Look at Identity Management," April.2010

[3] Denning P.J, "Overhead," .2008

[4] Linn J, RSA Laboratories, "Trust Models Guidelines." Feb.2004

[5] Zuo .Y, Luo X, and Zeng F, "Towards a dynamic federation framework based on SAML and automated trust negotiation," in Proceedings of the international conference on Web information systems and mining, Berlin, Heidelberg, pp. 254–262. 2010

[6] Boursas and Danciu V. A "Dynamic inter-organizational cooperation setup in Circle-of-Trust Environments," in Network Operations and Management Symposium, NOMS.IEEE.pp.113 –120. 2008

[7] Josang A. and Simon P, "User Centric Identity management," presented at the AusCERT Conference, April 2005

[8] Tarwireyi P, Flowerday S and, Bayaga A "Information security competence test with regards to password management," in Information Security South Africa (ISSA), 2011, pp. 1 –7. 2011

[9] Hong kong Admin Region, "Identity Management,"

[10] Alpar G, Hoepman H and Siljee J (2011, Jan) "The Identity Crisis. Security, Privacy and Usability Issues in Identity Management," arXiv: 1101.0427, 2008

[11] Ellie K "Identity I: SAML Single Sign On Tutorial." May .2011

[12] Abliz M "Negotiating Trust in Identity Metasystem," .2009

[13] Ajayi O, Sinnott R and Stell A, "Dynamic trust negotiation for flexible e-health collaborations," New York, NY, USA, pp. 8:1–8:7. 2008

[14] Cao Y and Yang L "A survey of Identity Management technology," in Information Theory and Information Security (ICITIS), IEEE International Conference on, 2010, pp.287 – 293. 2010

[15] Darbrowski M and Pacyna P, "Cross-identifier domain discovery service for unrelated user identities," in Proceedings of the 4th ACM workshop on Digital identity management, New York, NY, USA, 2008, pp. 81–88. 2008

[16] Baldoni R, "Federated Identity Management systems in e-government: the case of Italy," Electronic Government, an International Journal, vol. 9, no. 1, pp. 64 – 84, 2012

[17] Furht B and Escalante A, "Handbook of Cloud Computing", Springer, 2010

[18] Makhaye S, Edger J, Adigun M.O "Towards developing identity management framework to reduce management overhead in cloud infrastructure," presented at the SATNAC.September.2011

Siboniso Comfort Makhaye is an MSc Computer Science student at the University of Zululand in the Department of Computer Science. His research interests include identity management in cloud infrastructure.

Acknowledgements

This work is based on the research supported in part by the National Research Foundation of South Africa -Grant UID: TP11062500001 (2012-2014)