Multi-Tenancy Authorization System with Federated Identity for Cloud-Based Environments Using Shibboleth

Marcos A. P. Leandro, Tiago J. Nascimento, Daniel R. dos Santos, Carla M. Westphall, Carlos B. Westphall
Post Graduation Program in Computer Science (PPGCC)
Federal University of Santa Catarina (UFSC-INE) - Florianópolis, Brazil
{ marcosleandro, tiagojn, danielrs, carlamw, westphal }@inf.ufsc.br

Abstract— **The services provided in clouds may represent an increase in the efficiency and effectiveness in the operations of the enterprise business, improving the cost-effectiveness related to services and resources consumption. However, there is concern about the privacy of data, since such data are outside the client's domain. For these services to be effectively enjoyed by organizations it is necessary to provide access control. The objective of this work is to provide identity management, based on digital identity federation, with authentication and authorization mechanisms for access control in cloud computing environments to independent, trusted third-parties.**

Keywords-cloud computing; identity management; multi-tenancy; federation; Shibboleth; access control; authentication; authorization.

I. INTRODUCTION

Cloud computing enables the use of services and resources on demand. It uses existing technologies such as virtualization, web services, encryption, utility computing and the Internet [1] [2].

The services provided in clouds may represent an increase in the efficiency and effectiveness in the operations of the business enterprise, improving cost-effectiveness in relation to the consumption of resources and services. Cloud computing systems have many superiorities in comparison to those of existing traditional service provisions, such as reduced upfront investment, expected performance, high availability, infinite scalability, tremendous fault-tolerance capability and so on [3]. Enterprises such as Salesforce.com and Google build and offer a cloud service, while many companies and government entities consider building private cloud data centers or integrating cloud services into their infrastructure [4].

However, there is concern about the privacy of data, since such data are outside the domain of the client. That is, on the one hand we have the advantages of the services available, but, on the other hand, there is concern about security. For these services to be effectively enjoyed by organizations is necessary to provide access control.

The success of cloud computing depends on the evolution of the customer mechanisms of Identity and Access Management (IAM) to service providers. IAM plays an important role in controlling and billing user access to the shared resources in the cloud [5]. IAM must evolve for the cloud to become a trusted computing platform [6]. For consumer organizations using the services offered in the cloud it is necessary to implement a safe and reliable IAM model [1] [7] [8].

IAM systems need to be protected by federations, which are groups of organizations that establish trust among themselves to cooperate safely in business. Identities used in this context are called "federated identity". The user can be authenticated in an organization and can use the services of another organization of the federation without the need to repeat the process of authentication (Single Sign-On). Some technologies implement federated identity, such as the SAML (Security Assertion Markup Language) and Shibboleth system [5] [9].

The aim of this paper is to propose a multi-tenancy authorization system using Shibboleth [10] for cloud-based environments. The main idea is to demonstrate how an organization can use Shibboleth to implement in practice a system of access control in a cloud computing environment, without a trusted third-party.

The following sections are organized as follows: Section II describes related work; Section III introduces the basic concepts of cloud computing; Section IV describes the concepts of identity management and presents the architecture and operation of the Shibboleth; Section V presents the proposed multi-tenancy authorization system; Section VI presents the scenario of implementation of the proposed system and how it was implemented; Section VII presents the results and Section X presents the conclusions and future work.

II. RELATED WORK

In [11], an architecture for a new approach to the problem identified as "Mutual Protection for Cloud Computing (MPCC)" is presented. The main concept underlying MPCC is based on the philosophy of Reverse Access Control, where customers control and attempt to enforce the means by which the cloud providers control authorization and authentication within this dynamic environment, and the cloud provider ensures that the customer organization does not violate the security of the overall cloud structure itself. This work only provides a theoretical framework.

In [12], an approach for IDM is proposed, which is independent of Trusted Third Party (TTP) and has the ability to use identity data on untrusted hosts. The approach is based on the use of predicates over encrypted data and multi-party computing for negotiating the use of a cloud service. It uses active bundle—which is a middleware agent that includes PII data, privacy policies, a virtual machine that enforces the policies, and has a set of protection mechanisms to protect itself. An active bundle interacts on behalf of a user to authenticate to the cloud service using the user's privacy policies. A prototype using the technology of Java agents on the JADE environment was developed.

In [13], an entity-centric approach for IDM in the cloud is proposed. The approach is based on: (1) active bundles—similarly to [12]; (2) anonymous identification to mediate interaction between the entity and cloud service by using the entity's privacy policies. Angin et al. [13] proposed the cryptographic mechanisms used in [12] without any kind of implementation or validation.

In comparison with the related work, the infrastructure obtained to provide identity management and access control aims to: (1) be an independent third party, (2) authenticate cloud services using the user's privacy policies, providing minimal information to the SP, (3) ensure mutual protection of both clients and providers. This paper highlights the use of a specific tool, Shibboleth, which provides support to the tasks of authentication, authorization and identity federation. Beyond these objectives, the main contribution of our work is the implementation in cloud and the scenario presented.

III. Cloud Computing

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [14].

In this business model, a customer only pays for services used and as use, without prior commitment, enabling cost reductions in IT deployment and a scalability of far greater resources, which are abstracted to users in order to appear unlimited, and presented through a simple interface that hides the inner workings [15].

From the provider side, the services to be provided are automatically prepared and managed in a multi-tenant model, where a physical server can simultaneously respond to multiple users through virtualization technologies of computing resources.

There are three types of service models that may be offered by cloud computing [14]:

1) Software as a Service (SaaS): providing applications running in the cloud, where the customer has virtually no access control or management of the internal infrastructure;

2) Platform as a Service (PaaS): providing a set of tools that support certain technologies of development and all the necessary environment for deploying applications created by the customer, who is able to control and manage them within the limits of its application;

3) Infrastructure as a Service (IaaS): providing basic computing resources such as processing, storage and network bandwidth where the client can run any operating system or software and maintain as much control as possible.

IV. Identity Management

Digital identity is the "representation of an entity (or group of entities) in the form of one or more elements of information (attributes) that enable the entity to be recognized only within a context" [9].

Identity Management (IdM) is a set of functions and capabilities, such as administration, management and maintenance, discovery, information exchange, policy enforcement and authentication, used to ensure identity information, thus assuring security. An identity management system (IMS) provides tools for managing individual identities in a digital environment [9].

Some specialized features for IMS includes a Single Sign-On (SSO), where a user does not need to be signed on several times to call various applications, and can reuse the authenticated status of a previous application in the same session [16].

An IMS consists of protocols and software components that address the identities of individuals throughout the life cycle of their identities. It involves three main types of entity: the user, the Identity Provider (IdP) and Service Provider (SP). IdPs are responsible for issuing and managing user identities and issue credentials. SPs (also known as relying parties) are entities that provide services to users based on their identities (attributes) [17].

A. Functions of Identity Management Systems

Following are the main functions of an IMS:

- *Provisioning:* the practice of provisioning of identities within an organization addresses the provisioning and deprovisioning of several types of user accounts (e.g. end user, the application administrator, IT administrator, supervisor, developer, etc.) [8].
- *Authentication:* is the process of ensuring that the individual is who he claims to be, and is identified through various mechanisms, such as login, password, biometrics, token, etc. [16].
- *Authorization:* a common need in security is to provide different access levels (e.g. deny/allow) for different parts or operations within a computing system. This need is called authorization [16].
- *Federation:* it is a group of organizations or SPs that establish a circle of trust that allows the sharing of information of user identities to each other [17].

B. Shibboleth

The OASIS SAML standard defines an XML-based framework for describing and exchanging security information between on-line business partners. This security information is expressed in the form of portable SAML assertions that applications working across security domain boundaries can trust. The OASIS SAML standard defines precise syntax and rules for requesting, creating, communicating, and using these SAML assertions [18].

The Shibboleth [10] is an authentication and authorization infrastructure based on SAML that uses the concept of federated identity. With it you can create a safe structure that simplifies the management of identities and provides the user with a SSO for different organizations belonging to the same federation, and who share their identity information in order to do so. The Shibboleth system is divided into two entities: the IdP and SP (Figure 1).

The IdP is the element responsible for authenticating users. It maintains and controls their credentials and attributes, disseminating this information to requests from entrusted organizations. It is composed of four components:

1) Handle Service (HS): authenticates users along with the authentication mechanism and creates a handle token (the SAML assertion that carries the credentials) to the user. Allows an organization to choose the authentication mechanism.

2) Attribute Authority (AA): AA handles requests for SP attributes, applying privacy policies on the release of these attributes (Attribute Release Policies - ARP). Allows the user to specify who can access them. Allows the organization to decide which directory service is used.

3) Directory Service: (external to Shibboleth) local storage of user attributes.

4) Authentication Mechanism: (external to Shibboleth) allows users to authenticate with the central service with only a login/password pair.

The SP Shibboleth is where the resources are stored, that are accessed by the user. It enforces access control on resources based on information sent by the IdP. A single SP may be composed of several applications, but will still be treated as a single entity by an IdP. It has three main components:

1) Assertion Consumer Service (ACS): responsible for receiving messages (SAML) to establish a secure environment.

2) Attribute Requester (AR): responsible for obtaining and passing user attributes to RM.

3) Resource Manager (RM): intercepts requests for resources and makes decisions to control access based on user attributes.

The WAYF ("Where Are You From", also called the Discovery Service) is an optional feature on the Shibboleth system, responsible for allowing an association between a user and organization. When trying to access a resource, the user is forwarded to an interface that asks you to choose the institution to which it belongs. After choosing the institution, the user is redirected to start the authentication process. The WAYF service can be distributed as part of a SP or as part of the third code operated by a federation. In cases where it is used with SPs offering resources for registered users in several IdPs it becomes quite useful.

The flow of operation of Shibboleth is represented in Figure 1.

In Step 1, the user navigates to the SP to access a protected resource. In Steps 2 and 3, Shibboleth redirects the user to the WAYF page, where he should inform his IdP. In Step 4, the user enters his IdP, and Step 5 redirects the user to the site, which is the component HS of the IdP. In Steps 6 and 7, the user enters his authentication data and in Step 8 the HS authenticate the user. The HS creates a handle to identify the user and sends it also to the AA. Step 9 sends that user authentication handle to AA and to ACS. The handle is checked by the ACS and transferred to the AR, and in Step 10 a session is established. In Step 11 the AR uses the handle to request user attributes to the IdP. Step 12 checks whether the IdP can release the attributes and in Step 13 the AA responds with the attribute values. In Step 14 the SP receives the attributes and passes them to the RM, which loads the resource in Step 15 to present to the user.

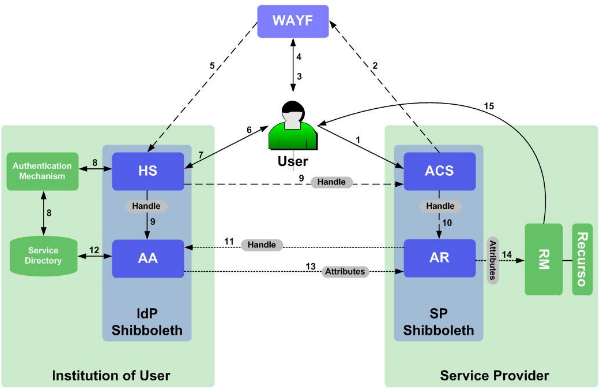

Figure 1. Operation of Shibboleth.

V. Federated Multi-Tenancy Authorization System on Cloud

According to [5], in order to ensure access control in open environments such as cloud computing, IdM can be implemented in several different types of configuration Figure 2. Firstly, IdM can be implemented in-house. In this configuration, identities are issued and managed by the user companies. Also, IdM itself can be delivered as an outsourced service, which other companies and consumers use. This is called Identity as a Service (IDaaS). There are several commercial offerings in the market. In this configuration, identities are issued and managed by user companies and/or IDaaS providers. In a “managed” hosting case, an IDaaS provider maintains a complete set of employee data that a user company outsources. In other cases, IDaaS providers only maintain pseudonyms of employees, which user companies map to real employee identities. Lastly, each cloud SP may independently implement a set of IdM functions. This configuration requires user companies to maintain a different set of identities for each of the relying parties.

In this work, it was decided to use the first case configuration (in-house), where the client company has complete control and responsibility for the digital identities of its users.

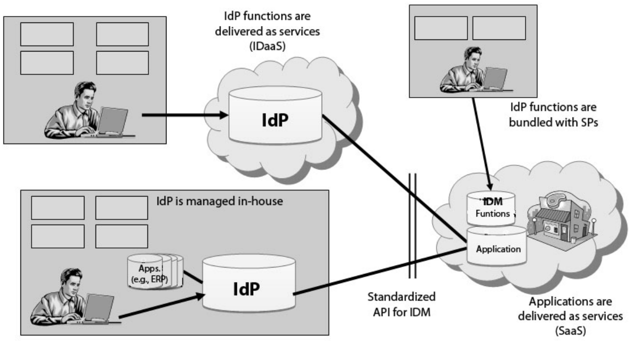
IdP functions are delivered as services (IDaaS)
IdP
IdP functions are bundled with SPs
IdP is managed in-house
Apps. (e.g., ERP)
IdP
IDM Funtions
Application
Standardized API for IDM
Applications are delivered as services (SaaS)

Figure 2. Configurations of IDM systems on cloud computing environments [5].

This work presents an authorization mechanism to be used by an academic institution to offer and use the services offered in the cloud.

The part of the management system responsible for the authentication of identity will be located in the client organization, and communication with the SP in the cloud (Cloud Service Provider, CSP) will be made through identity federation. By establishing trust between the parties, the CSP will request the authentication of users to the IdP located in the client. Thus, the user's data remain under the care of his own company, enhancing privacy and preventing loss of information.

The access system performs authorization or access control in the environment. The CSP should be able to interpret and separately allow access in accordance with the privileges of each user. The institution has a responsibility to provide the user attributes for the deployed application SP in the cloud.

The authorization system should be able to accept multiple clients, such as a multi-tenancy. The concept of multi-tenancy [19] states that an application is used equally across a series of users, each receiving comparable or equitable levels of responsiveness and bandwidth through the use of the Tenant Load Balancer.

VI. SCENARIO

The setting is an academic federation sharing services in the cloud (Figure 3).

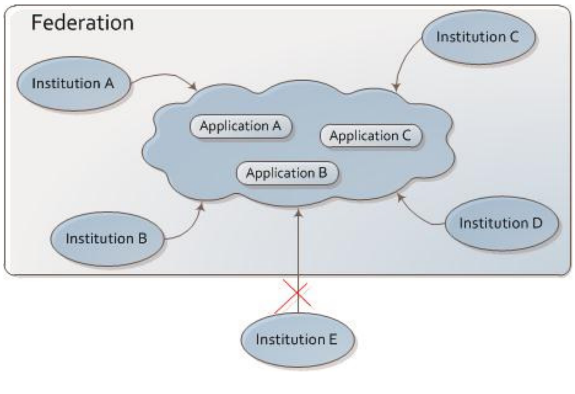

Figure 3. Academic Federation sharing services in the cloud.

A service is provided by an academic institution in a CSP, and shared with other institutions. In order to share services is necessary that an institution is affiliated to the federation.

For an institution to join the federation it must have configured an IdP that meets the requirements imposed by the federation. Since these requirements are expressed in the form of access and privacy policies defined by SAML.

Once affiliated with the federation, the institution will be able to authenticate its own users, according to the authentication and authorization system described in the previous session, since authorization is the responsibility of the SP.

A. Implementation of the Proposed Scenario

For testing and demonstration, a SP was primarily implemented in the cloud. Resulting in the deployment of an Apache server on a virtual machine hired by the Amazon Web Services cloud provider—as illustrated in Figure 4. In this server, beyond the installation of the Shibboleth SP, an application was chosen to serve as an example of the resource to be offered as a service: the software development and collaborative editing of documents DokuWiki [20]. The concept of a lazy session was used to allow users to access the wiki anonymously for reading, and only having to authenticate when permission was needed to edit documents.

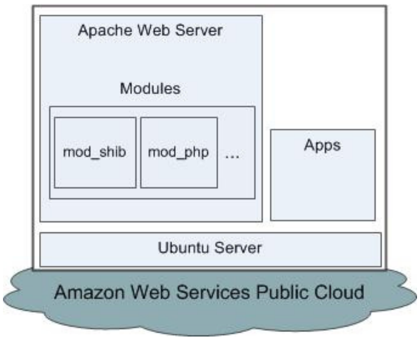

Figure 4. Cloud Service Provider Diagram.

Authorization within the Shibboleth SP can be accomplished in three ways:

- via directives in the .htaccess Apache file, where instructions "require" may include specific users, groups, etc.;
- via the <AccessControl> element that provides several possibilities for more complex use cases of access control [10];
- via the application, which is free to create internal rules according to the attributes available.

The SP was configured with authorization via application, to differentiate between common users and administrators of Dokuwiki.

Before releasing access to users, it was necessary to specify which attributes, among those released by the IdP, the application would be using and how they would be used. This step is application specific, and Figure 5 shows the contents of the file /etc/dokuwiki/local.php, which combines the attributes of the IDP "Shib-inetOrgPerson-cn " and "Shib-eduPersonPrincipalName" to the attributes of the application "var_remote_user" and "var_name", respectively. Other combinations are also possible.

```
$conf['auth']['shib']['var_remote_user'] =
'Shib-inetOrgPerson-cn';
$conf['auth']['shib']['var_name'] = 'Shib-
eduPerson-eduPersonPrincipalName';
```

Figure 5. Contents of the file /etc/dokuwiki/local.php

Later, a cloud IdP was installed (Figure 6), only to illustrate that each institution has its own IdP control, without regard to whether it is local or cloud.

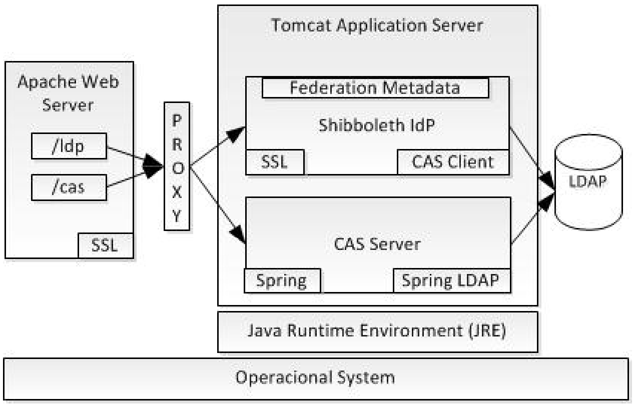

Figure 6. IdP detailed diagram.

The authentication mechanism is external to Shibboleth, and for this purpose we used the JASIG CAS Server [21] that performs user authentication through login and password and provides SSO via a web interface, and then passes the authenticated users to Shibboleth. The CAS has been configured to search for users in a Lightweight Directory Access Protocol (LDAP). To use this directory OpenLDAP [22] was installed in another virtual machine, also running on Amazon's cloud.

To demonstrate the use of SP for more than one client, another IdP was implemented, also in cloud, similar to the first. From this point, the concept of multi-tenancy is necessary, since the service provided by SP will be shared by multiple clients. To support this task Shibboleth provides a WAYF component, which is responsible for allowing the association between a user and an organization. This mechanism was set up by the SP to manage the institutions belonging to the federation.

VII. Use Cases: Analysis and Test Results within Scenario

The result of the deployment of IdPs and SPs is shown in Figure 7.

In this resulting structure, each IdP is represented in a private cloud, and the SP is in a public cloud.

Once the scenario was implemented, some tests were performed. The results highlighted two main use cases:

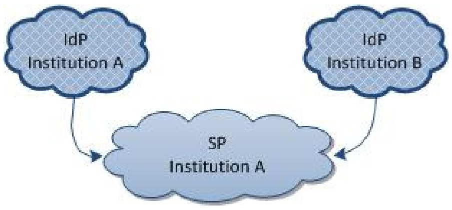

Figure 7. Resulting deployment scenario

A. Read access to documents

In the case of read-only access, the service offered allows anonymous use. To perform this type of access the user simply types the URL of the desired service in the Web Browser. In this case, the service is also available for external access to the federation.

B. Access for editing documents

In order for a user to have permission to alter documents, authentication is required. In this case, the user will require authentication through the "Login" button that directs them to a URL protected by Shibboleth. The Mod_shib process verifies that the user does not have an open session and forwards it to the Discovery Service of the Federation (WAYF), where the user chooses their institution and is forwarded to the URL of the IdP of the institution chosen, by an HTTP redirect. On the Web page of the IdP, the user enters their credentials (username and password) and if successful, cookies are registered and a handle—a SAML assertion with data from the Authentication—is created.

This handle is used by the SP to request user attributes from the IdP, which analyzes the request based on previously established rules of release. If the handle is valid and the request is accepted, another cookie is created and the user is finally redirected to the SP (the Web application that was originally accessed), and their attributes are sent by the Shibboleth module to this application as values of the environment variables, so that it can use them any way it chooses. With these attributes, the application uses internal authorization rules to determine whether the user has administrative rights on the system.

VIII. Conclusions and Future Work

The adoption of secure IdM in a cloud environment addresses the issues of identity provisioning, authentication, authorization and federation. The use of federations in IdM plays a vital role in enabling organizations to authenticate their users cloud services using any chosen IdP [7].

The work of Albeshri and Caelli [11] only provides a theoretical framework. Ranchal et al. [12] developed a means to protect privacy in a cloud environment and a prototype using the technology of Java agents on the JADE environment. An active bundle was used, that is a container with a payload of sensitive data, metadata, and a virtual machine. Angin et al. [13] proposed the cryptographic mechanisms used in [12] without any kind of implementation or validation.

The focus of this work was aimed at an alternative solution to a IDaaS, which is a solution where the activities concern outsourced IdM, and therefore, the data and sensitive information of users are outside the domain of the organization, since they are controlled and maintained by a third party.

The infrastructure obtained to provide identity management and access control aims to: (1) be an independent third party, (2) authenticate cloud services using the user's privacy policies, providing minimal information to the SP, (3) ensure mutual protection of both clients and providers. This paper highlights the use of a specific tool, Shibboleth, which provides support to the tasks of authentication, authorization and identity federation.

Shibboleth was very flexible with regards to its use in a cloud environment, allowing a service to be provided reliably and securely. In addition, Shibboleth is based on SAML, which means it is compatible with international standards, thus ensuring interoperability.

With the settings applied to the scenario, it became possible to offer a service allowing public access in the case of read-only access, while at the same time requiring credentials where the user must be logged in order to change documents.

As future work, we propose an alternative authorization method, where the user, once authenticated, carries the access policy, and the SP should be able to interpret these rules. Thus, the authorization process will no longer be performed at the application level.

We also suggest expanding the scenario to represent new forms of communication, and thus create new use cases for testing. For example, (i) provide a service deployed on a new SP where this service is provided by another institution; (ii) provide more than one service in the same SP.

A further example would be to use pseudonyms in the CSP domain, which should ensure the nature of the individual user without the need to expose their real information.

REFERENCES

[1] B. Grobauer, T. Walloschek, and E. Stocker, “Understanding Cloud Computing Vulnerabilities,” IEEE Secur. Priv., vol. 9, no. 2, pp. 50–57, Mar.-Apr. 2011, doi: 10.1109/MSP.2010.115.

[2] F. Maggi, and S. Zanero, “Is the Future Web more Insecure? Distractions and Solutions of New-Old Security Issues and Measures,” Proc. 2nd Worldwide Cybersecurity Summit (WCS 11), 1-2 June 2011, pp. 1–9.

[3] M. Zhou, R. Zhang, D. Zeng, and W. Qian, “Services in the Cloud Computing Era: A survey,” 4th Intl. Univ. Communication Symposium (IUCS 10), pp. 40–46, doi: 10.1109/IUCS.2010.5666772.

[4] “Identity Federation in a Hybrid Cloud Computing Environment Solution Guide,” JUNIPER Networks, accessed in Oct. 2011. Online at: http://www.juniper.fr/us/en/local/pdf/implementation-guides/8010035-en.pdf.

[5] E. Bertino, and K. Takahashi, Identity Management - Concepts, Technologies, and Systems. ARTECH HOUSE, 2011.

[6] E. Olden, "Architecting a Cloud-Scale Identity Fabric," Computer, vol. 44, no. 3, Mar. 2011, pp. 52–59, doi: 10.1109/MC.2011.60.

[7] “Security Guidance for Critical Areas of Focus in Cloud Computing,” CSA, accessed in May 2011. Online at: http://www.cloudsecurityalliance.org.

[8] “Domain 12: Guidance for Identity and Access Management V2.1.,” Cloud Security Alliance. - CSA, accessed in Sep. 2011. Online at: https://cloudsecurityalliance.org/guidance/csaguide-dom12-v2.10.pdf.

[9] D. W. Chadwick, Federated identity management. Foundations of Security Analysis and Design V, Springer-Verlag: Berlin, Heidelberg 2009 pp. 96–120, doi: 10.1007/978-3-642-03829-7_3.

[10] “Shibboleth 2 Wiki,” SHIBBOLETH, accessed in Sep. 2011. Online at: https://wiki.shibboleth.net/confluence/display/SHIB2/Home.

[11] A. Albeshri, and W. Caelli, “Mutual Protection in a Cloud Computing environment,” Proc. 12th IEEE Intl. Conf. on High Performance Computing and Communications (HPCC 10), pp. 641-646, doi:10.1109/HPCC.2010.87.

[12] R. Ranchal, B. Bhargava, A. Kim, M. Kang, L. B. Othmane, L. Lilien, and M. Linderman, “Protection of Identity Information in Cloud Computing without Trusted Third Party,” Proc. 29th IEEE Intl. Symp. on Reliable Distributed Systems (SRDS 10), pp. 368–372, doi: 10.1109/SRDS.2010.57.

[13] P. Angin, B. Bhargava, R. Ranchal, N. Singh, L. B. Othmane, L. Lilien, and M. Linderman, “An Entity-Centric Approach for Privacy and Identity Management in Cloud Computing,” Proc. 29th IEEE Intl. Symp. on Reliable Distributed Systems (SRDS 10), pp. 177–183, doi:10.1109/SRDS.2010.28.

[14] “Definition of Cloud Computing,” NIST, accessed in May 2011. Online at: http://csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_cloud-definition.pdf

[15] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz et al., “A View of Cloud Computing,” Communications of the ACM. Association for Computing Machinery, vol. 53, no. 4, 2010. p. 50–58.

[16] A. Belapurkar, A. Chakrabarti, H. Ponnapalli, N. Varadarajan, S. Padmanabhuni, and S. Sundarrajan, Distributed Systems Security: Issues, Processes and Solutions, John Wiley & Sons, 2009.

[17] A. Bhargav-Spantzel, J. Camenisch, T. Gross, and D. Sommer, “User Centricity: A Taxonomy and Open Issues,” Journal of Computer Security - The Second ACM Workshop on Digital Identity Management - DIM 2006, vol. 15, iss. 5, IOS Press Amsterdam, October 2007, pp. 493–527.

[18] “Security Assertion Markup Language (SAML) V2.0 Technical Overview,” OASIS, accessed in Sep. 2011. Online at: http://www.oasis-open.org/committees/download.php/27819/sstc-saml-tech-overview-2.0-cd-02.pdf.

[19] R. PalsonKennedy, and T. V. Gopal, “Assessing the Risks and Opportunities of Cloud Computing – Defining Identity Management Systems and Maturity Models,” Proc. Trendz in Information Sciences & Computing (TISC 10), pp. 138–142, doi: 10.1109/TISC.2010.5714625.

[20] “Dokuwiki Features,” DOKUWIKI, accessed in Sep. 2011. Online at: http://www.dokuwiki.org/dokuwiki.

[21] “JASIG CAS”, JASIG, accessed in Sep. 2011. Online at: http://www.jasig.org/cas.

[22] “OpenLDAP Foundation,” OpenLDAP, accessed in Sep. 2011. Online at: http://www.openldap.org.