Requirements and Initial Design of a Grid Pseudonymity System

Joni Hahkala, Henri Mikkonen, Mika Silander, and John White

Abstract—**Traditionally, grid users have been identifiable and traceable beyond reasonable doubt by their digital certificates. However, Grids are used in an ever-increasing variety of contexts and thus, the number of usage scenarios has augmented accordingly. In bio-medicine and other health-related fields a need for anonymous access to grid resources has been identified. Anonymous access to resources prevents the resource owners and other external parties from tracing the users and their actions. Such anonymity of resource usage in Grids is needed above all in commercial contexts, e.g. protecting the development process of a new medicine by anonymizing the accesses to medical research data bases. In this paper we identify the requirements and give an initial design for pseudonymity system addressing these needs.**

Index Terms—**Authentication, Authorization, Grid Security, Pseudonymity.**

I. Introduction

The Grid computing model envisages a heterogeneous fabric of computing resources that is provided to users in a transparent way. In this model, Grid users may run processes on computing resources and store and access data on storage resources that may not be owned by them or even their parent organization. The use of resources on any Grid infrastructure entails a balance between the owner's need to oversee and account for the resource usage and the user's privacy requirements.

From the Grid users' point of view, complete anonymity is desirable for maximum protection. This requirement can come from researchers in a field of competitive, commercial or basic, research [1], [2]. These researchers may wish to work in secrecy and prevent their competitors from following their actions on a Grid. This would include being able to anonymize the credentials used for job submissions and the reading and writing of data. In general, this is not possible due to requirements that a Grid user should be traceable for accounting purposes and in the case of usage policy violation.

Hence, the anonymity problem is to find a compromise between the requirements of the Grid resource owner and users. The proposed solution to this problem is the concept of a lesser degree of anonymity, *pseudonymity*. A pseudonymous identity, or *pseudonym* for short, is a unique anonymous identity given by a trusted third-party (service) to a Grid user. Only this trusted third party is able to re-establish this identity association later if necessary. In situations where resource owners detect misuse of their resources, the trusted third-party can act as an middle man to solve grievances or in serious cases can be requested to disclose the true identity of the suspected abuser, subject to the policies of the particular Grid infrastructure or the law.

The system presented in this paper is designed to hide the identity of the user invoking the operations on the Grid. If used properly and provided there is a sufficiently broad mix of operations and end users, the system will also prevent the correlation of operations and thus ensure the resource owners cannot identify users by workflow tracking.

The rest of the paper is organised as follows: Chapter II looks into work generally related to pseudonymity. Chapter III offers a detailed study of the requirements. Chapter IV analyses the problem in light of the requirements and constraints. Chapter V discusses some architectural and functional issues as Chapter VI describes various solutions to the problem and specifies the one chosen. We summarize our findings in Chapter VII and propose future work in Chapter VIII.

II. Related work

Pseudonymity and pseudonym identifiers have already been covered by several specifications and software. Their definitions and interpretations are discussed in this section.

A. Shibboleth and SAML

Shibboleth[1] is Internet2's project to provide Single Sign-On (SSO) on the Web. The current version (1.3) bases on Security Assertion Markup Language (SAML) 1.1 [3] specifications, but the upcoming Shibboleth 2.0 will support major portions of SAML 2.0 [4] Both the current version of Shibboleth and the SAML 2.0 standard support short-lived opaque name identifiers. A typical use case starts at a Service Provider (SP) which wants to know some attributes of the user. Instead of authenticating the user directly, the SP redirects the user to the Identity Provider (IDP) for authentication. Once authenticated and authorized, the IDP generates an opaque name identifier for the user and communicates it to the requesting SP. The name identifier is then utilized by the SP for obtaining user attributes from the IDP.

Manuscript received February 20th, 2008. This work was supported in part by the European Union grant EGEE-II INFSO-RI 031688 and by the Joint Committee of the Nordic Research Councils for Natural Sciences (NOS-N) through its Nordunet-3 programme project "Innovative services and tools for Nordugrid".

Joni Hahkala, Henri Mikkonen and John White are with Helsinki Institute of Physics, CERN/PH, CH-1211 Geneva 23, Switzerland (phone: +41-22-76-76179; fax: +41-22-76-73600; e-mail: firstname.surname@cern.ch).

Mika Silander is with Helsinki Institute of Physics, PL 9250, 02015 TKK, Finland (phone: +358-9-8562-0909; fax: +358-9-451-5194; e-mail: mika.silander@hip.fi).

[1] Shibboleth site - http://shibboleth.internet2.edu

Proceedings of the 2008 High Performance
Computing & Simulation Conference ©ECMS
Waleed W. Smari (Ed.)
ISBN: 978-0-9553018-7-2 / ISBN: 978-0-9553018-6-5 (CD)

As the name identifier is opaque and short-lived, the SP cannot determine any additional user information apart from the attributes that are provided by the IDP. The attributes may include only the Virtual Organization (VO) membership information that is enough for the SP to authorize but not to individualize the user.

The integration of Shibboleth's SAML attribute framework and Grid security has been studied by the GridShib project [5]. The goals of the project include e.g. utilization of the Shibboleth attributes in the user authorization process, but also pseudonymous access for the Grid users [6]: Shibboleth's opaque name identifiers are used in the subject fields of the X.509 certificates which are issued to the users online after Shibboleth authentication. The users' Shibboleth attributes can be utilized with these pseudonym certificates too.

B. idemix

The approach described in the previous section is very IDP-centric as the IDP keeps track of its users' accesses to the SPs. Anonymous credential systems allow the user's transactions to be carried out in a way that they cannot be linked to the same user [7]. A software called idemix (identity mix) [8] is one example implementation of such a system. The users establish pseudonyms with SPs that are used for creating credentials containing a set of attributes. Afterwards, the users present the credentials with desired sets of attributes to the same or other SPs by using zero-knowledge proofs. These proofs ensure the legitimate possession of the credentials but reveal no information about the true identity of the user employing them. The credentials can be used for obtaining new credentials, but only one master secret is related to all of them. Mechanisms for retrieving the identity of a user locally (one SP) or globally (all the SPs) exist, but they require the user's cooperation.

C. WS- specifications*

As an alternative to SAML, WS-* (Web Services) specifications also provide a model for federation between IDPs and SPs. From the set of specifications, WS-Federation defines a Pseudonym Service which maintains alternate identity information for its users [9]. The pseudonym identifiers are part of the security tokens that are used by resources' Security Token Services (STS) for authentication and authorization purposes. In addition to the Shibboleth-style short-lived (or one-time) opaque pseudonym identifiers, anything between them and constant clear-text identifiers are supported. As the communication with the service itself can occur via IDPs, the resources' STSes, or directly with the resource or the requestor, numerous use cases are supported. A Pseudonym Service and the claims-based authorization model can be used to describe the set of attributes required to access a resource and the IDP can assert that a particular Grid user possesses those attributes, without divulging their actual identity.

III. Requirements

In order for a pseudonymity system to function with current Grid middleware it should fulfill the following requirements.

Requirement 1. ***Confidentiality***
The pseudonymous identity must hide the true identity of the user.

The true identity must remain unknown to the service provider sites, their administrators and other legitimate Grid users as well as external parties. This implies encrypted communications is needed between pseudonym attestee and attester.

Requirement 2. ***Non-repudiation/Retrievability***
The true identity of a Grid user must be, a posteriori, unambiguously traceable via the pseudonymity system.

This functionality is mandatory in cases of misuse and may be imposed by regulatory or law enforcement issues. To this end, the pseudonymity attester must authenticate pseudonym requestors and maintain a record of the pseudonyms issued.

Requirement 3. ***Uniqueness and Short Life-time***
A pseudonym must be a unique, short-lived one-time identity in the Grids in which it is to be employed.

A pseudonym's Distinguished Name (DN) must not clash with the existing user DNs, nor with other pseudonyms as this would undermine the overall user authentication and violate the earlier requirement of retrievability. And ideally, only one Grid operation or set of operations should be performed under the protection of a pseudonym. For the next operations, a new pseudonym should be requested. This approach reduces the ability of outside observers to collect data for correlation attacks with the intent of discovering the true identity of the user. Pseudonymous credentials should be ephemeral to reduce the damage in cases of credential compromise. Due to ephemerity and large volume of issued credentials, the issuance itself should involve no manual intervention nor procedures.

Assuming there are several independent pseudonym attesters active in a Grid, each must be assigned an own unique name space. This name space prevents attesters from accidentally issuing pseudonyms with identical DNs.

Requirement 4. ***Identity Protection***
The pseudonymity attester must be the only party able to obtain the true identities of users.

The pseudonymity attester must adequately protect the records of issued credentials and the systems into which they are stored. Only authorized people are entitled to uncover the true user identities.

Requirement 5. ***Credential Source Compatibility***
The pseudonymity system should interoperate with different sources of Grid user credentials.

Even though the user authentication is based on credentials, they may not necessarily come directly from the user's client software. The user's short- or long-term credentials can be stored in online credential repositories or be delegated to other Grid services acting on a user's behalf. For example, some portal usage scenarios involve the delegation of the user's proxy certificate [10] directly to the portal with no user intervention. Hence, the pseudonym system must support a broader set of use cases, not only those implied by direct user access.

Requirement 6. ***Information leakage prevention***
The pseudonymity system must actively counteract the leakage of information that allows the unique identification of a pseudonym user.

The operations and actions a pseudonym user performs and the set of additional personal attributes the user may have requested for inclusion into the pseudonym credentials, may provide enough information to uniquely identify the user. The pseudonymity system should therefore attempt to actively reduce and hide sources of such information. In the case of personal attributes from auxiliary authorization systems, the pseudonymity system should either prevent uniquely identifying pseudonyms to be issued, or, warn the user about the high probability of disclosure prior to using such a credential. Other covert sources of information, e.g. IP numbers of job submission hosts, metadata in submitted files and job description language attributes are harder to deal with and their removal or anonymization is ultimately up to the users themselves.

Requirement 7. ***Maintaining security***
The pseudonymity system must not provide ways to circumvent existing security.

The pseudonymity must not erode the security of the systems. There might be cases where enforcing a detailed policy would need the user's identity to be revealed and in these cases the policies can't be enforced at the time. Later, the authorized persons can do the enforcement of these policies and the corresponding actions if needed.

IV. Problem Analysis

Many of today's Grid middleware systems authenticate users with PKI certificates. Thus, in addition to the requirements presented in the previous section, we impose on ourselves an implementation constraint, that of compatibility: *The pseudonymity certification must be compatible with the certificate-based authentication and interoperate seamlessly with existing Grid middleware*. This also means the pseudonymity certification must work in concert with other commonly used auxiliary authorization systems such as Virtual Organization Membership Service (VOMS [11]) and Community Authorization Service (CAS [12]) without any significant changes to them.

Existing Grid user certificates and software can be employed with little effort to ensure that pseudonym requesters are unambiguously authenticated with cryptographically strong mechanisms. For the same reason, SSL/TLS channels can easily be set up to guarantee the confidentiality of communications. These fulfill the requirements 1 and 2 and comply with the above implementation constraint.

The pseudonym credential itself can be modeled as a standard X.509 [13] user certificate but having an anonymizing DN. The set of resources available to a Grid user acting under a pseudonym will be more limited than if the user had employed their ordinary user certificate. This is due to the fact that authentication and authorization decisions must be based solely on auxiliary attributes provided by auxiliary authorization systems, the exact user identity being unavailable. Thus, users should be able to request some of their real identities' attributes to be included in the pseudonyms and this implies the pseudonymity system needs to interact directly with auxiliary authorization services. The VOMS auxiliary authorization service models the user attributes as Attribute Certificates [14] (AC). It is commonplace to include these into the extensions of X.509 certificates and this is a further motivation to use X.509 certificates as the format for pseudonym credentials.

Certificate Authorities (CAs) may freely issue certificates unconstrained by any name space. In Grids however, the uniqueness of certificates is guaranteed by reserving specific name spaces for each CA. Only those certificates issued in conformance with the name space restriction are accepted as valid credentials in a Grid. The uniqueness of pseudonym credentials implied by requirement 3, can be ensured similarly by assigning unique name spaces to the pseudonym attesters.

Requirement 3 also states the pseudonym credentials need to be short-lived which implies a high volume of credentials to be issued. Hence they should be generated programmatically. The pseudonym system must therefore incorporate functionality similar to online certificate authority (online CA) services, e.g. EJBCA [15].

Requirement 4 has two implications: firstly, the pseudonym credential must not contain any information as to the identity of its requester, secondly, the internal security procedures and measures of the pseudonym attester must ensure the access to the records is strictly limited to authorized personnel.

Requirement 5 describes the different types of sources from which pseudonym credential requests may originate. Pseudonym certificates requested by the user, either with the help of their long-term user certificates or a proxy certificate generated from the former, will leverage the existing X.509 authentication as is. In the course of using Grid resources, the user may delegate their rights to further components acting on their behalf such as the credential repository service, MyProxy [16]. These, in turn, may delegate the credentials further to Grid portals and hence, pseudonymity requests from portals need to be handled. In order to increase entropy and thus hinder statistical correlation attacks (req. 6), a portal should request new unique pseudonym credentials for each job launched. Portals will delegate the pseudonym user's rights further to the point where the job reaches a Computing Element(CE) [17]. The CE is responsible for collecting the resources defined by the job description and selects a computing node for the execution of the job. The collection is done with the permissions of a limited proxy. The CE may also decide to request new pseudonym credentials for each resource access needed for the staging of the job, thus, the pseudonymity system must accept requests from CEs as well.

Requirement 6 is the most difficult to address since there are many sources that indirectly provide more information concerning the pseudonym user's identity. The pseudonym attester may however, guarantee that the additional personal attributes the user wishes to include in the pseudonym credential, e.g. role, group membership information, capabilities etc, will not uniquely identify the user. This requires modification of auxiliary authorization services since these must provide the pseudonymity system information about whether the requested user attribute combination is uniquely identifying or not.

According to the requirement 7, adding a pseudonymity system into the overall security infrastructure must not weaken security nor introduce new security holes. The ability of the pseudonymity system to circumvent purely identity based limitations like blacklists is at first sight one such hole. However, an abuser of pseudonyms will be detected equally and can be deprived the usage of pseudonyms. Other limitations on the user's credentials like a proxy certificate limitation, must prevail.

The Functional Requirements (FR#) and software components that are minimally needed to implement a working pseudonymity system are summarized below:

FR1 The pseudonymity system should authenticate all requests relying on existing Grid security mechanisms, i.e. SSL/TLS communication and X.509 certificates.

FR2 The communications in all interactions should be protected with authenticated and encrypted SSL/TLS channels.

FR3 Pseudonym credentials should be modeled as X.509 certificates.

FR4 Additional individual attributes (role, group etc) should be modeled as Attribute Certificates.

FR5 Pseudonym credential requests should be honoured to entities authenticating with user long-term X.509 certificates and proxy X.509 certificates.

FR6 The credential issuance of the pseudonymity system must not include manual operations, in other words, it should operate in the same manner as an online CA.

FR7 Auxiliary authorization services must offer functionality that allows the pseudonymity system to judge whether additional user attributes to be included in the pseudonym credential identify the user uniquely.

FR8 A pseudonym credential requested with a credential having rights limitations, must, if granted, return a pseudonym credential with identical limitations. A limited proxy is an example of such a credential.

FR9 The pseudonymity system must not create ways to circumvent the security of the system.

V. Discussion

In this section we discuss some architectural and functional issues of the pseudonymity system before describing the alternative solutions in the next section.

A. *On the architecture of the pseudonymity system components*

We outlay our solution alternatives using three independent components: the pseudonymity service, the online CA service and an Attribute Authority. Having this separation allows us to benefit from existing and well-tested attribute authority and online CA software. Also, we avoid reimplementing their functionality within the pseudonymity service. In this setting, the pseudonymity service acts as a Registration Authority authenticating the users and validating their requests before forwarding the requests to the online CA. The pseudonymity service fulfills the traceability requirement 2 by maintaining records of the pseudonyms issued to the authenticated users. Having an automated online CA ensures the timely delivery of pseudonym credentials in accordance with FR6.

Due to the short-life time of pseudonymous credentials, we anticipate that certificate revocation functionality is not vital. However, such functionality can be added later non-intrusively if deemed necessary. This is similar to the fact that Certificate Revocation Lists (CRLs) are not used against individual proxy credentials, but against the underlying long-lived ones.

It is likely that a virtual organization offering Attribute Authority services will also provide a pseudonymity service for its user community. Therefore, even though this division into separate components apparently violates requirement 4, we consider this an insignificant relaxation of constraints.

B. *On generating the pseudonymous identities*

In principle, the pseudonymous identifiers could be requested by the user, generated by the pseudonymity system, or, possibly even the online CA in some cases, depending on the implementation. This process could also include some modifications to the request for supporting different protocols and message formats in the communication with the online CA by one single message schema between the client and the pseudonymity system. In this paper we limit ourselves into stating that all of the above options can be made sufficiently secure for the purposes of pseudonymity and postpone the final choice.

After this first stage the Grid user has a pseudonymous credential with a random DN. This credential does not necessarily have the user's attributes attached as these are granted by their Attribute Authorities. Therefore, in order for the users to gain access to additional resources enabled by their attributes, the attributes have to be retrieved on a second stage from a trusted Attribute Authority that has the knowledge of who possesses the pseudonymous identity. Variations to this two-step creation of a pseudonymity credential are described in further detail in the next section.

C. *On required modifications to Attribute Authority components*

In all the scenarios described in the next section, the internal data models of Attribute Authorities need to be extended to associate pseudonyms as aliases of real users. Upon reception of an attribute request related to a pseudonym, the Attribute Authorities should return information on the degree of uniqueness of the user attributes as stated in FR7. This is not normal feature of Attribute Authorities and thus implies slight changes to the ones supported. The threshold degree and what is done when this limit is reached need be configurable on a pseudonym service basis. Ultimately it is however the task of the VOs to attempt to ensure the user groups remain sufficiently large to prevent this from occurring. Also the Attribute Authority must have a way of cleaning up the old expired pseudonyms so that the pseudonymity list doesn't become unmaintainably large over time.

VI. Solutions

We discuss the pros, cons and differences of three alternative architectures. The last one which we propose for implementation, is in our understanding the most advantageous architecture.

A. First scenario: All-in-one pseudonymity service

In the first alternative fulfilling the requirements of chapters III and IV, the Grid user communicates only with the pseudonymity service to acquire a complete pseudonym credential including the desired set of auxiliary user attributes. First, as shown in Fig. 1, the Grid user requests a pseudonymous credential from the pseudonymity service. Next, the pseudonymity service contacts the user's Attribute Authority and the user attributes are added to the pseudonym credential request. This request is then passed on to the online CA for signing. Once signed, the pseudonymity service returns the now valid credential back to the user.

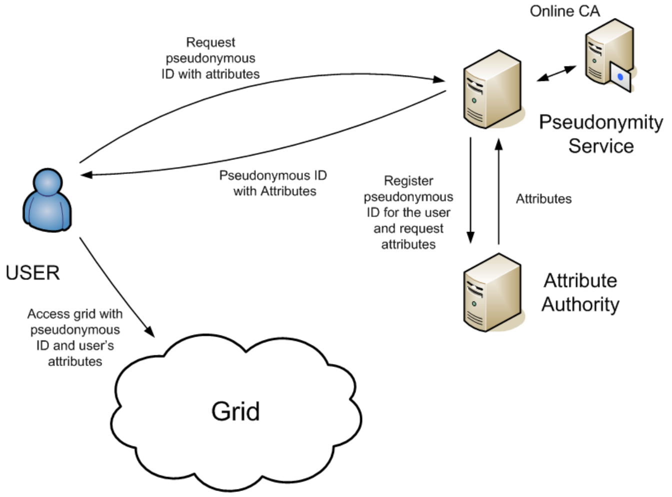
Online CA
Request pseudonymous ID with attributes
Pseudonymity Service
Pseudonymous ID with Attributes
Register pseudonymous ID for the user and request attributes
Attributes
USER
Attribute Authority
Access grid with pseudonymous ID and user's attributes
Grid

Fig. 1. All-in-one pseudonymity service. The pseudonymity service contacts both the Attribute Authority and the online CA.

This scheme has some disadvantages in that the pseudonymity service will have to handle attribute requests from the Grid user. If the attribute requests are not handled by the pseudonymity service, then the Attribute Authority will have to return all attributes with the pseudonymous credential for that particular user each time. Another complication is that the pseudonymity service needs to implement all the APIs, communication and error handling of the Attribute Authorities that need be supported. If any of these change, the pseudonymity service has to be changed accordingly.

An advantage of this scheme is that the pseudonymity service has the possibility to decline to issue a pseudonymous identity altogether if the requested user attributes are uniquely identifying. The pseudonymity service would basically replace the Attribute Authority so when the user wants a short lived proxy, he vould use the pseudonymity service instead of Attribute Authority resulting in pseudo proxy instead of normal proxy.

B. Second scenario: Pseudonymous identities with user-driven identity registration

By making the users themselves register their pseudonymous identities to the Attribute Authorities as shown in Fig. 2, we eliminate the need to support this registration in the pseudonymity service. In addition, the users will request their auxiliary attributes directly from the Attribute Authorities exactly as in current Grid middleware. This latter point removes the burden of supporting the different APIs of Attribute Authorities in the pseudonymity service. Both features simplify the API and the internal architecture of the pseudonymity service.

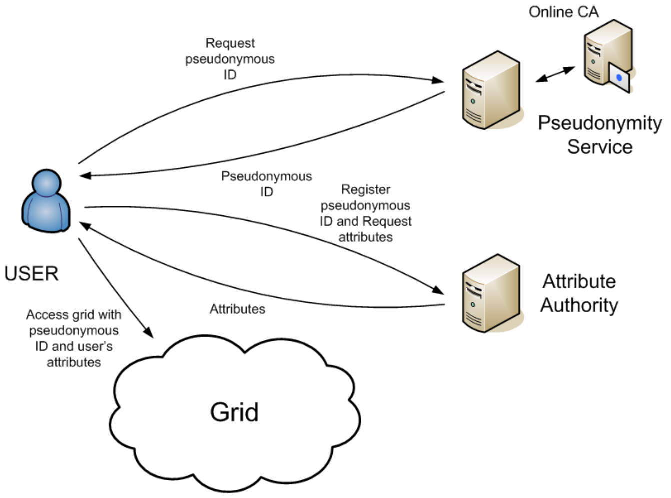

Fig. 2. Pseudonymous identities with user-driven identity registration.

This two-phase approach differs technically from the first scenario in that the attributes will have to be included in a proxy certificate derived from the pseudonymous certificate. From the end user perspective this is irrelevant since it is functionally equivalent to ordinary proxy certificates used for single sign-on.

The pseudonymous credential is associated to the real user in the Attribute Authority by having the user pass this mapping directly, as shown in Fig. 2. This method poses a possibility for misuse as the user can pass somebody else's pseudonym thus causing mismatch between the mapping in the online CA and that in the Attribute Authority. Complex and rigorous controls would have to be implemented for this method.

In contrast to the first scenario, the pseudonymity service has no possibility to discern whether the requirement of non-uniquely identifying user attribute sets (FR7) is met. This responsibility is pushed entirely to the Attribute Authorities.

C. Third scenario: Pseudonymous identities with automatic identity registration

A second method of mapping the pseudonym to a real user within the Attribute Authority is for the pseudonym service to contact the Attribute Authority and register the pseudonymous credential on behalf of the user. This prevents the Grid user from tampering with the pseudonymous to real user credential mapping in the Attribute Authority. Once the Attribute Authority has this mapping, the Grid users can contact the service and request their user attributes to be added to the pseudonymous credential just like they do with their real credentials.

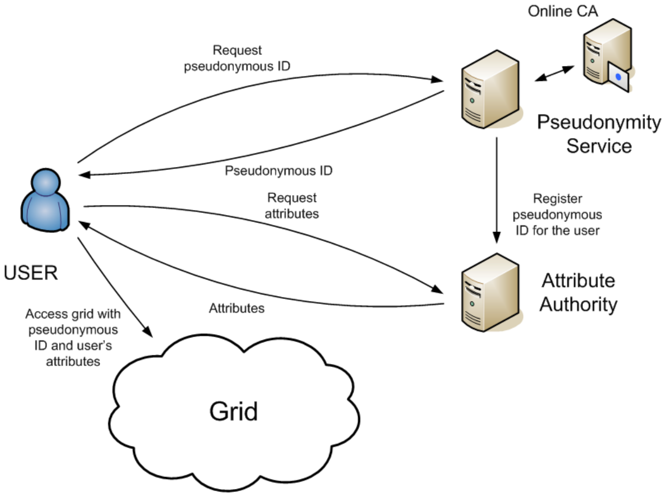

Fig. 3. Pseudonymous identities with automatic identity registration.

In addition to the simplification of the pseudonymity service described in the second alternative, this final solution has the benefit of removing the possibility of tampering with the pseudonymous identity. Another benefit is that the existing client software implementations can be used for requesting the attributes from the Attribute Authorities by pointing them to use pseudonym credentials instead of the real user credentials.

This solution is the one deemed most promising as it takes the best advantage of the existing systems incurring the least changes to them. Also from the user point of view the system has a distinct pseudonymity step and after that the Grid systems operate the normal way.

VII. Conclusions

This paper describes the requirements and initial design of a general pseudonymity service that provides pseudo anonymous access to the Grid. The system allows the users to employ their attributes to access the Grid while hiding their true identity.

The repercussions of hiding the user's identity are hard to determine without getting real world experience with a prototype. A prototype will also shed light on the magnitude of the information leakage problem (req. 6). Currently, to reduce the risk of leaks both the community of users employing pseudonyms and the mix of actions and operations they perform in a Grid need to be large. Ideally, every action on the Grid would use a different pseudonym and use it only once. This makes it difficult to correlate different actions of any single user. On the other hand, even different one-time pseudonym identifiers may be correlated if they are used from the same IP address and this address is not used by any other pseudonym user. The pseudonymity system may thus only partially counteract the leakage problem. Ultimately, the users themselves are required to actively reduce such risks.

The large groups needed for preventing the correlation of actions to a single user pose also a problem. For example using pseudonyms for file access means that the access has to be based solely on the groups and attributes of the user. If the groups are large, it means that there are many people that have access to the files, thus there is less privacy of the files. Also there is less compartmentalization of users and thus in case of compromise of a user has bigger potential for damage. In the end the group size is a balancing act between the VO needs of identity hiding and resource security.

Also, some legislative concerns must be addressed. For example, in some countries the law expects site administrators to know the real identities of the users. The pseudonymity system does maintain the link, that is obtainable, between the real user identity and the pseudonymous version but this may not be enough for some regulations. Unless some governmental identity escrow is available, this effectively bans the usage of pseudonyms within these jurisdictions.

It is foreseeable that for large-scale deployments the certification policy and the pseudonym user's authentication sequence should be approved by a Grid Policy Management Authority (GridPMA). Another issue in probable conflict with most site policies is the generation of long-term pseudonym credentials for anonymizing long running jobs. Also, a long-term credential implies a higher security risk than an ephemeral one.

VIII. Future Work

The near term work is to implement a prototype of Fig. 3. It will allow us to gain practical experience of the system and the identified problem areas, especially the information leakage problem.

Another important goal to pursue is to ensure the mix of operations and pseudonymous users is sufficiently broad to prevent correlation attacks. Also, the connection source tracking needs to be investigated. To this end, grid portals are ideal: using pseudonyms through a portal effectively prevents IP addresses from being tracked assuming job results are also retrieved through the portal or stored into the grid storage using a pseudonym. We will explore the benefits and drawbacks of including portals into the overall architecture.

ACKNOWLEDGMENT

The authors wish to hereby thank Olle Mulmo, Ákos Frohner and David Groep for fruitful discussions, ideas and feedback contributing to the preparation of this paper.

REFERENCES

[1] EGEE Design Team, "EGEE middleware architecture and planning (release 1)," Tech. Rep., Aug. 2004.

[2] O. Mulmo, "Global security architecture for web and legacy applications," Enabling Grids for E-science in Europe, Tech. Rep., Sep 2005.

[3] J. Hughes and E. Maler, "Security Assertion Markup Language (SAML) 1.1 Technical Overview, Committee Draft," May 2004, http://www.oasis-open.org/committees/security/.

[4] ——, "Security Assertion Markup Language (SAML) 2.0 Technical Overview, Working Draft 04," Apr. 2005, http://www.oasis-open.org/committees/security/.

[5] The Globus Alliance, "GridShib Project Web Site," http://gridshib.globus.org.

[6] V. Welch, T. Barton, K. Keahey, and F. Siebenlist, "Attributes, Anonymity, and Access: Shibboleth and Globus Integration to Facilitate Grid Collaboration," in *4th Annual PKI R&D Workshop: "Multiple Paths to Trust"*, NIST Gaithersburg MD, USA, Apr. 2005.

[7] J. Camenisch and A. Lysyanskaya, "An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation," in *EUROCRYPT '01: Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques*. London, UK: Springer-Verlag, 2001, pp. 93–118.

[8] J. Camenisch and E. V. Herreweghen, "Design and implementation of the idemix anonymous credential system," in *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security*. New York, NY, USA: ACM Press, 2002, pp. 21–30.

[9] BEA Systems, BMC Software, CA, IBM Corporation, Layer 7 Technologies, Microsoft Corporation, Novell, and VeriSign, "Web Services Federation Language (WS-Federation), Version 1.1," Dec. 2006, http://www.ibm.com/developerworks/library/specification/ws-fed/.

[10] S. Tuecke, V. Welch, D. Engert, L. Pearlman, and M. Thompson, "Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile," RFC 3820, IETF, June 2004. [Online]. Available: http://www.ietf.org/rfc/rfc3820.txt

[11] R. Alfieri, R. Cecchini, V. Ciaschini, L. dell'Agnello, Á. Frohner, A. Gianoli, K. Lőrentey, and F. Pataro, "VOMS, an Authorization System for Virtual Organizations," in *1st European Across Grids Conference*, Santiago de Compostela, Spain, Feb. 2003.

[12] L. Pearlman, V. Welch, I. Foster, C. Kesselman, and S. Tuecke, "A Community Authorization Service for Group Collaboration," in *POLICY '02: Proceedings of the 3rd International Workshop on Policies for Distributed Systems and Networks (POLICY'02)*. Washington, DC, USA: IEEE Computer Society, 2002, p. 50.

[13] R. Housley, W. Polk, W. Ford, and D. Solo, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile," RFC 3280, IETF, Apr. 2002. [Online]. Available: http://www.ietf.org/rfc/rfc3280.txt

[14] S. Farrell and R. Housley, "An Internet Attribute Certificate Profile for Authorization," RFC 3281, IETF, Apr. 2002. [Online]. Available: http://www.ietf.org/rfc/rfc3281.txt

[15] EJBCA, "The J2EE Certificate Authority Web Page," http://ejbca.sourceforge.net/.

[16] J. Basney, M. Humphrey, and V. Welch, "The MyProxy online credential repository," *Software: Practice and Experience*, vol. 35, no. 9, pp. 801–816, July 2005.

[17] P. A. et al., "CREAM: A Simple, Grid-accessible, Job Management System for Local Computational Resources," in *Proceedings CHEP06, Mumbai, India*, 2006.