Using Policy-based Management for Privacy-Enhancing Data Access and Usage Control in Grid Environments

Wolfgang Hommel
Munich Network Management Team
Leibniz Supercomputing Centre
D-85748 Garching n. Munich, Germany
hommel@mnm-team.org

Abstract

Preventing the misuse of personally identifiable information and preserving user privacy are key issues in the management of IT services, especially when organizational borders are crossed. In this paper, we first present an analysis of the differences between Grid environments and previous models of inter-organizational collaboration. Based on requirements derived thereof, we demonstrate how existing policy-based privacy management architectures can be extended to provide Grid-specific functionality and can be integrated into existing infrastructures. Special emphasis is put on privacy policies which can be configured by users themselves, and distinguishing between the initial data access and the later data usage control phases. We also discuss the application of this approach to a XACML-based privacy management system.

1 Motivation

As an essential part of their terms of use and privacy statements, organizations define which information about a customer and its users they demand in order to provide a service, and for which purposes the collected data will be used. Often, this information is required for accounting, billing, and service personalization. Generally, it thus includes personally identifiable information (PII), i.e., data that can be used to uniquely identify a single person.

To prevent misuse of this sensitive data, such as selling addresses to marketing agencies, legislative regulations exist which restrict how the collected information may be used. One common principle is that data must only be used for purposes which the user has agreed to. As intra-organizational solutions, so-called privacy management systems have successfully been deployed over the past few years. Whenever a user’s data is about to be accessed, rulesets are evaluated to determine whether the current access attempt is in accordance with the privacy policy the user has agreed to. In inter-organizational service usage scenarios, such as Grid computing, privacy protection becomes a much more complicated issue, because multiple organizations – typically also located in different countries – are involved and service providers need to retrieve user data from the user’s home organization in an automated manner.

Privacy management thus becomes a two-tiered process: First, users must decide which of their data may be submitted to a service provider at all, and second they must be able to monitor and control how their data is being used. In the research areas of privacy enhancing technologies (PET) and federated identity management (FIM), various solutions to these issues have been suggested, with several of them already being used in production environments; a short overview will be given in section 2.

However, these solutions are not suitable for certain characteristics of Grid environments, such as the concept of virtual organizations, and cover only the user’s PII, thus neglecting data submitted along with Grid jobs, such as medical records used as input data for those programs. We discuss these differences of Grid environments in section 3. In section 4, we demonstrate how existing policy-based privacy management can be adapted to provide the additional functionality required in Grids. A proof of concept based on the policy language XACML (eXtensible Access Control Markup Language) is presented in section 5. The integration of the discussed privacy management components into existing infrastructures is outlined in section 6.

2 Privacy management in distributed collaborative environments

The various privacy management issues on an inter-organizational level are neither a new nor a Grid-specific research issue. We confine the following discussion of the
data is only used for the purposes agreed to by the user. Enhanced solutions additionally provide interfaces to the users, so that they can verify that their privacy preferences have not been violated. Privacy management systems, such as EPAL (Enterprise Privacy Authorization Language), are typically also policy-based. Access to user data by any application is handled by a privacy policy enforcement point. A PDP decides whether the application is allowed to access a particular user attribute for a given purpose. Thus, the key difference to traditional access control is the consideration of the purpose behind the data access.

Furthermore, so-called obligation monitors can be used to trigger the fulfillment of obligations which are part of privacy policies. Obligations can, for example, be used to restrict the data retention, so all user data has to be deleted 90 days after service usage has been finished and all invoices have been settled. As an organization’s privacy policies may change over time, it is vital for the privacy management system to keep track of which version of the policy was in use when a user signed up for a service. The sticky policy paradigm [3] glues the relevant policies to the user data so they cannot be separated anymore. Protocols of data access and usage are kept to support the organization’s internal auditing processes. Parts of this information can be made available to the user to prove that her data has only been used for the intended purposes. However, the usefulness and reliability of this information is very limited, because malicious service providers could arbitrarily falsify the presented data. Thus, all recent approaches are based on certified software running on trusted computing platforms in order to guarantee the genuineness of the information given to the users (see [2]).

3 Grid characteristics and resulting requirements for privacy management

On the technical level, Grid computing is based on a Grid middleware which provides the required transparency layers and tools for submitting Grid jobs. In the past, the middleware development has focussed on the core functionality, but with increasing use in productive environments, security as well as privacy properties become more and more important. As many of the organizations involved in Grid projects have identity management systems deployed, there is an increasing demand to leverage the existing local infrastructure when participating in Grid projects. Concerning privacy management, however, Grids have several characteristics and thus specific requirements which are not yet met by the approaches discussed in the previous section.

Starting with the technical aspects, which are – unlike the organizational issues discussed below – applicable to all Grids in general, it must be considered that using a Grid infrastructure differs from using other distributed systems
and services in the concept of (Grid) jobs. When submitting Grid jobs, the user cannot only provide input data to a static service provided by an organization, but let her own program code make use of the CPU and storage capacities provided by the organizations which are involved in the Grid.

This immediately leads to the consequence for privacy and data protection in Grids that any data related to a user’s Grid job must be treated similarly to the user’s PII:

- The Grid job’s code should be considered intellectual property of the Grid user. This also affects whether a service provider may modify the program code, e.g. in order to optimize it for the local computing architecture.
- Input data for the Grid job may contain sensitive data. In this case, both the Grid user and the service provider share a couple of responsibilities. The Grid user must have the permission to submit the data to the service provider; this is a non-trivial organizational issue because the utilized Grid service providers are, in general, unknown when the input data is collected. Furthermore, the service provider to which a Grid job has been submitted is typically not allowed to make any use of the input data other than feeding it into the Grid job’s code.
- On the service provider side, the considerations for the input data must also be applied to the Grid job’s output data. Depending on the Grid job, the output data may be even more sensitive than the input data. Thus, there must be an agreement about how the output data is being treated, both while the Grid job is running and after it has finished.

Further aspects, such as whether the service provider is allowed to backup or even archive these Grid job components, must also be considered. It is also apparent that privacy and data protection settings may vary with each Grid job, independent of the user’s preferences regarding her own PII. As a consequence, this logical separation between PII and Grid job privacy management should also be accounted for, e.g. in the design of user interfaces.

Unless contractual agreements can be arranged for all organizations within a Grid project, the location transparency offered by the Grid middleware would even contradict the primary privacy management goal, i.e. that users get to know exactly by whom their data is being used. Thus, the traditional approach that users can define privacy preferences on a per-organization basis must be complemented by means to define property-based privacy policies. As an example, certain PII such as the user’s email address should only be distributed to service providers which guarantee to use it for contacting the users in case of technical problems, but not for other purposes such as sending marketing emails.

4 Policy-based privacy management in Grids

Our key motivation for using a policy-based privacy management approach in Grid environments is to leverage existing identity and privacy management infrastructure components in order to reduce the IT service management overhead of solutions specific to the Grid domain. In this section, we motivate how policy-based privacy management can be used in Grid environments and demonstrate how the existing approaches can be extended to incorporate the Grid-specific requirements in general. For any transmission of sensitive data, more than one policy may be relevant; in practice, there typically are four layers of policies:

1. Users can specify their privacy preferences, i.e. the conditions and obligations under which they are willing to release their data to a service provider.
2. The user’s home site (identity provider) has privacy policies in place which typically provide default settings for all of its users.
3. Each service provider has its own privacy policies, which are not necessarily Grid-specific.
4. Grid projects and virtual organizations may have privacy policies which must be honored by all participating organizations and applied to all users.

In most approaches and implementations, the number of layers may vary with scenario-specific requirements, such as service-specific policies in addition to service-provider-wide policies. There can be multiple policies in each layer, and it needs to be determined for each individual data request which policies are relevant. There may be conflicting policies, e.g. if a service provider requires an attribute such as the nationality when the user is unwilling to release it. In practice, disjoint policies are ensured only in the same layer by the management frontends; thus, e.g. users are forced to formulate consistent policies. We must distinguish between privacy policies for PII and for Grid jobs on the user layer:

- The user’s personal privacy preferences will usually stay the same over a certain period of time and are independent of the submitted Grid jobs to a high degree.
- While it must be possible to configure privacy policies for individual Grid jobs, multiple Grid jobs might belong to the same research project. Thus, to reduce the management overhead, privacy policies must be applicable to groups of Grid jobs.

Figure 1 shows the resulting privacy management architecture for the user’s Grid home site. Compared to previously used architectures as discussed above, a logically separated management user interface is provided as part of the self services, which allows to configure project- and
Grid-job-specific privacy policies. Furthermore, not only Grid-wide applicable policies must be exchanged between the involved organizations, but also the policies of those research projects whose users are spread among multiple organizations. The same policy distribution mechanisms can be used for both use cases, as long as they provide metadata support to restrict a) to which organizations the policies are transferred to and b) which other users may access and modify them. The syntactical basis for identifying and naming objects, often referred to as policy namespace, must be extended as follows:

- Instead of targeting a policy to a single service provider, it must be possible to specify policies for arbitrary groups of organizations, up to a Grid environment such as a virtual organization as a whole.
- New identifiers for projects, Grid jobs, and their components, such as code, input data, and output data are required. It is impractical to stick to a predetermined set of elements; instead, the involved organizations must agree on the granularity of the policies and on a common vocabulary to be used in these policies.
- New conditions and obligations are required, for example to state that a Grid job’s code may be modified by the service provider for optimization purposes. Also, obligations such as data retention limits, will typically differ between personal data and Grid job data.

On the service provider side, no extensions to the privacy management architecture would be required, except for the support of the newly defined obligations. Alas, so far only very few Grid service providers support privacy management at all; the integration of privacy management components into Grid-specific workflows is discussed in section 6.

5 Use within a XACML-based framework

To demonstrate the feasibility of our approach, we have applied the extensions described in the previous section to a privacy management framework which was designed for use in federated identity management (see [1]) and uses the policy language XACML. It has been implemented for the Shibboleth federated identity management software and thus is also suitable for use in Grid middleware projects such as GridShib [5]. Like most modern policy languages, XACML supports scenario-specific vocabulary, e.g. for the specification of obligations, without the necessity to extend the PDP; thus, any standard compliant XACML PDP can be used also for the Grid job policies. We have extended the used namespace to support

- the definition of and referring to groups of service providers as well as virtual organization identifiers,
- the specification of Grid projects as groups of Grid jobs, the Grid jobs themselves, and their components;
the granularity exemplarily chosen for the components
is code, input, and output.

- new conditions, such as code-optimization and
disallow-backup, as well as new obligations, e.g.
delete-after-execution.

It is part of our ongoing work to analyze real world Grid
environments to determine which policy granularity and
vocabulary lead to a viable trade-off between very fine grained
control and good usability with low management overhead.
Figure 2 shows an example of a Grid job policy, which al-


Figure 2. Example XACML Grid job policy to allow code optimization

6 Integration of privacy management com-
ponents on the service provider side

As described in sections 2 and 4, the use of FIM pro-
tocols ensures that personal and Grid job data is only dis-
tributed to Grid service providers that are suitable from the
privacy management perspective. Thus, privacy manage-
ment on the service provider side primarily pursues three
goals:

1. All personal and Grid job data may only be used in
accordance with the privacy policies specified by the
service provider.
2. All user and Grid job obligations must be fulfilled,
which requires an obligation monitoring component.
3. It shall be possible for the user to verify whether the
obligations have been fulfilled and that her data have
not been used for other than the agreed purposes.

Figure 3 shows the resulting privacy management archi-
tecture and their interfaces to the Grid middleware. One
goal is the protection of any personal and Grid job data from
direct access, which is achieved by channelizing all data
read, update, and delete attempts through the privacy PEP.
This will require several changes to existing middleware on
the data persistence layer, as well as additional error han-
dling for privacy violation exceptions.

For many supercomputing service providers, this also
necessitates that batch scheduling systems, which enqueue
Grid as well as regular jobs to be run on the machines ac-
cording to a local job execution policy, also contact the pri-


ranting the users reliable insight into how their data has
been used by the service provider remains an open issue:
Grid users presently typically have shell access or can ac-

Grants web portals. This provides
suitable feedback channel, which could be used to make,
for example, logfile excerpts available to the user. However,
there is no guarantee that the shown logged information is
sound and complete. Thus, until secure and trusted oper-
ating systems are used for Grid resources, the user’s infor-
mational self-determination can already be supported, but
the guaranteed enforcement of privacy policies cannot be
verified in an absolute objective manner.
7 Conclusions and outlook

In this paper, we have first motivated the necessity of privacy management in Grid environments. After sketching the state of the art, we analyzed the characteristics of Grid environments, derived their special requirements, and demonstrated that existing approaches fall short of fulfilling these requirements. We then presented how policy-based privacy management can be adapted to Grid environments and applied this methodology to a XACML-based architecture. Finally, we discussed that the realization of a policy-based privacy management approach is a straightforward task for Grid home sites, but complex and challenging for Grid service providers. Open issues from policy-based management research, such as the verifiable guaranteed enforcement of privacy policies, have not been solved in the Grid context either. Our future work will include the analysis of current Grid environments to determine the practically suitable granularity of privacy policies for Grid jobs, based on the concept of layered policies presented here. We will also refine the interface specification between the obligation monitor and the Grid middleware and work on a proof-of-concept implementation. Finally, we plan to analyze how the privacy management for services used via federated identity management and for Grid environments can further be integrated to minimize the administrative overhead and provide a consistent user experience.

Acknowledgment The authors wish to thank the Munich Network Management (MNM) Team for helpful discussions. The MNM-Team, directed by Prof. Dr. Heinz-Gerd Hegering, is a group of researchers of the University of Munich, the Munich University of Technology, the University of the Federal Armed Forces Munich, and the Leibniz Supercomputing Centre.

References