Abstract interdomain security assertions: A basis for extra-grid virtual organizations

by B. E. Carpenter
P. A. Janson

One significant challenge in building grids between organizations with heterogeneous security systems is the need to express and enforce security policies that specify the users in one organization (the source domain) who are allowed to access the resources in another organization (the target domain). This requires linking the syntax and semantics of security assertions referring to users and their attributes in the source domain to those referring to resources in the target domain. This paper suggests some basic requirements for solving this problem, in particular, an *abstract* form of interdomain security assertion (IDSA) relying, for instance, on globally meaningful URIs (Uniform Resource Identifiers) to refer to users, resources, and their attributes. This canonical abstract form of IDSA is, however, used strictly for assertion mapping purposes. It may—but need not—be visible in any concrete security assertion syntax in any domain. The paper further suggests different scenarios in which URIs for users, resources, and attributes defined in one domain can be mapped to semantically meaningful references—with varying degrees of granularity and accountability—in another domain where they would otherwise be meaningless.

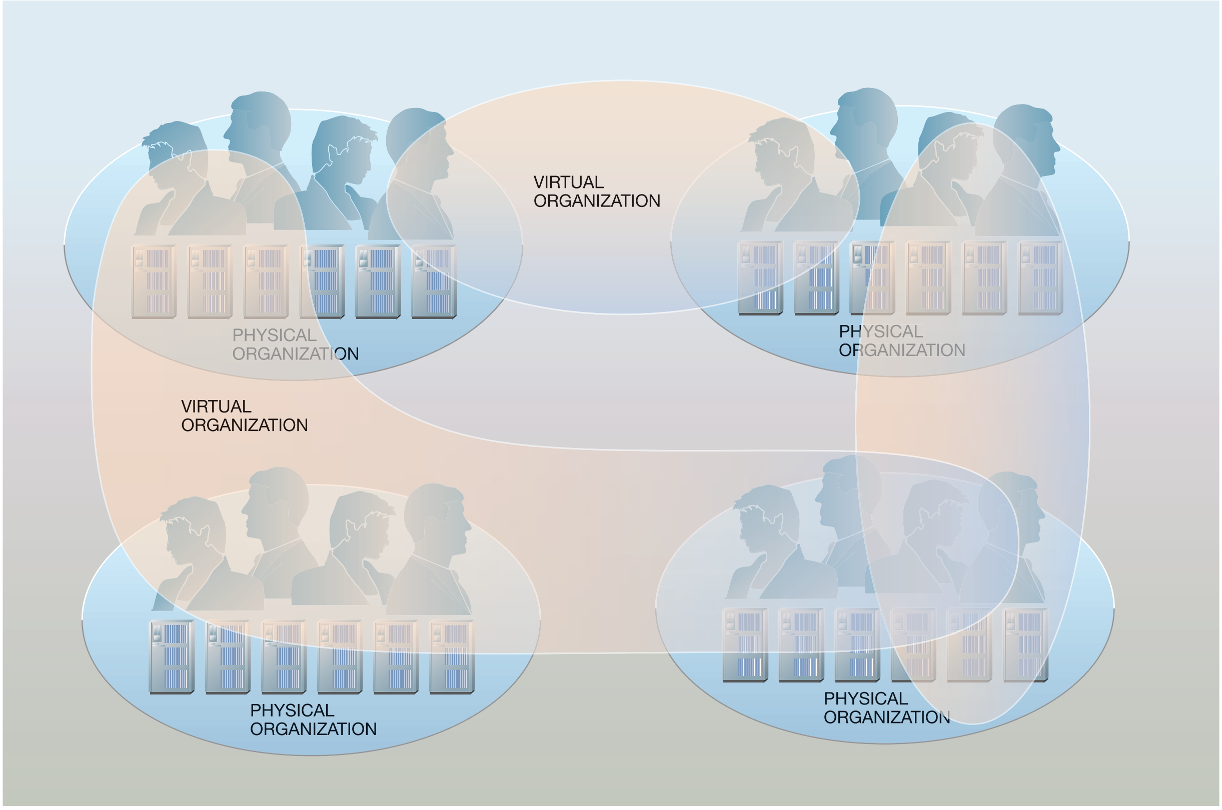
Grids are collections of networked computers that pool their resources together in such a way that users may utilize processing, storage, software, and data resources from any of the interconnected computers, leading to greater resource sharing and higher utilization ratios. Such grids can have many different definitions and objectives and may exhibit many different properties. For the purposes of this paper, the key characteristic of a grid is that it allows organizations to pool computing resources (processors, storage, information, applications, etc.) to enable users to benefit from a potentially far larger pool of resources than would otherwise have been available to them.

Terminology. From this perspective, the ultimate grid would include the whole Internet, so that anyone could tap anyone else's resources, assuming they were available and financial compensation was arranged. While isolated examples of grids of this type are certainly emerging—for instance, the SETI@home and folding@home efforts—a generic and global-scale scenario of this type, which we call an *inter-grid*, is not about to happen. The ideas to be discussed in this paper would certainly assist in implementing such a scenario, but they are in no way sufficient to address all issues.

At the other end of the spectrum, today, most grids are internal to some existing organization. Such grids, which we call *intra-grids*, pool resources across departments, sites, or other entities within some larger organization. Because of the existence of such a larger organization, the entities participating in the

0018-8670/04/$5.00 © 2004 IBM

Figure 1 A pattern of extra-grid virtual organizations



intra-grid have often deployed their own information technology (IT) infrastructures according to common architectural guidelines and using a relatively simple trust model. The resulting grid thus benefits from relative homogeneity among the interconnected systems, which greatly simplifies the pooling of resources. Such intra-grids generally do not need any of the ideas discussed in this paper.

This paper addresses challenges encountered in what we will refer to as *extra-grids*, namely, grids resulting from the pooling of resources across organizational entities that did not follow common architectural guidelines to deploy their IT infrastructures. Such grids emerge, for instance, when different organizations are integrated following a merger or acquisition, or when separate organizations decide to pool their IT resources for some common goal but otherwise want to retain their autonomy. In the merger or acquisition scenario, all resources of participating organizations may become part of the new pool, and all users of participating organizations may in principle enjoy some access to resources of other organizations. By contrast, in the second scenario, typically, only some resources from each of the participating organizations will be pooled, while most others will remain outside the pool, and only a subset of the users of each participating organization may use any of the pooled resources.

The grid community often refers to the notion of a "virtual organization" (VO). In the context of this paper, this notion of a VO corresponds to the set of resources that are pooled and the set of users who can tap these pooled resources. Figure 1 shows a sample scenario of how VOs may typically be formed. The issues of how a VO is named, referred to, represented, populated, and managed over time, and issues of how grid policies are set, how information may be imported to or exported from a VO, and how intellectual property issues may be settled between a VO and its participating organizations are all important, but they are outside the scope of this paper. The only aspect of a VO that is important for this paper is the trust that it implies between participating organizations. How that trust can be represented and materialized will be discussed subsequently.

Objectives. This paper focuses on the issues arising in extra-grids from the need to define and enforce security policies spanning different organizations using heterogeneous security mechanisms within the same VO. When organizations using heterogeneous security regimes decide to pool some of their resources into a VO for the benefit of some of their users, they need to be able to specify which of their users should have what rights to access which of their resources under what circumstances. The challenge is that such cross-organizational policies are typically beyond the normal expressive power of security assertions used to represent policies in any of the individual organizations participating in the VO. Specifically, the syntax and semantics of security assertions in one organization have no immediately obvious way to refer to entities (users, resources, attributes) defined in some other domain. As a result, if a user or application authenticated in one organizational security domain presents its security credentials to another organization in the same VO, using different credential syntax and semantics, those credentials simply cannot be understood by that target organization.

Even if two organizational security domains trust one another, as they do in a common VO, and even if they have established some sort of security service gateway to translate the syntax of foreign credentials into locally understandable ones, any user name and attribute from one domain would still be undefined semantically in the other domain. Specifically, even if two organizations, for example, example.com and targetex.org know and trust one another and are able to translate the credentials for a user called Alice in the domain example.com into a format palatable for targetex.org's security infrastructure, the name Alice would still be meaningless because it was never defined and seen before in targetex.org's domain. No form of standardized authorization query interface could even help paste over these differences and obviate the need to refer to foreign users, resources, or other entities in whatever domain a policy is defined.

The challenge is compounded by a frequent grid requirement[1] that each organizational security domain within a VO should retain full control over who can access which of its resources. Thus, it is entirely targetex.org's choice whether Alice should have access to its resources. Not even knowing about the existence of a user ID such as Alice in the domain example.com, much less about what it means, in the absence of other measures, targetex.org could not use that name to express Alice's rights in any security policy it wants to set in its own domain. A mechanism is required by which a security domain in a VO can express authorization (or denial thereof) of some local resource access by an alien identity from another domain (in this example, Alice).

This paper does not describe a complete architecture, much less a working design, for implementing extra-grids, nor does it describe a concrete solution to the extra-grid security challenges raised above (many frameworks and standardization proposals are being developed to this end.) It merely makes a number of fundamental observations about these challenges and suggests that some basic assumptions about abstract security assertions seem to be required for any of the emerging frameworks and standards to succeed in addressing the challenges of extra-grids. It further suggests a number of possible scenarios, offering different degrees of granularity and user accountability, for linking user, resource, and attribute references across heterogeneous extra-grid domains.

In the section "Communication between domains within a VO," we briefly review how interdomain communication is assumed to take place in intra-grid scenarios. In "Use and modeling of security assertions in interdomain scenarios," we briefly discuss the notion of security assertion and review what forms such assertions are expected to take in extra-grid scenarios. The requirement for some interdomain security assertion (IDSA) abstraction is derived in "Fields required in an IDSA." In the following section, we suggest different design approaches by using abstract assertions for enabling the mapping of concrete security assertions at the boundaries between heterogeneous security domains. This is followed by a few remarks on how the suggested abstract security assertions fit in the context of the emerging Open Grid Services Architecture (OGSA),[1] after which the paper is summarized and concluded.

Related work. The problem of interdomain security in heterogeneous systems has been recognized for quite some time and addressed in many early designs. Greenwald[2] discussed the issue of mapping name semantics across domains, which he called compartments, and proposed the notion of a "handle" for what would today be referred to as interdomain security assertions. The complexity, size, and heterogeneity of the world he was addressing at the time had, however, nothing in common with what extra-grids are facing today. De Capitani and Samarati[3] address a more general form of security policy management for what they call "federated" domains, a terminology used today in grid and Web services scenarios. They also recognize the need for domain autonomy in setting access controls on their own resources and the problem of foreign references being undefined in local domains. However, they solve these issues by superposing a full federated layer with its own superdomain mechanisms, which imposes far more coupling than is desirable in today's grids.

Kindred and Sterne[4] propose an Internet Domain Name Service-based design for mapping security references at the boundaries between dynamically evolving domains in virtual private networks (VPNs). The design rests on a VPN membership administration mechanism, which addresses issues at a much lower layer than what is needed in grid scenarios.

More recently, in the context of the OGSA, the Global Grid Forum (GGF) Authorization Framework[5] has also recognized the interdomain problem and even suggested in its appendix a number of configurations for linking authorization information across domains. It is, however, just a framework and provides only a context for discussing solutions without details on any concrete design. The GGF OGSA Authorization Requirements proposal[6] goes one step further in identifying requirements for concrete designs, but does not dwell on the interdomain issue in particular. A more directly relevant GGF publication will be referred to later, in context.

Communication between domains within a VO

Security mechanisms concerned with identification, authorization, and normal grid interactions will require communication between the organizational domains constituting a VO. In a grid environment we assume that these interdomain communications use Open Grid Services Infrastructure (OGSI) mechanisms—specifically, WSDL[7] (Web Services Description Language) and SOAP[8] (Simple Object Access Protocol). The emerging update of OGSI as the Web Services Resource Framework (WSRF) does not affect this assumption.

As suggested in Figure 2, we normally assume that interdomain messages are secure against snooping or modification. This requirement applies everywhere, not just during the hop from one real organization to another—extra-grid VO traffic needs to be protected even within the boundaries of its participant organizations. From this we draw two conclusions:

1. All grid messages, including interdomain messages, within an extra-grid VO are assumed to be fully protected with Web services security (WS-Security).[9]
2. Given this, secure interdomain VPNs or link encryption may be deployed but are not required.

We should add, however, that even when grid mechanisms are used to establish contact between a service requestor and a service provider, it is possible that actual execution of the service will involve protocols other than OGSI. In this case, WS-Security cannot be relied upon, and a secure VPN may still be required. This is outside the scope of this paper.

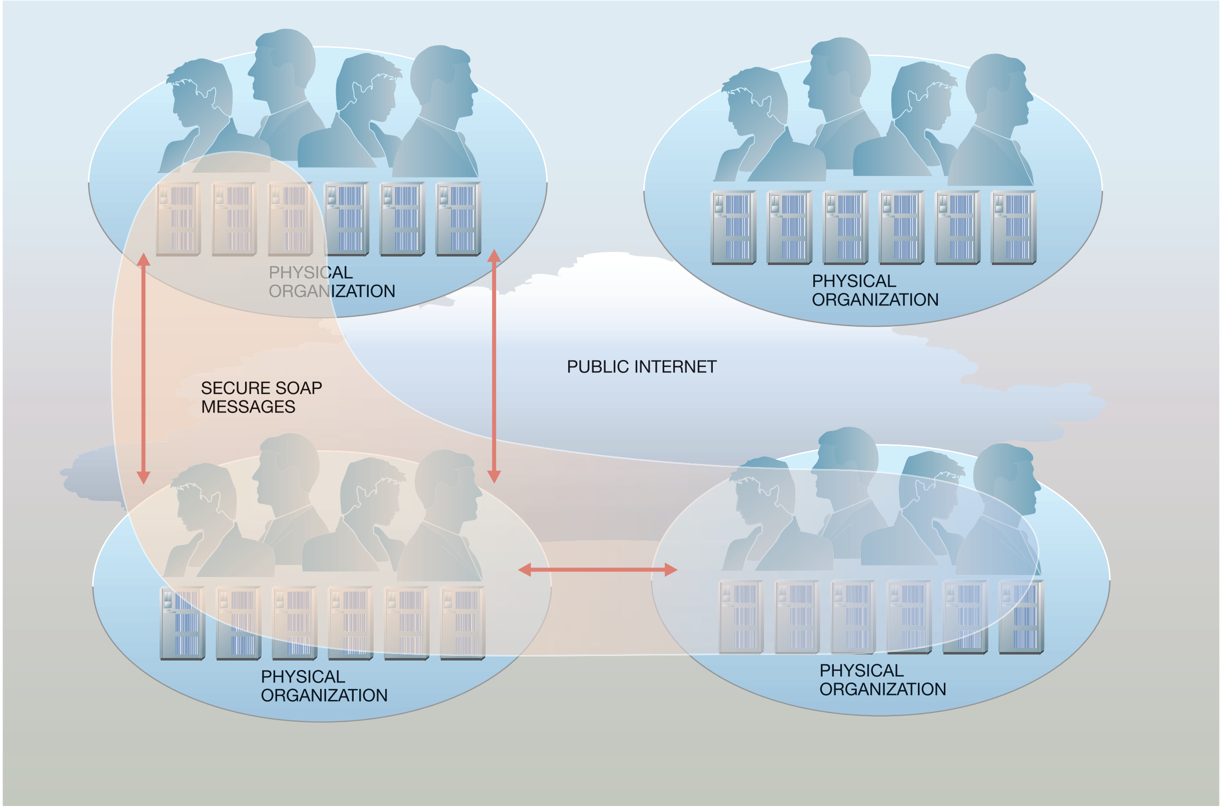
Use and modeling of security assertions in interdomain scenarios

An extra-grid VO federates two or more security domains, each with its own methods for identification and authentication of users and for authorization to access resources. A user identified and authenticated in one domain may need to be authorized for resource access in another domain. To achieve this, security assertions (SAs) describing who can access what, where, and under which circumstances will need to refer to entities (users and resources) in different organizations within the VO.

In general, we must assume that different forms of SA are in use in different domains. Specifically, we can expect to find Kerberos tickets,[10,11] X.509v3 certificates,[12] Globus** Grid proxy certificates, proprietary solutions such as IBM RACF* (Resource Access Control Facility), Microsoft Passport or others, as well as solutions based upon SAML (Security Assertion Markup Language) in the future.[13]

Thus there will, inevitably, be a need to translate or exchange between the security assertion syntaxes of different participating domains. Many interdomain security solutions, (e.g., the solution described in Reference 3) have been designed in the past around the idea of superposing a full federation protocol layer on top of participating domains, thus forcing all par-

Figure 2 An extra-grid virtual organization on the public Internet



ticipating domains to implement an extra mechanism to enable interdomain security. In today's global-scale Internet, this approach is no longer tenable. It is however, necessary that all participating domains, if they do not support a common SA syntax, at least need to understand some common *abstract* SA semantics to ensure that any field relevant to the VO in one SA syntax can be meaningfully mapped into some corresponding field in the SA syntax of any other VO domain. Since VOs may be formed recursively and among arbitrary (overlapping) sets of organizations, the only practical assumption is that there must be a general agreement about the abstract semantics of these assertions.

We refer to such abstract assertions as interdomain security assertions (IDSAs). We contend that all existing SA syntaxes convey semantic information that can be covered by the abstract semantic fields described in the next section. In other words, no existing concrete SA system design ever needs to carry a field whose semantics are not covered in the following section. We cannot emphasize enough that the proposed IDSAs discussed in that section are purely abstract. They might be mapped into any concrete IDSA syntax, such as SAML, but do not need to be. In that sense, abstract IDSAs are totally independent of the syntax.

One could rapidly extrapolate from this that IDSAs should be defined with a standard concrete syntax and should be transmitted between the domains within a VO using a standard protocol. Such an extrapolation is indeed the subject of a GGF recommendation.[14] This again leads to imposing unified SA protocols across all domains, a solution which is generally unacceptable in loosely federated interdomain scenarios. While the suggestion may be attractive in some respects, it may specify both too much and yet too little to address the extra-grid security requirements envisioned in this paper. It may specify too much in that we prefer to seek consensus on the IDSA as an abstraction before doing so definitively at the syntactic level, which would be convenient but is not strictly required. At the same time, it is insufficient in that, as will be seen in the rest of this paper, much more than agreement on SAML assertions is required to enable authorization in extra-grid scenarios: mechanisms for enabling the mapping of semantics across domains are required and not built-in to SAML or any other interdomain security design we know of, although Reference 4 offers hints in that direction in a VPN context.

Fields required in an IDSA

In the most general case, an IDSA is an abstract representation of a statement specifying the permission of some principal or subject (and/or group) to operate in some way (through some action or operation) on some target resource or object. It can be used (as a concrete syntax) in authentication and authorization interactions between domains, but more generally it can be used in the abstract to define how fields from one form of concrete SA may be mapped to another.

We contend that all existing SA technologies (most notably SAML, among the most recent ones) use some subset of the following semantic fields:

1. User ID (subject NameIdentifier in SAML)
2. Authentication method and strength and context (authentication method in SAML)
3. Group/attributes (user attributes in SAML)
4. Resources granted (resources in SAML)
5. Entitlements granted (resource attributes in SAML, "wild cards" in the proposal[13])
6 Operations granted (actions in SAML)
7. Privacy restrictions (conditions in SAML or obligations in XACML[15] [Extensible Access Control Markup Language])

It should be noted that the first three of these fields pertain to and are defined in a user's home service domain while the last four pertain to and are controlled by the service domain where the affected resources reside. Thus, a complete IDSA relies on authentication regimes and authorization policy mechanisms that reside in two domains. This represents a first explicit extension to the model suggested in Reference 14, which talks only about pushing assertions from the subject domain to the target resource domain or pulling them from within the target resource domain. Pushing alone would require that resource access controls be defined in the requesting subject domain, which is contrary to current thinking in the grid community. Pulling from the target resource domain would assume that any potential VO user in a requesting domain be predefined in the target resource domain, which would require potentially impractical preconfiguration, which is generally not done. The contribution of this paper is the observation that extra-grid scenarios will require combining elements of assertions under the control of source and target domains in practice, and this will in turn require suitable mapping mechanisms to build complete assertions from disparate elements.

It is also noteworthy that privacy restrictions (normally attached to data resources) are typically "sticky" in the sense that they must stay with the data; that is, if a user in one domain requests data services from another domain, whatever privacy restrictions are attached to the resulting data must travel back to the requesting domain and subsequently be applied to any future request for the same data in that domain.

At the minimum, an IDSA must include either a user ID, a group, or an attribute describing the "permission level" claimed by a requestor, and it must specify a resource or entitlement describing what the requestor is trying to access. There can be a default to some implicit value for all other parameters.

IDSA fields related to users. The following fields define users and groups.

User ID—This is initially defined in the user's home domain (but is likely to be mapped, as we describe subsequently). In an IDSA, it would also need to include a user's home domain name or descriptor as well as a format descriptor (as indeed appears in a SAML assertion). All concrete SA syntaxes use some form of user ID.

Authentication method and strength and context—These include biometric, smart card, secure token, or password for method and strength, and date/time, location, authenticating entity, and incoming user access channel for context. These are set in the user's home domain at the time of authentication. They qualify the user ID, and may be used by the authorization process to decide whether the ID authentication is sufficiently trustworthy. Few concrete SA syntaxes actually provide any such support today, but recent developments such as SAML suggest that such features may become increasingly common in the future.

Group/attributes—These are also defined in the user's home domain to qualify the user ID and may be referenced in a remote domain by the authorization process. Groups are analogous to UNIX** groups, but may be defined recursively. (The top-level group in a VO is likely to be the set of all users within that VO.) Roles, like those found in role-based access control systems, are also typical examples of user attributes, akin to (and often implemented as) groups. Other examples of attributes include citizenship, top secret clearance, clearance for bank transfers above $1 million, or clearance to spend $1 thousand worth of processing and storage resources per day; that is, characteristics of users that classify them into various categories and may entitle them to some privileges. While most concrete SA syntaxes today support some form of group or role mechanism, support for other attributes is rare.

IDSA fields related to resources. The following fields are defined by the authorization policy management process of the resource domain.

Resources granted—Identifies the resources to which this IDSA gives access.

Entitlements granted—It is often impractical to formulate SAs for each individual resource. Modern authorization management systems formulate SAs on the basis of collections of like resources. Entitlements are a vehicle to this end. An entitlement is some regular expression defining a collection of resources, which qualifies the resources by classifying them into categories, just as user attributes qualify users. Examples of entitlements would be all cars with New York license plates, all bank accounts with balances under $1 million, all Web pages below this home page, or all processors or disks in a "farm." Support for entitlements in concrete SA syntaxes is rare today.

Operations granted—What the user/group is or is not (in the case of negative rights) allowed to do to the resources (e.g., read/write/execute, consume 200 CPU hours worth of processing power, transfer amounts under $10 thousand).

Privacy restrictions—Sticky privacy constraints attached to target resources (e.g., "do not copy," or "DB can be read for statistical collection purposes but not for printing of individual records"). Support for expressing and carrying privacy restrictions in concrete SA syntaxes is rare today but might develop as privacy issues become increasingly sensitive in many modern application settings.

Making IDSAs canonical. Even though we are discussing only abstract semantics and not concrete syntax, it is highly desirable to make the abstract fields canonical (i.e., design them so that there is only one unified global way to express a given semantics). To that end, we suggest that, whenever possible, IDSA fields should be represented using a URI (Uniform Resource Identifier) as the abstract syntax, regardless of whether and how this is used in any concrete representation. This suggestion is in fact similar to what is found in SAML and in the architecture proposal in Reference 14.

User IDs, groups, attributes, entitlements, and resources can be referred to as URIs.

An abstract user ID should be thought of as a mailto: URI, such as mailto:alice@example.com. If not, it must be represented as some other duly registered form of URI to be understandable across heterogeneous domains in an extra-grid. Using proxies with restricted access privileges in lieu of user IDs would also be possible, but this would change nothing because they would need to be identifiable by some form of accepted URI.

An abstract group or attribute could be thought of as the scheme: //example.com/g/groupname or the scheme //targetex.org/attr/attributename. An abstract resource must be thought of as a URI. For instance, a disk resource could be represented as ftp://site1.example.com/?disk=10TB. This merely illustrates the gist of this proposal to use URIs to represent everything in extra-grid SAs.

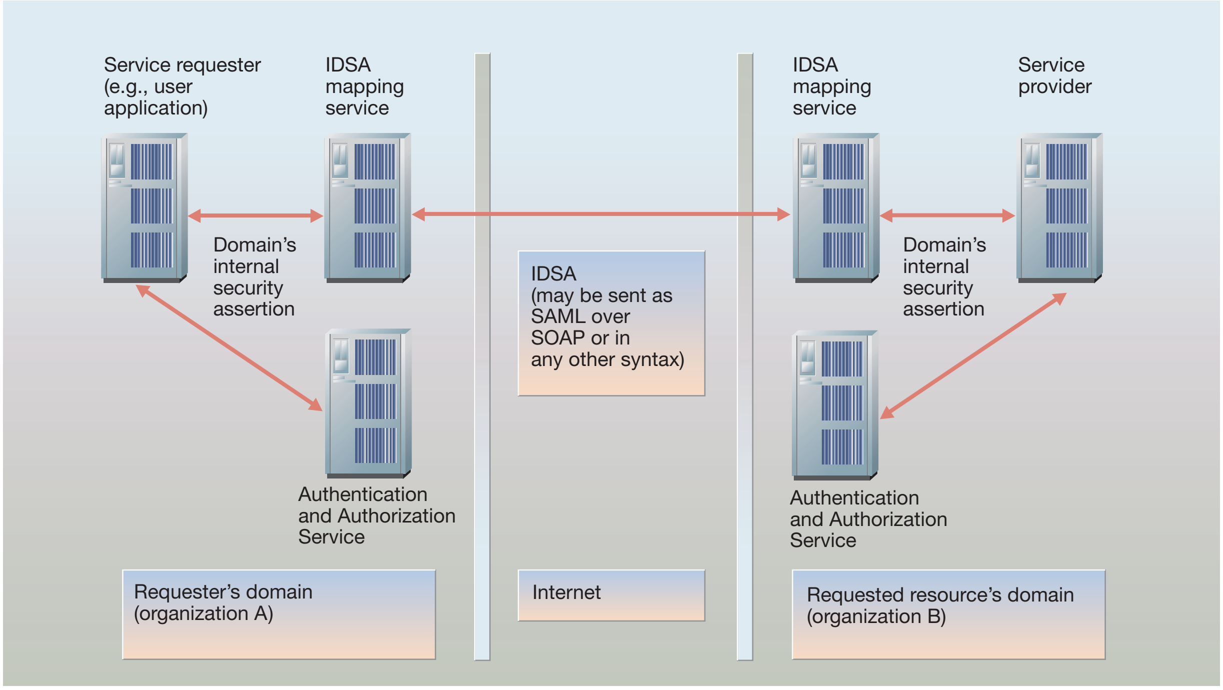
An abstract entitlement could be thought of as http://example.com/home (referring to all pages in that subtree). Alternatively, a SELECT or other SQL (structured query language) statement defining some view on a target database could be used to define the scope of a user's entitlement to access the target fields addressed by that statement.

It should also prove possible to represent the other fields as URIs—SAML certainly aims in that direction. This is not unlike the way XML (Extensible Markup Language) defines any of its namespaces as a URI, even though that URI need not denote any resource on the Web that is truly addressable.

IDSA mapping

Of course, an organization (and a grid implementation) could use a concrete form of IDSA natively,

Figure 3 IDSA mapping



Service requester (e.g., user application)
IDSA mapping service
IDSA mapping service
Service provider
Domain's internal security assertion
IDSA (may be sent as SAML over SOAP or in any other syntax)
Domain's internal security assertion
Authentication and Authorization Service
Authentication and Authorization Service
Requester's domain (organization A)
Internet
Requested resource's domain (organization B)

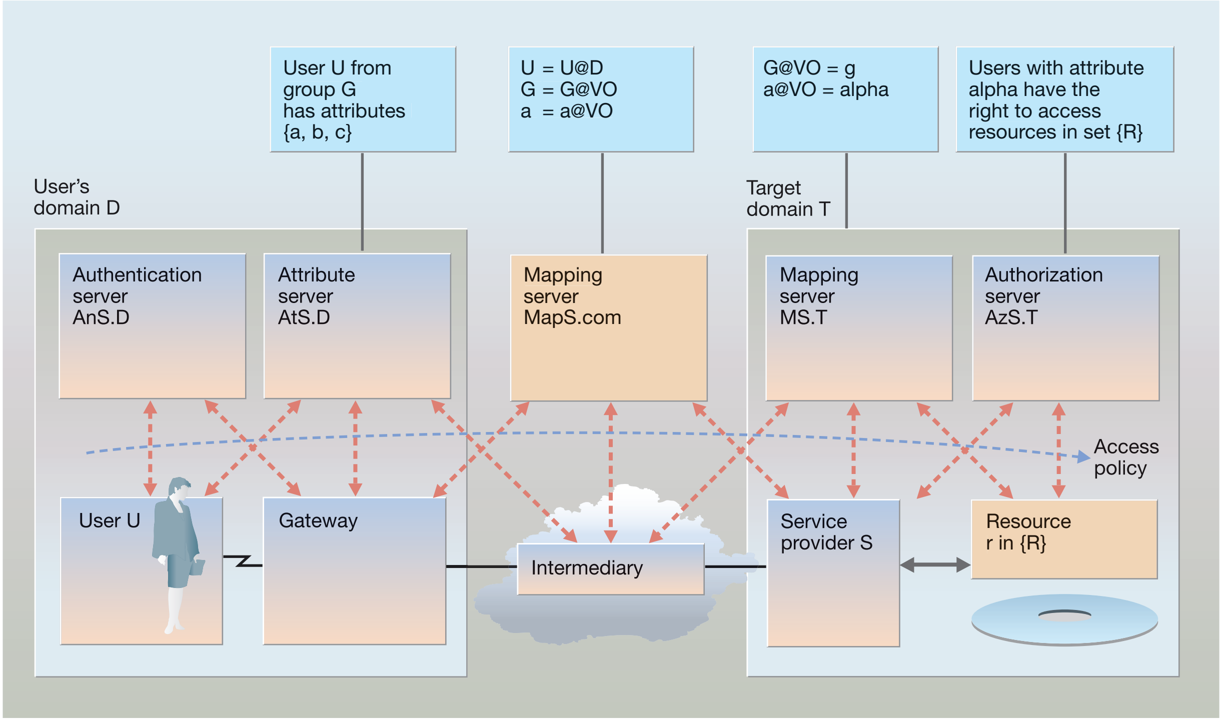
as suggested with SAML in Reference 14, but this is not required, and is unlikely to be practical or in fact useful in cases where Kerberos, X.509v3, or proprietary solutions are already deployed. Thus, by default we assume that SAs expressed in the policy syntax of the local domain will need to be translated (conceptually and actually) to and from abstract IDSAs at domain boundaries within a VO. The power of using URIs wherever possible in IDSAs is that these may—and, in that case, should—be used without translation within the individual domains, unless a proprietary domain is ignorant of URIs, in which case mappings are unavoidable.

Figure 3 suggests a typical setup for the mapping service between requesting user and target resource domains. Figures 4 and 5 illustrate by example the setup and the time line that a typical request goes through, and suggest when, where, and how IDSAs are involved. This is only one of several possible scenarios. It should not be construed as unique and absolute, and indeed not even necessarily one of the best solutions.

Referring to these two figures, when user U in domain D and group G wishes to access resource R in domain T, U first acquires whatever credentials (identity, group memberships, attributes, permitted operations) it needs from its local authentication server (AnS.D) and attribute servers (AtS.D), and these credentials are expressed in the SA syntax of a record in domain D. This is accomplished using whatever native protocols are in use in that domain, possibly Web or grid services protocols.

User U then dispatches its request, together with the relevant credentials, to the target resource. Network routing and related functions will take the request as far as possible towards the target until some gateway along the way—presumably a SOAP intermediary—catches it to intervene and map the carried credentials (and target resource identifier, if it is not in a form suitable for direct use in its own domain). In our example, this happens to be a mapping server in some random third domain, MapS.com. (That mapping server might as well be a dedicated server inside domain D but it does not have to be—it can

Figure 4 IDSA topology



also be some shared service somewhere within the VO.)

The mapping server understands the SA syntax of domain D and knows how to map user IDs, group IDs, and attributes into abstract URIs. It then passes the request on to the next intermediary, which in our example happens to be another mapping server, this time inside the target domain, MS.T.

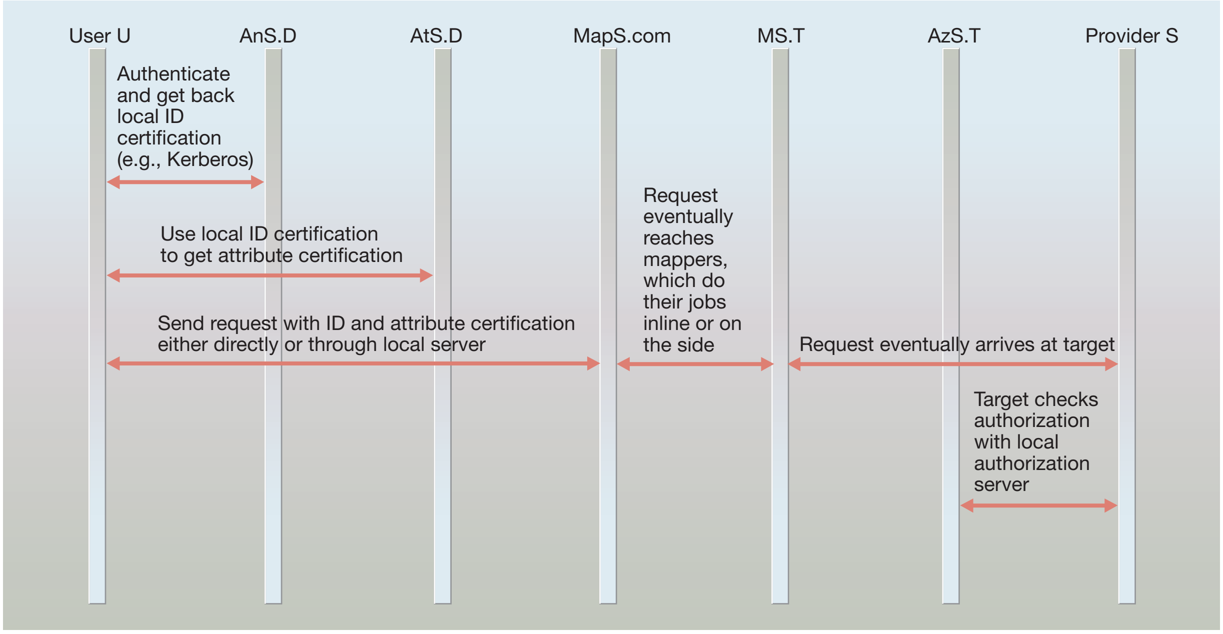
That mapping server does understand the abstract SA syntax produced by the first mapping server and also understands the SA syntax used in the local domain T, so it can perform the necessary translation for that destination. Both of the mapping servers may or may not co-reside with intermediate network nodes or gateways, so they may be invoked inline to the request flow or, as suggested in our example topology, they may sit on the side of intermediate nodes and be invoked by any of several possible different intermediate nodes. Their invocation occurs by using whatever protocols are in force locally, possibly by using SOAP intermediaries if Web or grid services protocols are the local practice.

Eventually MS.T passes the request on to the target server S, which may invoke a local authorization service (AzS.T) to decide whether access should be granted to resource R. This implies that the original user parameters (user ID or group ID or other attributes such as 'a' in this example) must have been mapped somewhere along the path into some local representation that is meaningful and typically found in access control policies of the local resource domain. This is a prerequisite for the elements of SAs coming from the source and target domains to be linkable and reconcilable across the extra-grid VO.

The next set of questions that this scenario raises thus concerns how the local SA syntaxes get mapped into an abstract one and back. More specifically, what mappings need to be created in the intermediate mapping servers, and how do they get initialized there? For a start, we consider the issue of user ID mapping. Mapping other fields of SAs, not covered here, can use similar techniques.

From a WS-Security perspective, the various mapping services referred to above would amount to

Figure 5 IDSA timeline


User U
AnS.D
AtS.D
MapS.com
MS.T
AzS.T
Provider S
Authenticate and get back local ID certification (e.g., Kerberos)
Use local ID certification to get attribute certification
Send request with ID and attribute certification either directly or through local server
Request eventually reaches mappers, which do their jobs inline or on the side
Request eventually arrives at target
Target checks authorization with local authorization server

nothing else but elaborate Secure Token Services (STS) as defined in the WS-Trust portion of that set of architecture documents[9] (which is not yet finalized as of this writing).

User ID mapping within the IDSA abstraction. We assume that an abstract user ID is represented by a URI such as mailto:. In a given VO, it is however unlikely that all user IDs are known at all sites (i.e., in all domains) under that or any other common syntax. As noted in the introduction, alice@example.com is probably unknown at targetex.org, even if there is a VO contract between the two organizations. Thus, any authorization attempt by Alice for resources within targetex.org would fail. To resolve this, somewhere at the domain boundaries, some form of interdomain identity mapping is needed so that something like “alice@example.com” can be mapped into something meaningful in domain targetex.org.

There are several possibilities for this identity mapping (that may be selected by design or by policy):

1. “*Trivial*” *mapping*. User John.Doe (in domain example.com) is mapped as mailto:John.Doe@example.com. This can be used in cases where all VO users are indeed registered as users in every domain within the VO with their full identity. This is unlikely to be practical on a large scale (and in reality may be far from trivial, as noted below, because of the potentially large amount of mapping-table information that would need to be initialized and updated every time a user joins or leaves any of the organizations), but it is a definite possibility in limited-scale extra-grid scenarios. The following three options do not have this disadvantage.
2. Fully *anonymous mapping*. User John.Doe (in domain example.com) is mapped to mailto:anonymous@VO, where anonymous@VO in turn maps into some user ID in the target domain to represent any VO user. Authorization will obey that domain’s policy for VO users but detailed accountability and accounting as well as logging of access requests will be impossible on an individual user basis. However, initialization of the necessary mappings in servers is minimal, as it maps any VO user into the same user ID associated to the VO as a whole.
3. *Domain-accountable mapping.* User John.Doe (in domain example.com) is mapped into mailto:anonymous@example.com, where anonymous@example.com in turn maps into some predefined user ID in the target resource domain to repre-

sent any VO user from example.com. Authorization will follow the target resource domain's policy for users from example.com. Work authorized under the current IDSA can be accounted for and billed to example.com, though access requests under John.Doe's user ID cannot be logged for later tracing back to that user. Initialization of mapping tables in this case is somewhat more involved as it requires potentially n^2 mapping-table entries—one per foreign domain in each local domain.

4. *Group-accountable mapping.* User John.Doe (in admin group of domain example.com) is mapped into mailto:admin@example.com, where admin@example.com in turn maps to a user ID in the target resource domain that represents any VO user from example.com with the admin group or attribute. Authorization will follow the target resource domain's policy for users from admin@example.com. Work authorized under the current IDSA can be accounted for and billed to the admin team at example.com, though it cannot be logged for later tracing back to John Doe personally. Accountability is finer-grained than in options 2 and 3 above, though the amount of mapping-table initialization and maintenance work is somewhat larger.
5. *User-accountable but domain-authorized mapping.* User John.Doe (in domain example.com) is mapped as mailto:XXX@example.com where XXX is a user name synthesized on the fly, which may disguise John Doe's identity at the VO level but allows for separate accounting for work authorized under the current IDSA. Authorization may follow the target domain's policy for users from example.com, as in case 3. It would, however, not be as fine-grained as in option 1. In addition, if the equivalence of XXX and John.Doe is logged, for instance at the entry point into the target resource domain, access requests under user ID XXX can be traced back all the way to John.Doe@example.com himself. Such a design, while limiting the amount of required mapping-table initialization, appears to be a good compromise between granularity of accountability (per user) and granularity of authorization (per domain).

The preceding list of options refers to mapping of user IDs at an interdomain boundary within the IDSA abstraction so that the name of a user as expressed in its home domain, if authorized in the target domain, can be mapped into something meaningful in that target domain.

As suggested previously, option 1 which was referred to as "trivial," is in fact operationally far from trivial. For instance, if the concrete user ID format in the target domain is limited to 8 bytes, the mapping of mailto:John.Doe@example.com to something like JDXMPLCM requires a manual table that links the mailto user ID to an 8-byte user ID, which in practice would be an unacceptable administrative burden for any decent size organization.

Options 2 to 4 present related but much smaller practical problems, as it appears that they can be resolved algorithmically (e.g., by creating single entries such as ANONVO, ANONXMPL, and ADM_XMPL respectively). Note that they provide not only an increasingly fine granularity of access control (by VO, by organization or company, or by group or attribute) but also an increasingly fine grain of accountability for logging and auditing. Option 1 presents fine user-level accountability but at the cost of a potentially huge mapping-table initialization effort for all VO user IDs across all member domains. By comparison, option 5 presents a good compromise between granularity and mapping-table generation effort, as already suggested. It generates locally accountable user IDs on demand and may log their cross-domain meaning upon such automated generation but manages authorization on a per-domain basis.

OGSI services

We believe that the guidelines suggested in this document are fully consistent with the OGSA roadmap[16] and that the basic OGSI services described in that roadmap can make use of local mappings of IDSA contents, even in their latest WSRF version.

A user (or other service requestor) in domain A can obtain the local portion of a security assertion (effectively containing only the first three fields of an IDSA) as a result of local authentication. When authorization to access resources in another domain is sought, the partial security assertion must be relayed to the IDSA mapper of domain A. The mapper will translate the IDSA into some other concrete syntax according to local policy, and relay it to the IDSA mapper of the appropriate remote service domain B, possibly using SOAP secured by WS-Security. Domain B's IDSA mapper can further translate the equivalent security assertion into its local format and relay it as needed in domain B. The abstract IDSAs suggested in this paper need not appear anywhere in practice, but serve as a foundation for deciding on mapping options and guiding translation and initializing of tables at interdomain boundaries.

Conclusion

We have proposed the need for canonical abstract semantics for interdomain security assertions in a heterogeneous extra-grid VO, and we have suggested the fields that are required in such a canonical representation. The specific contribution of this paper is the observation that extra-grid scenarios will require combining elements of assertions under the control of source and target domains, and that this will in turn require suitable mapping mechanisms to build complete assertions from disparate elements. We have suggested using URIs whenever possible to represent canonical SA fields, and have sketched out the issues involved in mapping one important element, i.e., a user identity, in the context of the OGSA.

The proposed approach does, however, leave a number of issues to be addressed. Mapping user identities is a crucial aspect of mapping IDSAs, but it is far from the only one. Similar URI-based mapping of user group references as well as target resource and entitlement references, mapping of authentication mechanisms and contexts, permitted operations and privacy restrictions, and other operations are also required, and the same methods that we have suggested for user identities should be pursued for these other aspects of SA. There is little doubt that this can be accomplished, but possible designs should be explored and evaluated.

Once best practices for mapping IDSAs through URI-based canonical abstract semantics have been defined for each of the typical SA fields, corresponding designs should be implemented and tested. To the extent that these URI-based semantic representations remain purely abstract and never need to be implemented in any actual message flow, realizing a design based on this concept would not require any formal standardization or industry endorsement—it would simply be an implementation choice. However, there remains a significant gap between a conceptual and partially explored proposal such as ours and an actual realization that raises issues of interorganization agreement on some subset of URIs as the semantic basis—issues of on-the-fly, real-time mapping performance, and of course issues of security and trust in the realization of the mapping. Prototyping and hypothesis validation are thus called for.

Finally, while ad hoc interorganization agreements would be sufficient for addressing the issues at stake in extra-grid environments, standardization *would* be required to do the same in inter-grid scenarios. Indeed in such settings, domains must to be able to freely join and leave an inter-grid as desired without having to negotiate any prior agreement on some common URI base for the abstract semantics. Instead, the ad hoc prior agreements that are adequate in an extra-grid need to be replaced by global and standard agreements—and this would of necessity require standardization, not of any message flows but of some URI space where any user or group identity, any resource or entitlement, any authentication method, any permitted operation or privacy restriction, and any other commonly encountered SA field can be mapped into a meaningful and commonly agreeable semantic notion represented by some fixed URI.

The proposal raised in this paper thus opens a wide set of research issues to be explored.

Acknowledgments

We are grateful to our colleagues at IBM for their comments and background contributions, especially Nataraj Nagaratnam and Tony Nadalin, who provided many critical remarks and valuable suggestions for focusing the paper.

*Trademark or registered trademark of International Business Machines Corporation.

**Trademark or registered trademark of The Open Group or the University of Chicago.

Cited references

1. I. Foster, C. Kesselman, J. M. Nick, and S. Tuecke, “The Physiology of the Grid—An Open Grid Services Architecture for Distributed Systems Integration,” Global Grid Forum 5 Document, *Open Grid Services Infrastructure Working Group* (June 2002), http://www.globus.org/research/papers/ogsa.pdf.
2. S. Greenwald, “A New Security Policy for Distributed Resource Management and Access Control,” *Proceedings of the 1996 Workshop on New Security Paradigms*, Lake Arrowhead, CA; ACM, New York (1996) pp. 74–86, http://portal.acm.org/citation.cfm?id=304870.
3. S. De Capitani di Vimercati, and P. Samarati, “Access Control in Federated Systems,” *Proceedings of the 1996 Workshop on New Security Paradigms,* Lake Arrowhead, CA; ACM, New York (1996), pp. 87–99, http://portal.acm.org/citation.cfm?id=304871.
4. D. Kindred and D. Sterne, “Dynamic VPN Communities: Implementation and Experience,” *Proceedings of the DARPA Information Survivability Conference and Exposition (DISCEX II)* (2001) pp. 254–263, http://csdl.computer.org/comp/proceedings/discex/2001/1212/01/12120254abs.htm.
5. R. Baker, L. Gommans, A. McNab, M. Lorch, L. Ramakrishnan, K. Sarkar, and M. R. Thompson, “Conceptual Grid Authorization Framework and Classification,” Global Grid Fo-

rum Working Group on Authorization Frameworks and Mechanisms (February 2003), http://www.ggf.org/Meetings/ggf7/drafts/authz01.pdf.

6. V. Welch, et al., "OGSA Authorization Requirements," Global Grid Forum Open Grid Services Architecture Security Working Group (June 2003), http://www.globus.org/ogsa/Security/authz/OGSA-authorization-requirements-june3.pdf.
7. "Web Services Description Language (WSDL) 1.1," *World Wide Web Consortium* (March 2001), http://www.w3.org/TR/wsdl.
8. "Simple Object Access Protocol (SOAP) 1.1," *World Wide Web Consortium* (May 2000), http://www.w3.org/TR/SOAP/.
9. B. Atkinson, G. Della-Libera, S. Hada, M. Hondo, P. Hallam-Baker, J. Klein, B. LaMacchia, P. Leach, J. Manferdelli, H. Maruyama, A. Nadalin, N. Nagaratnam, H. Prafullchandra, J. Shewchuk, D. Simon, and C. Kaler, "Web Services Security (WS-Security)," *Organization for the Advancement of Structured Information Standards (OASIS)* Web Services Security Technical Committee, April 2002, http://www.ibm.com/developerworks/webservices/library/ws-secure/.
10. J. T. Kohl, B. C. Neuman, and T. Y. T'so, "The Evolution of the Kerberos Authentication System," in *Distributed Open Systems,* F. M. T. Brazier and D. Johansen, Editors, IEEE Computer Society Press (1994) 78–94, ftp://athena-dist.mit.edu/pub/kerberos/doc/krb_evol.lpt.
11. J. T. Kohl, B. C. Neuman, "The Kerberos Network Authentication Service (Version 5)," Internet Engineering Task Force, Network Working Group, Request for Comments RFC-1510 (September 1993), ftp://ftp.rfc-editor.org/in-notes/rfc1510.txt.
12. "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile," Internet Engineering Task Force, Network Working Group, Request for Comments RFC-3280 (April 2002), ftp://ftp.rfc-editor.org/in-notes/rfc3280.txt.
13. S. Farrell, I. Reid, D. Orchard, K. Sankar, S. Godik, H. Lockhart, C. Adams, T. Moses, N. Edwards, J. Pato, M. Chanliau, C. McLaren, P. Mishra, C. Knouse, S. Cantor, D. Platt, J. Hodges, B. Blakley, M. Erdos, and R. L. Morgan, "Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML)," Organization for the Advancement of Structured Information Standards (OASIS) Standard (November 2002), http://www.oasis-open.org/committees/download.php/1371/oasis-sstc-saml-core-1.0.pdf.
14. V. Welch, F. Siebenlist, S. Meder, L. Pearlman, and D. Chadwick, "Use of SAML for OGSA Authorization," Global Grid Forum Open Grid Services Architecture Security Working Group (June 2003), http://www.globus.org/ogsa/Security/authz/OGSA-SAML-authorization-profile-june4.pdf.
15. "XACML 1.0 Specification Set," Organization for the Advancement of Structured Information Standards (OASIS) Standard (February 2003), http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml.
16. F. Siebenlist, V. Welch, S. Tuecke, I. Foster, N. Nagaratnam, P. Janson, J. Dayke, and A. Nadalin, "OGSA Security Roadmap," Global Grid Forum 5 Document, Open Grid Security Architecture Security Working Group (July 2003), http://www.globus.org/ogsa/Security/draft-ggf-ogsa-sec-roadmap-01.doc.

Accepted for publication May 19, 2004.

Brian E. Carpenter *IBM Switzerland, 48 Avenue Giuseppe-Motta, 1211, Geneva 2, Switzerland (brc@zurich.ibm.com).* Dr. Carpenter is an IBM Distinguished Engineer working on Internet standards and technology. He is currently based in Switzerland, working on networking and grid technology. From 1999 to 2001 he was at iCAIR, the international Center for Advanced Internet Research, sponsored by IBM at Northwestern University in Evanston, Illinois. Before joining IBM, he led the networking group at CERN, the European Laboratory for Particle Physics, in Geneva, Switzerland, from 1985 to 1996. This followed ten years' experience in software for process control systems at CERN, which was interrupted by three years teaching undergraduate computer science at Massey University in New Zealand. Dr. Carpenter holds an M.A. degree in natural sciences from Cambridge University, and a Ph.D. degree in computer science from Manchester University. He is a Chartered Engineer (UK) and a member of the IBM Academy of Technology. He is an active participant in the 6NET project, in the Global Grid Forum, and in the Internet Engineering Task Force, where he has worked on IPv6 and on differentiated services. He served from March 1994 to March 2002 on the Internet Architecture Board, which he chaired for five years. He also served as a trustee of the Internet Society and was chairman of its board of trustees for two years until June 2002.

Philippe A. Janson *IBM Zurich Research Lab, Saeumerstrasse 4, Rueschlikon 8803, Zurich, Switzerland (pj@zurich.ibm.com).* Dr. Janson received a B.S. degree in electrical engineering from the University of Brussels and M.S. and Ph.D. degrees in computer science from the Massachusetts Institute of Technology. From 1976 to 1996, he held a tenured lecturer position in operating systems at the University of Brussels. In 1977, he joined the IBM Zurich Research Lab, where he worked initially on high-speed packet switches and the IBM Token Ring. In 1986, he worked on OS/2® LAN gateways at the IBM development lab in Austin, Texas. Back in Zurich in 1987, he managed several projects on heterogeneous networking and security. In 1995, he became head of a new computer science department at the IBM Zurich Lab, which he built up until 1999, with a focus on IT security technologies, smart cards, pervasive computing, and e-business. In 1995, he was elected to the IBM Academy of Technology, of which he was Vice President in 2000 and 2001, serving at the same time as program manager for university relations at the Zurich Lab. In 2001, he also became a member of the advisory board of the communication systems department of the Ecole Polytechnique Fédérale in Lausanne, Switzerland and was elected to the research council of the Swiss National Foundation. Since 2002, he has returned to an active research career as a Senior Technical Staff Member, working on Web services security. He holds a number of patents and has written over 50 papers in the areas of IT security and distributed systems as well as a book on operating systems. He received a Harkness Fellowship in 1972, and a number of IBM Invention and Outstanding Technical Contribution awards since then. He is a member of the ACM and of the IEEE Computer Society.