The Role of Agreements in Virtual Organisations[1]

Babak Sadighi Firozabadi & Marek Sergot

1 This work is supported by the EU FP6 Integrated Project TrustCom.

1 Introduction

Prevention of unauthorised access to data and other computational resources is one of the main topics of research in the field of computer security. Access control is concerned with devising formalisms for specifying precisely and unambiguously the conditions under which access is to be granted — this specification is usually called a 'policy', which is also the term we will employ in this paper — and designing mechanisms for enforcement of these policies, to guarantee that unauthorised access cannot take place and that authorised access is enabled. The existing solutions for access control are usually built upon the following implicit assumptions:

– there is a single line of authority specifying the access policies, which means that there is no conflict of interest, and

– there is a centralised installation and enforcement of the policies.

In recent years there have emerged a number of new technologies such as Peer-to-Peer, Service Oriented Architectures, and Grid Computing. These enable highly dynamic and transient collaborations to be formed among independent enterprises in order to share resources and perform business transactions.

These types of collaboration settings are normally called Virtual Organisations (VO) [FKNT02]. The assumptions regarding access control mentioned above do not necessarily hold for virtual organisations. The enterprises participating in a VO may have conflicting interests, leading to different sets of policies for access and use of the shared resources. Since the enterprises are autonomous and independent, and possibly competitors, there is consequently no guarantee that mutually agreed access policies will be adhered to: members of a VO may fail to, or choose not to, comply with the rules governing access in the VO. If there is no way of practical (physical) enforcement of VO policies then it would be useful to have a normative control mechanism for their soft enforcement, in the sense that VO members who avoid complying with the agreements can still be subject to sanctioning and remedial action as the consequence of their behaviour.

1.1 Existing Access Control Models

Existing access control models are originally designed for distributed applications operating on client-server architectures, that is, computer systems in which there is one central machine, the 'server', providing some kind of computational service, and many other machines, the 'clients', which communicate with the server to access that service. A basic assumption for these architectures is that there is a centrally supervised management of the entire system such that access policies will be updated and enforced as they are prescribed. For example, when a new user is introduced, its identity and its access permissions for each service will be added to the access control lists maintained at the 'server' machine. Given this assumption, the policy enforcement component is trusted always to comply with the prescribed policy (unless it develops faults). The question of what to do when a service provider deliberately fails to comply with the system's policies does not arise.

In contrast, in a system of heterogeneous and independently designed subsystems this assumption no longer holds. Consider this example: agents a1 and a2 are participating in an application with no central enforcement mechanism. a_1 wants to access data d that is stored remotely with agent a2. Upon an access request from a_1, a_2 has to decide whether to grant the access to a_1 or not. There are several possible cases:

– a_1 is permitted to access the resource, but there is no obligation on a_2 to grant that access. a_2 will not violate any policy regardless of whether it grants or denies the access.

– a_1 is not only permitted to access the resource, but is also entitled to it. This means that a_2 has an obligation to grant the access whenever a_1 requests it. A typical scenario is when a_1 is the owner of d and a_2 is the storage service provider for d. Another example is where d is owned by another agent a_3 and a_3 has authorised (or rather, entitled) a_1 to access d on a_2. a_2 violates the policy if it fails to grant access to the entitled agent a_1.

– a_1 has no permission to access d, and so a_2 is forbidden to grant the access. Note that a_2 may have the practical possibility to give access to a_1 even if it is not permitted to do so.

1.2 Entitlement

In the literature on computer security, and in computer science generally, the terms ‘right’ and ‘permission’ (and ‘privilege’, and others) are used interchangeably. We have chosen to use the term ‘entitlement’ to emphasise that we have in mind a concept stronger than mere permission.

Suppose that in a VO there is an agreement that member X makes available 15GB of its disk storage for use by other members (under certain other specified terms which we ignore for the sake of the example). Suppose that X also has its own local policies, to the effect that members from some group (or ‘domain’) D will not be granted access to its resources (because of some previous experiences with domain D, for example), and files containing gif images will not be stored (because of the danger of storing pornographic materials, say). Suppose now that one of the members of the VO, Y, attempts to store a file on X’s disks. X denies the access because the file contains gif images. Would we say that X has thereby violated (failed to comply with) the agreements operative in VO? If the answer is ‘no’ then the agreement is merely that Y has permission to store files on X’s disks. If the answer is ‘yes’, then Y is not only permitted to store files on X’s disks but is entitled to do so. The policy language used to express the VO agreements must be capable of making the distinction.

With this distinction a server might have a local policy to the effect that access to its resource will be granted to any permitted member (including entitled ones) between certain hours, but outside those hours access will be granted only to those who are entitled.

In general, a member X of some VO will be subject to (at least) two separate sets of policies: the agreements operative in VO, and the local policies defined for X. In current approaches such as [PWFK02], it is assumed that the agreement in the VO must always be consistent with the local policy defined for each resource. But how could this be ensured, in a system that is composed of

independently designed sub-units? It is possible to imagine applications where the assumption might hold up, for instance, when all the independent resource providers either formulate their local policies to be consistent with VO agreements, or specify in their own local enforcement mechanisms that in case of conflict between VO agreements and local policies, the VO agreements will take precedence. But such a remarkable degree of co-operation between all the resource providers will not be so common. Rather, it is to be expected that local policies will conflict in certain circumstances with VO agreements, sometimes because the resource provider is looking for a 'free ride', but also because there are some detailed local considerations (such as bad previous experiences with domain D) which would lead a resource provider to choose to violate agreements from time to time. Or suppose that the agreements require that X makes available its disk storage between 6 a.m. and midnight, but X has a local policy which restricts access to the hours of 8 p.m. and 11 p.m. How could this happen? Well, leaving aside the possibility that X is out for a free ride, perhaps when X joined the VO it was expected that accesses outside those hours would be very infrequent and therefore not worth worrying about. It may also be that a resource provider belongs to more than one VO, and finds itself in circumstances where it cannot comply with the VO agreements of both.

Our argument is that when dealing with access control in networks of heterogeneous systems without centralised control, the usual concepts of permission and prohibition are inadequate, and must be extended with (at least) one additional concept which we are calling entitlement. The question is whether a request, e.g., an access request, from an agent creates an obligation on the controlling agent to grant this access. There are then two further subsidiary questions. (1) What if the controlling agent ignores the request and consequently violates its obligation? (2) Under what circumstances may one agent create or pass on entitlements to another?

2 The Approach

Here we summarise an approach we have been developing for sharing resources in a virtual organisation according to a contractual agreement. A more detailed, formal description of this framework is given in [FSSB04]. The idea is that virtual organisation members, which in our case are enterprises, can share computational resources between themselves according to a virtual organisation policy which can be seen as a contractual agreement between the virtual organisation members. We see one member's obligation to provide a resource to another as the second member's right to access/use that resource. As a member of a virtual organisation, an enterprise will gain access to the resources of others and at the same time will have to release its own resources for use by the other members, according to the rules of sharing stated in the agreement. It is a feature of many virtual organisations that these agreements are dynamic, may be very short-lived, and may even be negotiated and created by computational agents without human intervention.

Resources in a virtual organisation are shared but are still managed

independently by their owners. Each member of a virtual organisation has its own local policy which specifies how it intends to grant access to its resources. An enterprise may change the terms of use of its resources, in its local policy, to optimise the resource usage as time passes.

Each enterprise as a member of a virtual organisation must publish a local policy that complies with the agreement of the virtual organisation. This local policy must be available for other virtual organisation members to examine, in order that they may be able to plan how to make use of the shared resources available. It is a separate question whether an enterprise will in fact comply with its own local policies. An enterprise might promise to provide a total amount of resource that exceeds what it can actually deliver. An enterprise may also be a member of several virtual organisations, and it might produce several local policies for its resources, each compliant with the agreement of one of the virtual organisations to which it belongs, but without being able to comply with all the virtual organisation policies at the same time. This is similar to the way flight companies sell tickets to more passengers than they have available seats.

Although a member publishes a local policy specifying how it will make its resources available, it is still possible in practice that it will deny access to its resource upon a request. If a request to access a resource is not granted, although specified as available in the local policy, then the enterprise violates the corresponding obligation in the virtual organisation policy. As a consequence of the violation, the enterprise must usually accept another obligation to be fulfilled, or in the absence of such an obligation, it will violate the entire agreement. The assumption is that members always have an incentive to continue being members of virtual organisations, and hence they will avoid, as far as they are able, any breach of agreement. In the case that an agreement is violated, one can expect that some kind of punitive actions may take place — for instance, expulsion of the defaulter from the virtual organisation. These further considerations however will be outside the framework presented in this paper.

It is important to note that all the member interactions are carried out without centralised control. Thus, a key issue concerns monitoring systems. The framework will provide both monitoring systems and enforcement mechanisms at different stages of the virtual organisation life. First, a mechanism for verifying that local policies satisfy virtual organisation agreements is devised. Then, a level of monitoring is performed for controlling the actual granting of access requests. It is reasonable to assume that if a request is granted then the resource is actually allocated for the requester to use. To make this explicit, the granting of a request can be in the form of a signed (digital) certificate stating that the requesting agent is entitled to access/use the resource, which in our case is the same as allocating the resource for use by the agent. In this way, we factor out of consideration the possibility that a request is granted but the resource or the promised level of quality is subsequently not provided. A system to enforce this would be one which has a central policy enforcer controlling access to the virtual organisation resources, even though these are owned by different virtual organisation members. Note that the resources are released by the policy enforcer, only if the user can show a valid (digital) certificate issued by the resource owner stating its right to use the resource. It is up to the resource owner

to keep track of the use of its resource and it cannot grant access to a resource that is already in use.

3 Open Issues

In order to realise the approach above there are a number of subsidiary issues that need to be addressed. Here we sketch the main requirements.

3.1 Accountability

Any regulative system requires that the entities subject to its norms (its 'policies' in the terminology of this paper) can be uniquely identified in order for them to be accountable for their behaviour.

Ensuring accountability in applications where there is no identity requirement of any kind is not possible. There are such applications but they fall outside the scope of this work. Applications that require a pseudonym identity for their users will give a weak handle for accountability ensurance. The main problem with such applications is that a user can have several identities in the system at the same time. A user can also disappear after a misbehaviour and reappear with another identity. One way of achieving accountability in such systems is to introduce reputation and scoring mechanisms which give the users economic incentive to retain the same identity for a longer period of time, e.g., by increasing the quality of service for those who have been lawful citizens of the virtual organisation for a longer period of time.

Most applications in which the users are involved in some kind of enterprise activity will require that users participate in the business with their real identities. For example, users must often identify themselves with their 'public key certificates', an encryption device which guarantees binding between their user identifiers in the computer system and their legal identities. These systems can still be very dynamic in the sense that the set of users, resources, and the relations between them change frequently.

3.2 Violation Detection and Complaint Procedure

A basic component of the infrastructure is to monitor that services are provided in accordance with the VO agreements and perhaps individual agreements made between members, and to detect violations (non-compliance) as they occur. The monitoring can be active, or it can be left to the members of the VO to initiate complaint proceedings against other members when agreements are unfulfilled. Here, we need to devise protocols, and associated cryptographic mechanisms, for ensuring that proper evidence is collected of both the actions and also the lack of actions of the agents.

There is a need for designing security (cryptographic) protocols to prevent false claims by agents. This is a question of guaranteeing evidence of actions (or lack of actions) on both the requester's and the service provider's side. For example, it should not be possible for a requester to claim without justification

that its request was not answered properly, and it should not be possible for the service provider to claim that it has fulfilled an obligation if it has not done so.

We believe that, as a starting point, existing cryptographic protocols [Sch03] can be adapted for this purpose.

3.3 Sanctioning mechanisms

It is necessary to devise effective sanctioning mechanisms in order both to encourage agents to comply with the rules of the VO and fulfil their obligations, and to provide implementable sanctions in cases where members fail, or choose not, to comply. Sanction mechanisms can be quite simple: temporary suspension of entitlements and privileges, for example, is easily implemented, as is decrease in the level of quality of service provided. More elaborate forms of sanctioning, such as the use of 'marginal accounts' or 'bonds' can also be devised. A systematic exploration of the possibilities and their effectiveness remains to be done.

References

[FKNT02] I. Foster, C. Kesselman, J. Nick, and S. Tuecke. *The physiology of the grid: An open grid services architecture for distributed systems integration*. "www.globus.org/ research/ papers/ogsa.pdf", January 2002.

[FSSB04] B. Sadighi Firozabadi, M. Sergot, A. Squicciarini, and E. Bertino. *A Framework for Contractual Resource Sharing in Coalitions*. In Proceedings of IEEE 5th International Workshop on Policies for Distributed Systems and Networks, pages 117–126, Yorktown Heights, New York, USA, June 2004. Computer Society Press.

[PWFK02] L. Pearlman, V. Welch, I. Foster, and C. Kesselman. *A Community Authorisation Service for Group Collaboration*. In Proceedings of the Third International Workshop on Policies for Distributed Systems and Networks, pages 50–59, Monterey, California, USA, June 2002. IEEE.

[Sch03] Bruce Schnider. *Practical Cryptography*. John Wiley & Sons, 2003.