Chapter 4

Security Policies Categorization for the Framework

This chapter categorizes and explains different types of security policies that have been focused in the implemented framework. The chapter also illustrates the methods for expressing and exposing these policies in the environment.

4.1 Security Policies

The definition of the term policy is often contextual and confusing. Policies are defined, implemented and utilized in a particular context. There can be policies for security, workload, quality of service, networking service, business processes and a multitude of other areas. Generally policies are classified under three levels: business level, domain level and device level [6]. Business level policies are concerned with high level business definitions, domain level policies are concerned with organization level issues and device level policies are concerned with issues related to individual resources. The implemented framework is mainly concerned with security policies that control access to grid services/resources. Some of these policies will operate at domain level and others will operate at device level. This chapter discusses different types of security policies that can exist among different entities in a grid environment. These policies operate at different levels of abstraction and the mechanisms for their expression, evaluation and enforcement are also different. The different policies have been classified into two categories:

i) Access Control Policies
ii) Non-Access Control Policies

Following paragraphs briefly discuss each of these.

4.1.1 Access Control Policies

Access control policies are those policies which play direct role in determining whether a resource/service can be accessed or not. Access control policies are mainly attached with services/resources but can also be applicable to domains and service providers. *e.g.* a domain may have the policy that trust with the requester must be greater than a particular threshold value in order to access any of the services/resources provided by that domain. Similarly a service provider may also have the policy that all the requesters must have X.509 certificates in order to access any of his services/resources. These policies are the example of access control policies applicable to a domain and a service provider. On the other side consider the policies: i) Service "SR-1" is accessible only to subjects of Domain "D" on Mondays from 09:00 am to 05:00 pm only and ii) Any subject of any domain can access the service "SR-2" but only for research purpose. These are the examples of access control policies applicable to a service. The focus of the thesis is on to describe mechanisms to express, evaluate and enforce access control policies associated with services/resources, domains and service providers. Different resources/services can have different number of access control policies associated with them. A subject must satisfy these policies in order to access that service/resource. The access control policies have further been classified into authentication policies, privacy policies, trust policies and authorization policies.

4.1.2 Non Access Control Policies

Non access control policies do not play direct role in determining access to a resource /service rather their nature and scope is different. *e.g.* a policy might state that if subject "S" of domain "D" accesses the service then data generated during access will be stored on hard disk "HD1" but for all other subjects it will be stored on hard disk "HD2". This policy is different from access control policy as enforcement of this policy comes into picture only once we have determined that the service/resource can be accessed. So it cannot be considered as access control policy in true sense. A non access control policy might also state that execution of a grid service cannot take more than one hour on any

machine for subjects of domain “E”. Similarly the policies like: “remove machine “M” from the grid if utilization falls below 50%” and “data stored on hard disk “HD4” will be replicated on hard disks “HD9” and “HD10” on alternate days”, are examples of non access control policies. Compared to access control policies, these policies are different in nature, scope and cover resource usage, operational and business level issues of virtual organizations. These types of policies must be enforced for all applicable access requests which are granted access to the requested service/resource. The mechanisms to express, evaluate and enforce these policies are very much different from those of access control policies. The implemented framework is mainly concerned with access control policies.

4.2 Access Control Policies Categorization

The access control policies have been categorized into authentication related, privacy related, trust related and authorization related policies. Figure 4.1 shows where among different grid entities in a grid environment, the authentication, privacy, trust, and authorization related access control policies can exist.

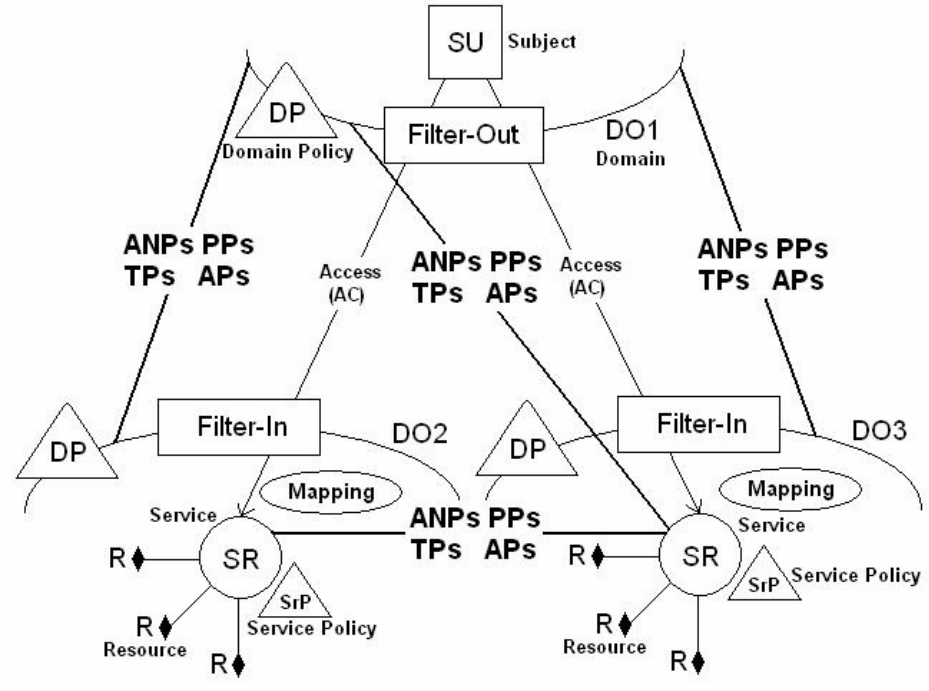

Figure 4.1: Schematic showing authentication, privacy, trust and authorization policies among different entities in a grid environment

In Figure 4.1, SU represents Subject, DO represents Domain, ANP represents Authentication Policy, AP represents Authorization Policy, TP represents Trust Policy, PP represents Privacy Policy, SR represents Service, SrP represents Service Policy and R represents Resource. Every service/resource is associated with Service Policy (SrP), which must be satisfied in order to gain access to that service/resource. SrP may include ANPs, PPs, TPs and APs. In general, ANPs, PPs, TPs and APs can exist among DO↔DO/SP/SU/SR, SP↔SP/SU/SR, SU/SR↔SU/SR and access to SR/R is controlled by SrP. Following sections discuss each of these access control policies.

4.2.1 Authentication Policies (ANPs)

Authentication policies deal with authentication related issues like authentication mechanisms and type of security credentials required by subjects for the invocation of a grid service. Based on authentication policies published by a service, users can take necessary measures to conform themselves to these policies. A grid service may require a particular authentication mechanism for its invocation. In grid environment, authentication mechanisms employed by service providers of different domains can be different. *e.g.* one service provider can use X.509 certificates and others can use Kerberos / username: password / custom security tokens. The services can also demand different security credentials from subjects of different domains based on their roles/trust relationships. *e.g.* a service can demand X.509 certificate from subjects of domain "A" and Kerberos ticket from all other subjects. The authentication policy can also specify whether the service accepts proxy credentials, delegated credentials, custom security tokens or anonymous security tokens *etc.* and if supported then under which conditions or with what constraints? *e.g.* if a service supports single sign-on through proxy certificates then up to which level they are acceptable? A critical service may decide not to accept proxy / delegated credentials from subjects. Using authentication policy, a service can also convey encryption and signature requirements. In grid, services should be able to convey the required authentication mechanisms / algorithms / requirements to different subjects so that subjects can access it. This can be achieved by publishing the authentication policy. Thus mechanisms to express, evaluate and enforce authentication policies are essentially required.

4.2.2 Privacy Policies (PPs)

Privacy policies describe privacy related concerns among service providers and requesters. Service providers may want a service to be accessed only for a particular purpose. They may also want a service to be accessible for a particular purpose by a particular subject with particular credentials. They may also want a service to be accessed for one purpose by subjects of one domain and for another purpose by subjects of another domain. All these are the examples of privacy policies on services. Mechanisms should be there to describe and enforce privacy policies among services and subjects of same/different domains.

Using privacy policies service requesters can also specify the purpose for which they want their private data / information to be accessed. Service requesters can also tell service providers as how they want their personal data / information to be used for a particular purpose. These types of privacy requirements are stored in the database in the form of privacy relationships. Representation of privacy relationships is explained in Section 5.2.1 The privacy relationships are used by authorization framework to provide privacy based access to services/resources.

In the implemented framework, privacy policies applicable to a service/resource have been attached with it through Service Policy (SrP). A subject must conform to privacy policy associated with the service in order to access that service and a service must also respect the privacy policies of a subject if it exposes subject's private data / information to other entities.

4.2.3 Trust Policies (TPs)

Trust policies describe trust related concerns of service providers and requesters. *e.g.* A grid domain may want X.509 certificates to be signed by a particular CA from members of one domain and may not want to accept the same from members of another domain. Subjects may want to access a particular service/resource if the direct or recommended trust with target service/service provider/domain is greater than a specific limit and not otherwise. A service also may require different trustworthiness from subjects of different domains for its access. A service may also want to accept delegated credentials if trust with the subject is greater than a threshold value and not otherwise. All these scenarios

involve trust related concerns and are examples of trust policies. Service providers and requesters may also want to enforce different trust relationships for different contexts. So mechanisms should be there to describe trust relationships and policies among subjects and services of different domains. The representation of trust relationships is explained in Section 5.3.1 while describing trust model.

Like privacy policies, trust policies applicable to a service/resource are associated with it through Service Policy (SrP). A subject must conform to these policies also in order to access that service. The implemented framework makes use of established trust relationships and policies to provide trust based access to grid services/resources.

4.2.4 Authorization Policies (APs)

Authorization policies deal with issues like who can access which services/resources and under what conditions. Access to a particular service/resource is controlled by set of authorization policies (along with authentication, privacy, trust and other types of policies) that are enforced at service provider's end. Authorization policies can be very complex as they may include a number of different types of constraints and conditions to be satisfied before granting access to the requested service/resource. Authorization also depends on the attributes held by subject, service/resource or environment. Authorization policies describe rules regarding which subjects can access which services/resources, when they can access the services/resources, under what conditions they can access the services/resources, what actions they can perform on the services/resources *etc.* Authorization policies may state whether all subjects or only the subjects belonging to a particular domain or specific subjects from specific domains can access the services/resources. Authorization policies applicable to a particular service/resource can be different for subjects of different domains. Authorization policies may also vary with the level of trust and privacy relationships with the target domain/service provider.

The access control policies described above (authentication, privacy, trust and authorization) are related with each other in a complex manner. *e.g.* Trust policies may include privacy policies. Authorization policies may require access to be provided if appropriate privacy and trust policies are enforced. Similarly, authentication may also be governed by policies directed by trust model. A single policy may include different

aspects *e.g.* the policy "Resource R1 is accessible through X.509 certificate from 10 am to 11 am on Wednesday; only for the research purpose and only if the trustworthiness of the subject is greater than 0.8; and if the subject is researcher of ABC domain." describes that authentication, privacy, trust and authorization related aspects can be present in a single policy.

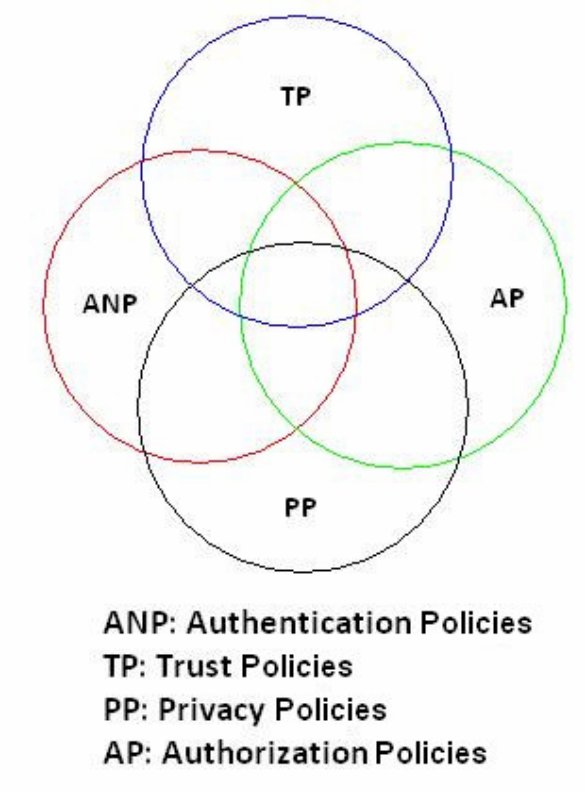

Figure 4.2: Interrelationship among different types of policies

Figure 4.2 gives an indication that authentication, privacy, trust and authorization related aspects can overlap and different combinations of these aspects are possible. The integrated model is an attempt to treat a policy involving these combinations as a unit. The policy specification and evaluation related functionality of authentication, privacy and trust related aspects have been incorporated into integrated policy based authorization model. The integrated model described in Chapter 5 is capable of handling a policy involving different combination of authentication, privacy, trust and authorization related aspects as a single unit. Without integrated model, it is not possible to treat the policy involving combination of authentication, privacy, trust and authorization related aspects as a single unit.

All types of access control policies discussed above are applicable to services/resources, domains and service providers but the mechanisms to associate these policies to services/resources, domains and service-providers are different. The policies applicable to services/resources are associated with them through Service Policy (SrP) and the policies applicable to domains and service providers are associated with them through Domain Policy (DP) and Service Provider Policy (SPP). Next section describes how these policies have been expressed in the security policy framework.

4.3 Policy Expression

In the security policy framework, the access control policies related to authentication, privacy, trust and authorization have been expressed in XACML and are stored in the policy database maintained by every domain. The main reason behind using XACML is that it is the most promising and notable effort to express and enforce general access control policies. It is OASIS standard and provides an extendable and portable way to express access control policies. XACML allow different domains to express and exchange policy information in an interoperable way. To express unique features of privacy and authorization policies, we are making extended use of XACML.

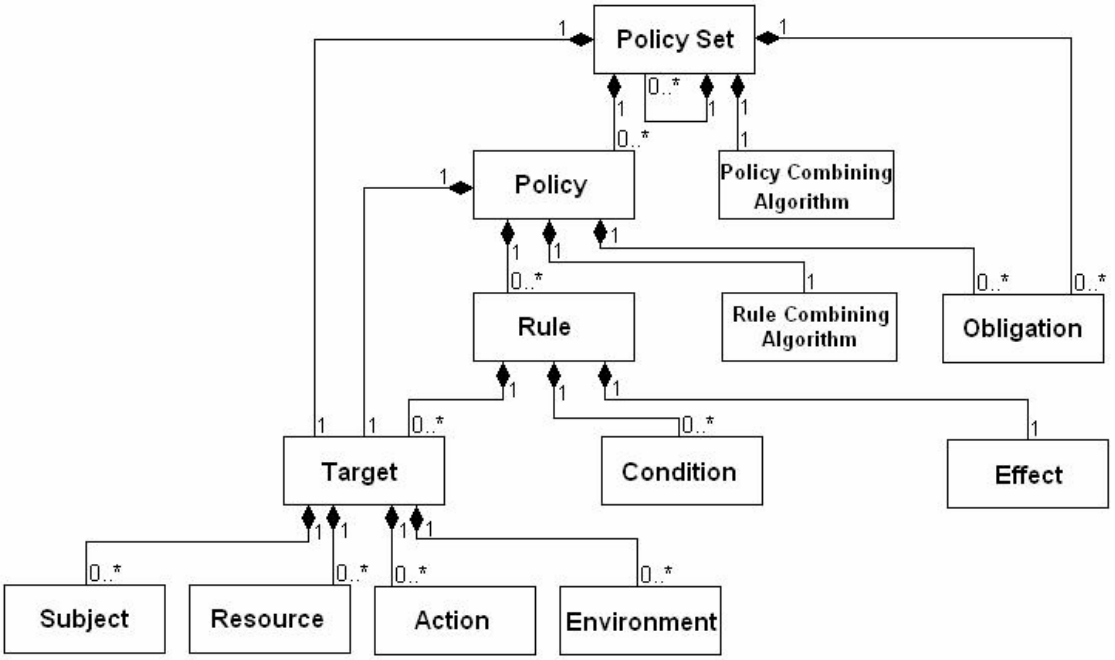

Figure 4.3: XACML policy language model

In XACML, policy is constructed as a set of rules against the target defined as a triod (Subject, Resource, Action). Figure 4.3 shows XACML policy language model and the relationship of its different elements with each other. The main elements of policy language model are: Rule, Policy and Policy Set. Following paragraphs briefly discuss each of these elements.

A Rule is the most elementary unit of policy. A Rule is a defined as "A target, an effect and a set of conditions". A target is defined as the space of decision requests that refer to actions on resources by subjects. Target helps in determining whether or not a rule is relevant for a request. An effect is either permit or deny, and is the intended consequence of the satisfied rule. Conditions are statements about attributes that upon evaluation returns either true, false or indeterminate. The conditions involve the calculation of attributes of the subject, the resource, the action or the environment. Conditions make the rule dynamic [41]. Multiple rules can be associated to a policy.

A Policy consists of a set of rules, a rule combining algorithm, a target and a set of obligations. A rule combining algorithm specifies the procedure by which the results of evaluating the component rules are combined when evaluating the policy. Obligations are the actions that must be performed by the enforcement system in conjunction with the enforcement of an authorization decision. A Policy Set consists of a set of policies, a policy combining algorithm, a target and a set of obligations. A policy combining algorithm specifies the procedure by which the results of evaluating the component policies are combined when evaluating the policy set. The obligations are the operations specified in the policy set that should be performed by the enforcement system in conjunction with the enforcement of an authorization decision [41].

XACML specification also defines six rule-combining and policy-combining algorithms namely: Deny-overrides, Ordered-deny-overrides, Permit-overrides, Ordered-permit-overrides, First-applicable and Only-one-applicable. In Deny-overrides, if any rule evaluates to deny, then the final authorization decision is also deny. Ordered-deny-overrides is same as Deny-overrides, except the order in which relevant rules are evaluated is the same as the order in which they are added in the policy. In Permit-overrides, if any rule evaluates to permit, then the final authorization decision is also permit. Ordered-permit-overrides is same as Permit-overrides, except the order in which relevant rules are evaluated is the same as the order in which they are added in the policy. In First-applicable, the result of the first relevant rule encountered is the final authorization decision. In Only-one-applicable, the result of the only one applicable

policy is the final authorization decision. Figure 4.4 shows the skeleton of an access control policy written in XACML.

```
⋮
<Policy ... PolicyId = "" ... RuleCombiningAlgoId= " " ...   >
      ⋮
      <Target>
      ⋮
      </Target>
      ⋮
      <Rule ... RuleId = "" ... Effect = "" ... >
            ⋮
            <Target>
                  ⋮
                  <Subject>
                  ⋮
                  </Subject>
                  ⋮
                  <Resource>
                  ⋮
                  </Resource>
                  ⋮
                  <Action>
                  ⋮
                  </Action>
                  ⋮
                  <Environment>
                  ⋮
                  </Environment>
                  ⋮
            </Target>
            ⋮
            <Condition>
                  ⋮
            </Condition>
                  ⋮
      </Rule>
      ⋮
      <Obligation>
            ⋮
      </Obligation>
      ⋮
</Policy>
⋮
```

Figure 4.4: Skeleton of policy written in XACML format

In the implemented framework, all types of access control policies have been expressed in XACML but to incorporate authentication, privacy and trust based access control information, we are making use of its extension capabilities. To express privacy policies, "<purpose>" element has been introduced that will specify the purpose for which the service/resource can be accessed. Another addition that has been made is the inclusion of "<context>" element that will specify the context associated with a particular access request. Any of the access control policy (authentication, privacy, trust and policy)

can make use of this element if required. To express rest of the policies, we do not require any addition to standard XACML specification available. Figure 4.5 shows the placement of "<purpose>" and "<context>" elements in standard XACML policy.

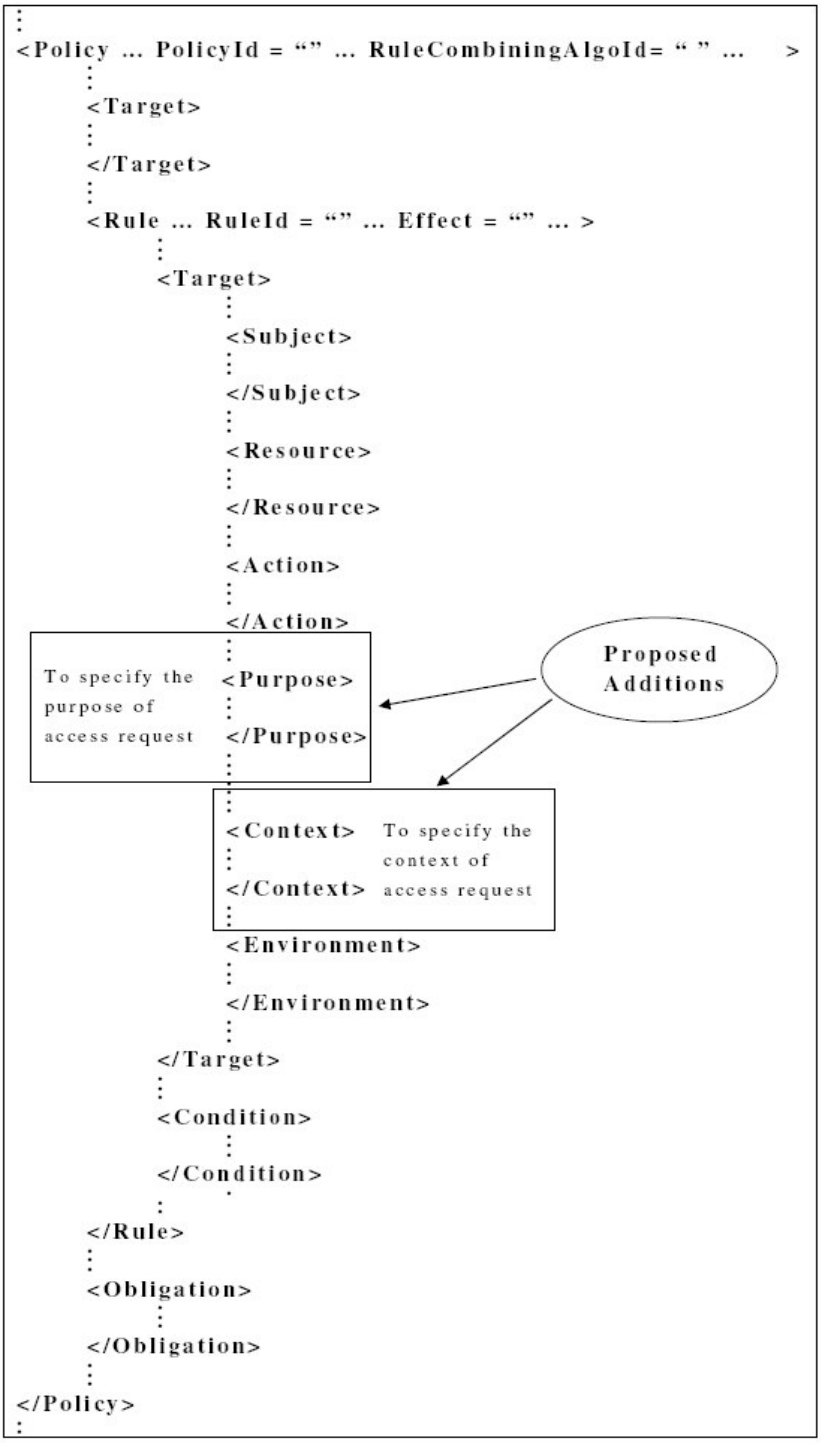

Figure 4.5: Skeleton of policy representing purpose and context attributes in XACML format

Figure 4.6 shows the skeleton of service policy file, which we are using in the security policy framework, to represent authentication, privacy, trust and authorization related access control policies applicable to a service/resource. Note that this file is different from XACML policy file but refer to other XACML policies through "Id" tag of "Policy" element. This file does not represent the complete set of policies applicable to that service/resource. As policies vary from subjects to subjects, the target element of other policies in the database is checked for the applicability of those policies to an access request.

```
⋮
<ServicePolicy ... PolicyId = "" ... PolicyCombiningAlgo= " " ...>
      ⋮
      <AuthenticationPolicy>
            ⋮
            <Policy ... Id = "" ...>
                  ⋮
            </Policy>
            ⋮
      </AuthenticationPolicy>
      ⋮
      <PrivacyPolicy>
            ⋮
            <Policy ... Id = "" ...>
                  ⋮
            </Policy>
            ⋮
      </PrivacyPolicy>
      ⋮
      <TrustPolicy>
            ⋮
            <Policy ... Id = "" ...>
                  ⋮
            </Policy>
            ⋮
      </TrustPolicy>
      ⋮
      <AuthorizationPolicy>
            ⋮
            <Policy ... Id = "" ...>
                  ⋮
            </Policy>
            ⋮
      </AuthorizationPolicy>
      ⋮
</ServicePolicy>
⋮
```

Figure 4.6: Skeleton of policy file associated with a service/resource

The other policies that describe requirements of a service like the types of security credentials accepted by service *i.e.* whether the service accepts X.509 certificates / Kerberos Tickets / Custom Security Token / Anonymous Security Token / username: password pair or whether the service is hidden service / private service / accepts anonymous access or whether the service provide trust based access *etc.* have been expressed in simple XML. These policies are attached to service/resource through service description file *i.e.* WSDL using customized "PolicyReference" element so that these requirements can be found while discovering service itself.

This chapter is mainly concerned with the expression mechanisms of different types of policies and not with their evaluation and enforcement. Next chapter presents the security policy framework along with its different models. The framework can be used to address key security requirements related to authentication, privacy, trust, authorization and to provide policy based access to grid services/resources. The chapter also describes the evaluation and enforcement of access control policies discussed in this chapter.