Applied Mechanics and Materials
ISSN: 1662-7482, Vols. 63-64, pp 59-63
doi:10.4028/www.scientific.net/AMM.63-64.59

Online: 2011-06-08

A study of push-based sequence authorization mechanism for grid security

Youchan Zhu [a] and Juanhua Bai [b]

Network Management Center, North China Electric Power University, Baoding,071003,Hebei Province, China

[a]ZYC_hd@sina.com, [b]crazying2009@163.com

Key words: grid authorization, authorization sequence, architecture, CAS.

Abstract. Grid security is one of the core issues of grid computing, and the authorization issue of grid security is a hot topic of current research. This paper describes three sequence patterns of the authorization, and focuses on the architecture of push sequence model and application example-Community Authorization Service (CAS). It analyzes and compares the push sequence pattern and pull sequence pattern. Authorization mechanism based on push sequence pattern solves the scalability issue of GSI quite well in some extent and improves the flexibility of the system.

1 Introduction

Grid computing integrates large, heterogeneous and wide-area distributed network resources, making network into a virtual "super computer". It provides services with certain QoS, and achieves resource sharing and task coordination. Because of the complexity of grid resources and a huge grid user base, a grid computing must consider and solve the problem that how to guarantee legitimate users can legally use the grid resources and the information of users and resources can be securely transferred in a grid. So safety has become one of the most important factors in grid environment, it is the basic guarantee for a grid to achieve its goal. In the grid security system, the authorization is a vital part of the process. It allows users and applications to access the grid resources effectively and protects the grid resources from unauthorized using.

2 Authorization mechanism

Authorization identifies and controls the permissions of users accessing to resources and services, through evaluating the identity of customers and security strategies, and according to reasonable authorized strategies determines and controls that the clients can use what resources under what conditions. Authorized process generally involves three basic entities: Subject, Resource and Authority. Subject: entities which request, receive, acquire, transfer, cancel, or entrust electronic authorization to exercise certain authority; it generally refers to a user or a process on behalf of a user. Resource: system components which provide services or hosts where services exist can execute access to services based on a set of rules and policies defined by entities. Authority: a management entity which can produce, validate and cancel an electronic certificate; it can perform certain permissions or state certain attributes. There are often three types of permission: one is attribute permissions. According to different subjects, it has different attribute values; the second is strategy permissions. According to different resources and services it releases authorized strategies; the last one is identity permissions, for example, certificate is a kind representative of identity permissions.

2.1 Authorization sequence patterns

Interaction between the entities can be defined as three sequences or patterns based on the different order of message routing in authorization process [1, 2], that is push sequence pattern; pull sequence pattern and agent sequence pattern. The authorization system and service provider (resource) belong to different trust region in the next diagrams.

(1) Push sequence pattern

Authorization system exists as a service, it release authorized decision in the form of authorized statement, as shown in figure 1. Subject first requests for authorized permissions to authorization center, and then authorization center decides whether to grant access permissions to the subject. The authorized permissions have certain effective time. After getting the authorized certificate, the subject submits the certificate directly to the target resource for services. The target resource responds to the subject a license after evaluation and creates a safe environment to establish secure session with the subject.

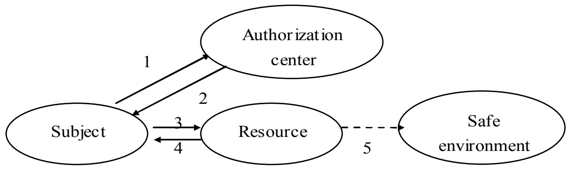

Fig.1 Push Sequece Pattern

(2) Pull sequence pattern

Authorized systems usually connect with applications or services and act as the strategy decision makers of them, as shown in figure 2.

Subject first request for services to the target resource, then the authorized services of the target resource contacts with the authorization center of subject domain to decide whether to grant the request. After getting the reply from the, The Target resource respond to the subject a license and create a safe environment to establish secure session with the subject.

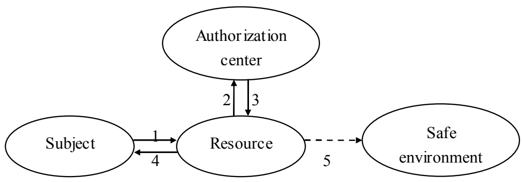

Fig 2 Pull Sequece pattern

(3) Agent sequence pattern

Authorization system exists as a service, as shown in figure 3. Subject first requests for authorized permissions to authorization center, then the authorization system request for services and provide decision-making information to the target resource as the agent of the subject. After getting the reply of it, the agent returns a license to the subject and creates a safe environment to establish secure session with the subject.

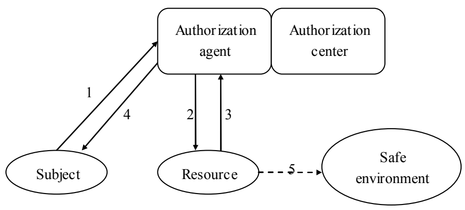

Fig.3 Agent Sequece pattern

In actual grid authorization system, the two models above sometimes are mixed to be authorized mixing sequences.

2.2 Authorization architecture

This paper mainly introduces the push sequence pattern based authorization architecture, as shown in figure 4. It includes authorization centers which release and decide authorized attributes, parameters, strategies. Functional components can define authorized conditions according to authorize authorized attributes, parameters and strategies, and make authorized decision. Authorization architecture is consisted of these functional components and entities above [1].

Authorization architecture introduced two access control functions. One is the Access control Decision-making Function (ADF): make authorized decision whether subjects can access services. It is equivalent to Policy Decision Point (PDP) defined in RFC2904, and usually as a functional component of the authorized server, independent of resources and applications. The other is Access control Execute function (AEF): make decision whether subjects can access services, it is equivalent to Policy Enforcement Point (PEP) defined in RFC2904, general located before its protected resources.

Obviously, ADF, AEF, subject and resource might belong to the same or different management domain; there can be a variety of combinations. No matter what kind of situation, there are three kinds of information transferred between subjects, resources and different authorization centers, that is: attributes, strategies and search and response of authorization.

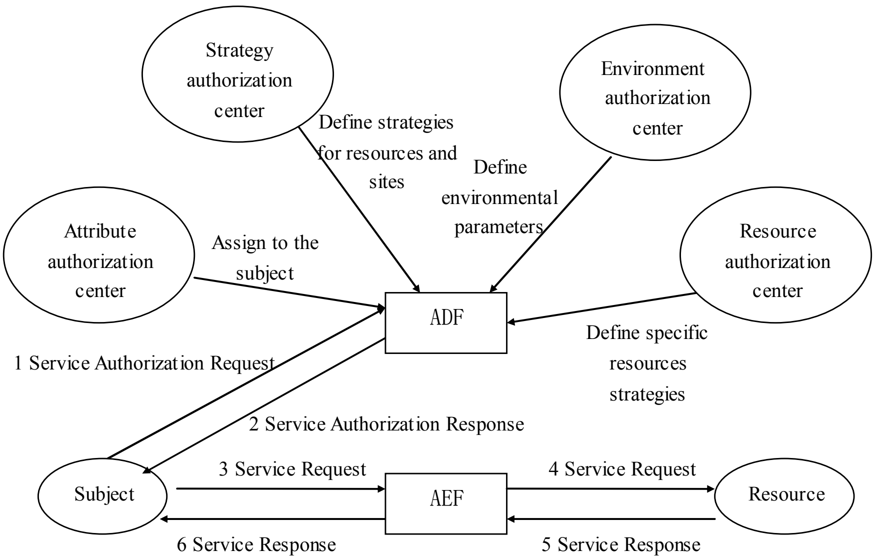

Fig.4 Authorization Architecture Based on Push Sequece

In the authorization structure based on push sequence, the subject sends service authorized request to the decision-making function; the decision-making function decide whether to grant permissions according to the data provided by each authorization center; if pass, the decision-making function returns a access services authorized token to the subject; then the subject submits the token to the execute function for it to make a decision whether the subject can access services after uniform certification and authorization research in the grid environment.

2.3 Community authorization service(CAS)

Grid Security Infrastructure(GSI) project is Globus's security infrastructure components package. It is an integrated security solution to solve the security problem of grid computing. Due to the grid heterogeneous autonomy, in order to support the local security mechanism of different sites, In GSI, authorization can be finished in local place using direct trust relationship between resource providers and resource consumers to describe the authorized strategies. GSI provides a mapping mechanism. When the user applies for services, its identity will be mapped to a local identity. It authorizes user's permissions through the local security strategies. The mapping mechanism of GSI requires each resource server to maintain a large and clumsy global/local mapping table. It is hard to expand to large-scale grid environments which have lots of resources and large users[4].And users and resources are dynamic variable, such authorization mechanism is not convenient to the whole system expands, reducing the flexibility of the system. The third party organizations authorized service-CAS [5] is introduced into GSI. It is responsible for managing user's access strategies, no longer mapping the user's identity in grid environment to a local identity.

CAS is a role based group authorization technology built in public key authentication and agent mechanism. The groups can be all or parts of the users of a VO, such as the internal machines in the same LAN belong to a group. A group is considered as a whole. With a group of GSI awarded certificates, it can start a CAS service and authorize a group certificate to it. A group is the bridge between users and resource servers. There is a table maintained in the CAS server of the group. It lists the roles of all members of the group and access strategies. At the same time, the group gets the authorization from each resource serve, and its content is stored in the local database of the resource server.

CAS releases global strategies as a strategy service center in the virtual organization. It includes: access control strategies of resources in the virtual organizations, member list and access control strategies of CAS server.

CAS system adopts the typical push sequence pattern, and its authorized service mechanism process as shown in figure 5.

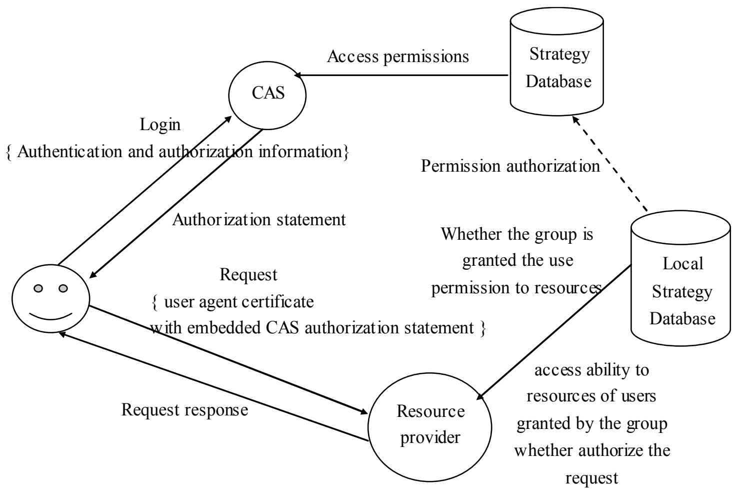

Fig.5 Authorized Service Mechanism Process

When a user use resources, it Certifies itself to CAS server with his own generated proxy certificate, and put forward the resource use request; once through the authentication server, CAS server determine whether the user can obtain the resources capability according to the organization strategies, and authorizes it to the user as form of authorization declaration; later the user request to the resource server and show access permissions obtained from CAS server; the resource server will validate the group access permissions according to local access control strategies and verify the user access permissions whether conform to the resource request; then determine whether to make service response. CAS gives a good solution to the scalability issue of GSI and improves the flexibility of the system.

3 Authorization sequence patterns analysis

In the push sequence authorization mode, the user provides authorization information to resource providers; the local policy engine from local interprets the authorization information, and makes decisions. It makes accurate decisions and provides efficient service, but local resources need to install unified policy engines, not easy to maintain and upgrade.

In the pull sequence authorization mode, the resources are required to obtain authorized decisions by querying strategy service centers; the center policy engine interprets all the authorization information. It is easy to extend resource services, maintain and upgrade. However, in the case of a large number of resource access, it will cause communication bottleneck, affecting the overall system performance.

Obviously, the two models have advantages and disadvantages, and their effective integration will make up the current system deficiencies.

There four classic Authorization systems in the current: AS and the virtual organization member service (VOMS) use the push sequence model; Akenti system and Permis system use the pull sequence model. Based on policy service center, they all have a central server, which is responsible for creating, maintaining and verifying identity, groups and roles. The advantages are the highly concentrated control, the single point management, and the flexible strategy markup language; the disadvantage is that in the case of heavier loads it will affect the overall system performance and easily become a communication bottleneck. However, using this pattern meets strategic needs and lack of GSI in a large-scale grid environment.

4 Conclusion

Grid security is important in the grid application. The grid would not have been large-scale practical application, until the security is guaranteed. There are lots of resources in a grid. it must be able to effectively protect these resources and prevent the use of unauthorized users, or irrational use of authorized users. It is difficult to apply the traditional GSI authorization in the growing and complex grid environment. This article describes three types of authorized sequence patterns, and focuses on the push sequence architecture and application examples, which meet the deficiencies of the GSI. For different grid systems, we can use the appropriate authority sequence pattern according to their specific needs, and develop practical authorization mechanisms.

Reference

[1] Markus Lorch, Bob Cowles, Rich Baker, et al.Conceptual Grid Authorization Framework and Classification.Global Grid Forum, 2004.

[2] J.Vollbrecht, P.Calhou, S.Farrell, et al.RFC 2904: AAA Authorization Framework.2000.

[3] Feng Liyuan.Research on Authorization of Grid Security Systems based on GSI[D] Shanghai Jiaotong University,2007.

[4] PANG Yan-guang, SUN Gong-xing, PEI Er-ming, MA Nan. A study of authorization architectures for grid security[J]Nuelear Electronics & Detection Teehnology,2006

[5] Pearlman L,Kesselman C, Welch V, et al. The Community Authorization Service: Status and Future.CHEP03,2003:24-28