Global Journal of Computer Science and Technology
Network, Web & Security
Volume 13 Issue 4 Version 1.0 Year 2013
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

A Secured Model for Resource Access in Grid Environment

By Dr. Gulshan Ahuja
Tilak Raj Chadha Institute of Management & Technology, India

Abstract - Grid computing provides a way to execute applications over autonomous, distributed and heterogeneous nodes. The main goal of grid technology is to allow sharing of resources and services under a set of rules and policies, which govern the conditions for access to the resources. This paper reviews the state of security and access control for resources in grid environment and presents a secured model for resource access in grid environment.

Keywords : grid computing, services, resource access.

GJCST-E Classification : H.3.5

A Secured Model for Resource Access in Grid Environment

Dr. Gulshan Ahuja

Abstract - Grid computing provides a way to execute applications over autonomous, distributed and heterogeneous nodes. The main goal of grid technology is to allow sharing of resources and services under a set of rules and policies, which govern the conditions for access to the resources. This paper reviews the state of security and access control for resources in grid environment and presents a secured model for resource access in grid environment.

Keywords : grid computing, services, resource access.

I. Introduction

Grid computing aspires to integrate technology and solutions, which enable and control access to computing resources. These resources are generally located at diverse locations and very little information regarding their exact location is known. The need to share resources, coupled with distributed and heterogeneous nature of the web environment, entails the formation of virtual organization.

A virtual organization is defined as a set of individuals and institutions, sharing resources and services, under mutually decided and agreed set of rules and policies. The resources to be accessed are not only limited to file sharing, but expand to a wide spectrum such as computers, software data and other resources as are required by a range of collaborative problem solving and resource brokering strategies.

In a virtual organization setup, individuals and institutions agree to share resources and collaborate on an adhoc dynamic basis, where each real organization is governed by its own set of internal rules and policies. The virtual organization poses challenges such as interoperability among domains, need to maintain separation of the security policies etc.

Security of grid services is a fundamental requirement behind any grid security model. Securing web services consists of providing security services such as authentication, confidentiality, integrity etc. to the exchanged messages. A security model to secure grid services must ensure that grid services when invoked by a service requester adhere to policy constraints, as specified by the hosting environment.

A no. of security standards for web services have been proposed as shown in Figure 1.1. Ensuring the integrity, confidentiality and security of grid services through the application of a comprehensive security model is critical, for both the organization and their customers. This is done using web services. Web services [1] provide an architecture that has the ability to deliver integrated, interoperable, solutions. Web services are loosely coupled applications, which use well known XML protocols like WSDL [2], SOAP [3] and UDDI [4] for representation and communicating across different security domains in the distributed environment.

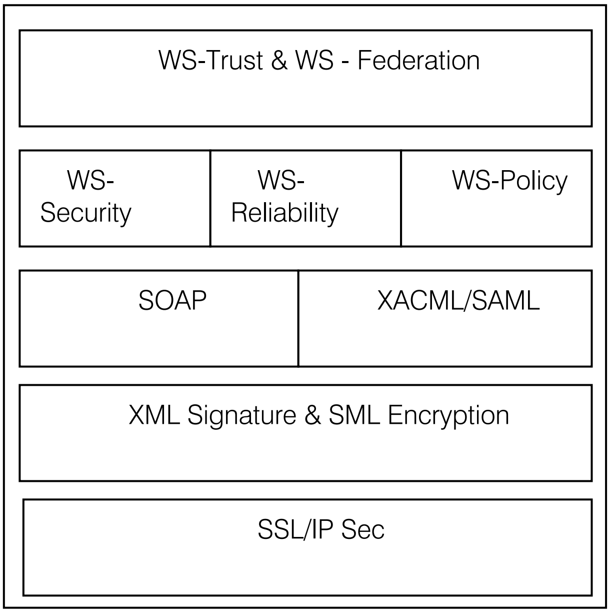


Figure 1.1 : Web Services Security Standards

Compared with the existing distributed object technologies such as DCOM, CORBA and J2EE, web services are opening and loose coupled. Web services create new security challenges because XML documents are encoded in text, rather than in binary form and can readily be transmitted through standard firewalls. These aspects make web services security more troubling and difficult. This paper presents a secured model for resource access in grid Environment. The security of the proposed approach has been improved by adopting secure certificates and use of users' attributes.

The remainder of the paper is organized as follows. Section II discusses the related work in the field of grid technology and web services. Section III presents details about web services security specifications. Section IV presents the problem statement. Section V presents a secured model for resource access in grid environment and discusses its working. Section VI concludes the paper and brings out future scope of work.

Author : Assistant Professor, Tilak Raj Chadha Institute of Management & Technology, Yamuna Nagar, India.
E-mail : ahujag_24@yahoo.com

II. Related Work

Over the past several years there has been a lot of work towards the development of grid technology. A good survey in this direction can be found in [5]. Most of the grid management systems provide various grid services such as security, data management, remote execution and monitoring. The grid applications pack different components in to a single package such as in case of Globus toolkit, Grid FTP for data management, GRAM for execution management and MDS for information service are packed together. The separation of service component provides flexibility in terms of selection of services but makes sharing of grid resources and authentication more complex.

An active grid at Grid Laboratory of Winsconsin [6], presented as a campus wide distributed computing environment, which was designed to meet the scientific computing needs of the university. It was built from autonomous sites from across the campus, which engineered to meet their own specific requirements and cooperated to join with the other sites. Natraj et al. [7] presented a comprehensive grid security architecture, which supported popular security models. Foster et al. [8, 9] presented an analysis of the unique security requirements of large scale distributed computing and a secure architecture. Damiani et al. [10] presented a fine grained access control model for SOAP e services.

III. Web Services Security Specification

Web services security specification (WSS) [11] allows protecting SOAP messages with XML security. WSS provides confidentiality using XML encryption and integrity using XML signature. An XML signature [12] provides integrity, authenticity and non repudiation by enabling entities to sign an entire XML document or some part of this document. An XML signature is an XML document containing information about the signing process, references to the signed parts and the signature value. To process an XML signature, the sender generates a digest for each referenced part before calculating the digital signature value using the specified algorithms. Then the signed XML message is formed by incorporating the signature value, the different digests and information about used algorithms and keys. This allows the recipient to proceed to the validation of this signature. XML encryption [13] provides confidentiality by allowing the encryption of XML data. The result of encryption is an XML document containing information about the encryption process and the encrypted data or reference to this data. The encryption of XML data requires the selection of an algorithm and a key that will be transmitted to the receiver of the message. Then data is serialized before using the chosen algorithm and key. Finally, the message to transmit is formed by adding the encrypted data or reference to this data.

WSS provides SOAP messages with security by using XML signature to sign a SOAP message and transmit the signature and XML encryption to encrypt the message. WSS transmits security information in the headers of SOAP messages, such as keys and security tokens that represent the identities and can be associated to digital signature in order to ensure authentication of the message origin. To secure a SOAP message, WSS denies security headers. In fact, the header of a SOAP message can contain one or more security headers where each of them provides security information on this message to a recipient that can be final or intermediate recipients. To sign one or more elements in a SOAP messages, the security header, added by the sender includes a signature, which conforms to that specified by XML signature. The recipient of the SOAP message proceeds to the validation of the signature. In case if validation fails, a fault message is delivered otherwise the signature is validated and a confirmation is sent to the sender in the header of the response message. To encrypt one or more elements of a SOAP message, the security header must include references to the encrypted elements and information about the used key. Then each element to encrypt is replaced by the equivalent encrypted data. The recipient of a SOAP message identifies the decryption key and the element to decrypt.

Thereafter, each encrypted element is decrypted. Encryption and decryption are performed according to XML encryption.

IV. Problem Statement

To provide security in the grid environment, a number of security implementation software's are available. These implementations use different techniques, protocols and tools for securing resource access in grid environment. GSI is the most common among these methods, which has been implemented in Globus Toolkit 4. GSI uses public key infrastructure (PKI) for encryption and decryption of data, secure socket layer (SSl) for authentication of entity, message integrity & message privacy, X509 public key certificate for delegation of rights etc.

There are middleware, which provide single sign on such as Microsoft's Passport and VeriSign. etc., but these cannot be used in the grid environment due to the following reasons. Passport uses a centralized server to provide authentication. It supports only user name and password method and does not support latest methods like delegation and proxy certificates. The services provided by these middleware agents are paid and a service requester must register with all service providers before utilizing services located at different sites. Moreover, these methods are not suitable for securing resource access in a grid environment. To address the above said problems, this paper proposes a certificate and attributes based model for secured resource access in a grid environment.

V. Proposed Model

Figure 1.2 depicts the detailed view of the proposed model. It mainly comprises of four entities such as (a) Client Requester (b) Service Provider (c) Certificate Authority (d) Attribute Authority.

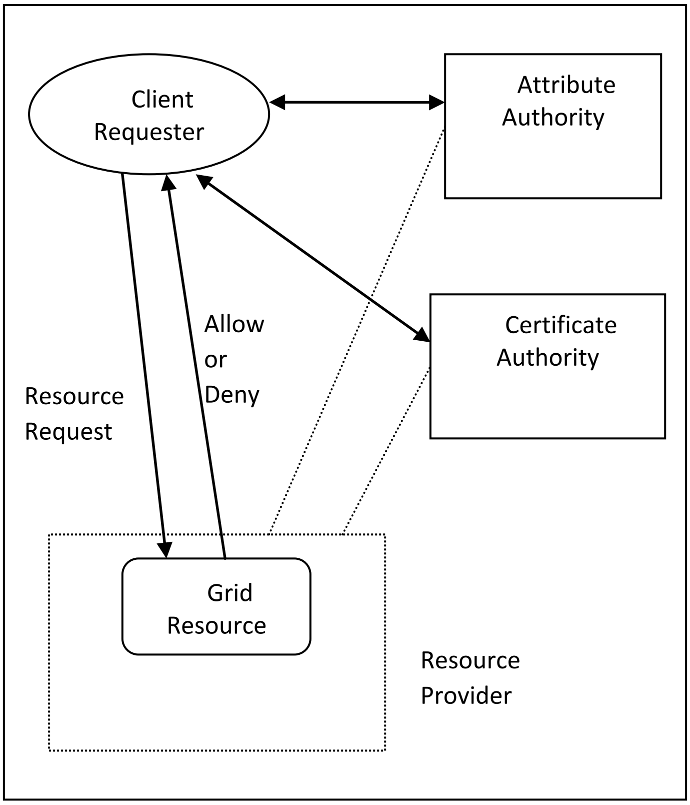


Figure 1.2 : Access Model for Resource Access

A detailed description of each entity is given below:

a) Client Requester (CR)

The CR a requester side component, which is responsible for obtaining digital certificate(s) from the CA, storing in requester's machine, obtaining required attributes from the AA and producing certificates and attributes as and when required for resource access.

b) Resource Provider (RP)

The RP controls the access top the requested resources and verifies the authenticity and authorization details corresponding to a resource request.

c) Certificate Authority (CA)

The CA is a server in the domain, which generates the X.509 certificates, which are used by the grid users and the RP for implementing security. The primary responsibilities of CA are to identify entities, which require certificates, Issuance, removal and archiving certificates and to maintain a name space of unique names for certificate owners. A certificate is represented as a data structure containing public key and pertinent details about the owner of the key. A certificate works as a tamper proof electronic document once signed by the certificate authority for use with the grid environment. A digital certificate contains information about the host who is being certified and its public key.

When a user wants to access a resource in a grid, he attaches a certificate to the request message. On receiving the message, the RP verifies the signature of the certificate within the certificate. After verification, the RP can safely accept the public key contained in the certificate,

d) Attribute Authority (AA)

An AA is an authority who is trusted by user to create and sign an attribute certificate (AC) on his behalf. The time and validity requirements for ACs allow creating ACs, which are long-lived and short-lived. The short validity period may be in number of hours instead of months or years as is the case is identity certificates. The longer-lived attribute certificates are issued where the authorization information is going to remain static for a long period. One more reason to use attribute certificates to contain and specify authorization information is for easily changing the authorization information without making any side effect.

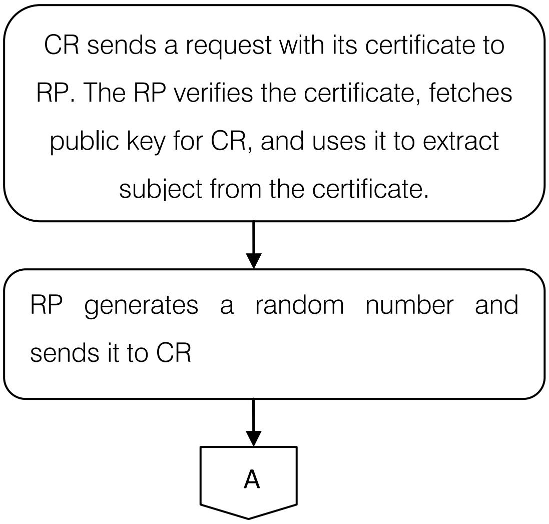
e) Working of the Proposed Model

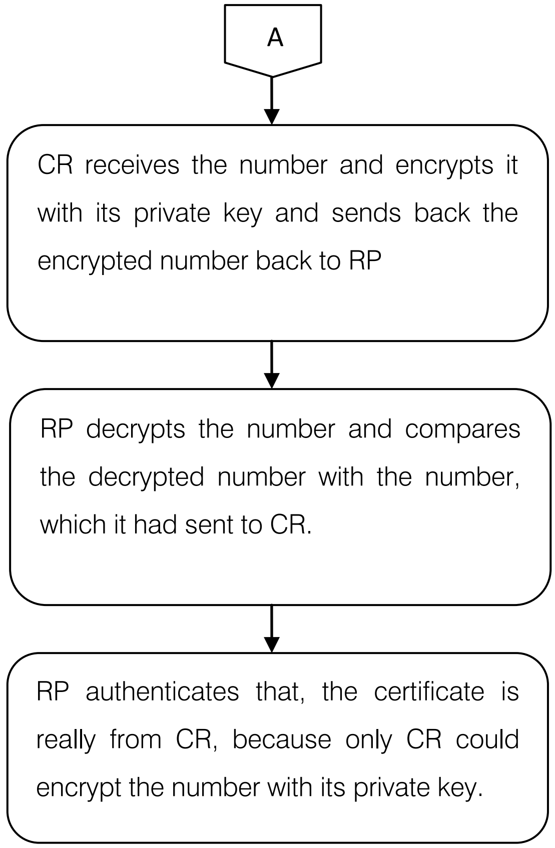
The step by step working of the proposed model is as follows.

Step 1: CR registers with the CA to obtain digital certificate and registers with the AA for requesters' attributes.

Step 2: CR sends a request to RP for accessing a resource. This request contains the digital certificate and requester's attributes.

Step 3: RP carries the authentication of request using procedure as depicted in Figure 1.3.


CR sends a request with its certificate to RP. The RP verifies the certificate, fetches public key for CR, and uses it to extract subject from the certificate.
RP generates a random number and sends it to CR
A



CR receives the number and encrypts it with its private key and sends back the encrypted number back to RP

RP decrypts the number and compares the decrypted number with the number, which it had sent to CR.

RP authenticates that, the certificate is really from CR, because only CR could encrypt the number with its private key.

Figure 1.3 : Authentication Process

Step 4: CR sends an attribute query message to AA and receives a response for requester's attributes. The request message and response message is sent using SOAP handler.

Step 5: RP evaluates the obtained attributes as per algorithm as shown in Figure 1.4. Each accessible resource in the grid environment is assigned with a set of applicable policies. For each policy, a set of constraints specify the conditions, which must be satisfied for positive evaluation of the applicable policies. Each policy is evaluated against set of constraints to find out whether the resource access request can be granted or not.

```
Algorithm: EvaluateAttributes(Input:

RR_ID // Identifier of the
Requested Resource
AS = ($A_1$, $A_2$, $A_3$..............$A_n$ )
// Submitted Attribute Set

PS = ($P_1$, $P_2$, $P_3$.....$P_n$) //Policy Set
CS // Constraints Set
Output: Allow/Deny)

        for all Pi in PS
          If (Pi _RRID == RR_ID)
            for each $C_i$ in CS
                evaluate Ci  against AS for
                outcome
                  if(outcome == false)
                        return Deny
                end if
            end loop
        end if
      end loop
                  return Allow
```

Figure 1.4 : Algorithm for Evaluation of Attributes

Step 6: The outcome of the algorithm is used to allow or deny access to the requested resource.

The use of requesters' attributes allows more robust access mechanism to be placed for resource access.

VI. Conclusion & Future Scope

This paper has presented a secured model for resource access in grid environment. The approach makes use of digital certificate and requesters' attributes for making decisions about resource access. The digital certificate is used to authenticate the user and requester's attributes are further used with an identified and requested resource type. The evaluation of attributes is carried by associating a set of constraints and policies with the resources. The paper has highlighted that how web services technology allows complex integration of technology and protocols to allow resource access.

References Références Referencias

1. Curbera F., Nagy W. A., Weerawarana S., “Web Services: Why and How”, Workshop on Object-Oriented Web Services, October 2001.
2. Chinnici, R., M. Gudgin, J. Moreau, and S. Weerawarana, “Web Services Description Language (WSDL)”, version 1.2, World Wide Web Consortium (W3C), http://www.w3.org/TR/wsdl12, July 2002.
3. Box, D., “Simple Object Access Protocol (SOAP) 1.1”, “http://www.w3.org/TR/SOAP”, May 2000.
4. Luc Clement et al., “OASIS UDDI Specifications”, http://www.oasis-open.org/committees/uddi-spec/doc/tcspecs.htm, 2002-2004.
5. “A Taxonomy and Survey of Grid Resource Management Systems for Distributed Computing”, Software Practice and Experience, pp. 135-163, 2002.
6. http://www.cs.wisc.edu/condoe/glow/index.html
7. Natraj Nagaratnam, Phillippe Janson, John Dayka, Anthony Nadalin, Frank Siebenlist, Ian Foster, ”Security Architecture for Open Grid Services”, Public Draft, July 2002.
8. Foster I., Kesselman C., “A Secure Architecture for Computational Grids”, 5th ACM Conference on Computer and Communications Security, 1998.
9. Foster I., Kesselman C, “The Grid Blueprint for a new Computing Infrastructure”, 1999.
10. E. Damiani, S.D.C di Vimercati, S. Paraboschi, and P. Samarati, “Fine grained access control for SOAP e-services”, In Proceedings of 10th International Conference on World Wide Web (WWW), Hong Kong, pp. 504–513, 2001.
11. Nadalin et al., “Web Services Security Specification 1.1”, OASIS Standard Specification, OASIS Committee, February 2006.
12. Mark Bartel, John Boyer, Barb Fox, Brian LaMacchia, Ed Simon, “XML Signature Syntax and Processing”, W3C Recommendation, June 2008.
13. “XML Encryption Syntax and Processing”, W3C Recommendation, Dec. 2002.

This page is intentionally left blank