Secure Grid Computing

Jianmin Zhu and Dr. Bhavani Thuraisingham,
University of Texas at Dallas

Abstract
Since late 1990s, Grid computing has become an increasingly important research topic within computer science. Grid computing is concerned how to share and coordinated use diverse resources in distributed environments. The dynamic and multi-institutional nature of these environments introduces challenging security issues, which include integration with existing systems and technologies, interoperability with different "hosting environments" and trust relationships among interacting hosting environments. We need new technical approaches to handle those security issues. During those years, many prominent companies and research institutes have proposed and implemented several architectures for grid and grid security. In this survey, first, we introduce the Globus Toolkit, some commercial Grid productions and Grid Testbeds. Second we describe several Grid security architectures and research methods of security issues from research institutes and Universities. Next we discuss the application of grid computing to the Global Information Grid (GIG). Finally we give some potential research topics..

Key words:
Grid Computing, Grid Security, OGSA, GSI and Web Services.

1. Introduction

Grid computing is about several processors distributed globally and sharing the computational resources to solve various problems. The major issues associated with grid computing is coordinating resource sharing as well as problem solving in dynamic, multi-institutional virtual organizations (VOs) [1, 2, 3 and 4]. Olderdistributed computing technologies cannot address this problem very well. Because grid computing technologies focus on dynamic, cross-virtual organizational resource sharing, it complements rather than competes with existing distributed computing technologies.

In 1998, Ian Foster and Carl Kesselman defined Grid in the book "The Grid: Blueprint for a New Computing Infrastructure": "A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities [1]." After two years, Ian Foster refined the definition: Grid computing is concerned with "coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organization." A set of individuals and/or institutions defined by some sharing rules form what we call a virtual organization (VO) [2]. In order to determine whether a system is a Grid, Ian Foster defined a Grid Checklist in 2002. It is a Grid if it is satisfied with all following features [3]:

- Coordinates resources that are not subject to centralized control
- Using standard, open, general-purpose protocols and interfaces
- To deliver nontrivial qualities of service.

The cluster management systems are not Grid, such as Platform's Load Sharing Facility and Veridian's Portable Batch System, because they are centralize control of the hosts that they manage.

In order to have a basic concept of Grid, we can compare grid architecture with Internet protocol architecture [2]. The Grid architecture is composed of following layers:

Fabric Layer: Provides the local services of a resource

Connective Layer: Core communication and authentication protocols

Resource Layer : Enables resource sharing

Collective Layer : Coordinates interactions across multiple resources

Application Layer: User applications use collective, resource, and connective layers to perform grid operations in a virtual organization

The paper is organized as follows: Section 2 introduces the Globus toolkit. Section 3 describes some commercial grid productions. Section 4 presents Testbeds for grid computing. Section 5 details basic concept of grid security and related research on grid security. Finally, section 6 summarizes the paper and provides directions.

Manuscript received August 25, 2006.
Manuscript revised August 30 , 2006.

2. Globus: A Toolkit for Grid Computing

The open source Globus [29] Toolkit is used for building Grid systems and applications. It is being developed by Globus Alliance, which includes some institutes, companies and Universities. More and more research institutes and companies are using Globus Toolkit to develop relative Grid projects. The Globus Toolkit is a set of software services that solve common problems when building distributed system services.

The latest version is GT4 (Globus Toolkit Version 4). GT software services address the basic issues relating to resource management, workflow management, file management, security and communication, and so on. Those software services is packaged and can be used either independently or together to develop applications and web services. They are deployed to support the development many Globus deployments, such as TeraGrid, EUGrid, China Grid, APgrid, etc). GT4 includes following core software services:

Web Services Components: Community Authorization, Data Replication, Grid Tele control Protocol, **OSGA**-DAJ, WS Core for Python, C and Java programming language, Grid Resource Allocation & Management (**GRAM**)

Non-Web Services Components: GridFTP, Replica Location, C Common Libraries and resource discovery

The Globus Resource Allocation Manager (GRAM) is the lowest level of Globus resource management architecture. GRAM provides a Web service interface for initiating submitting, monitoring, executing and terminating jobs on remote computer. GRAM is responsible for parsing and processing the Resource Specification Language (RSL) specifications enabling remote monitoring and managing of jobs already created.

The Open Grid Services Architecture (OGSA) is the most important architecture in grid computing, which is based on Web services concepts and technologies [29]. OGSA defines a set of core capabilities and policies which most Grid systems mainly concerns, such as how to establish security authentication? How to deploy and discover Web services? The objectives of OGSA are to [4]:

Managing resources across distributed heterogeneous environments. How to manage distributed resources is the most important concern of Grid systems. OGSA provides services for deploying, discovering and monitoring the logical or physical resources across different distributed heterogeneous environments.

Delivering seamless quality of service (QoS): Because of the dynamic nature of grid resources, it is very important to provide seamless and high quality of service, such as authorization, access control, single logon, distributed workflow, problem determination services, resource management performance and delegation.

Providing a common base for autonomic management solutions. In the Grid environment, many combinations of configurations will be used to management the different grid resources. A common management solution for these resources is necessary.

Defining open, published interfaces. For interoperability of different grid environments and with existing systems and technologies, grid architecture must be built on standard interfaces and protocols.

Exploiting industry standard integration technologies. The foundation of OGSA is Web services. How to leverage existing industry solution is very important issue.

Grid system is a service-oriented architecture, Web service is the basic concept of OGSA. The term Web services describes an important distributed computing paradigm. From technique point of view, web services are different from other distributed computing approaches, such as DCE, CORBA. OGSA in architecture is given in [4]. Web services standards were defined within W3C and some major industry companies, such as Microsoft, IBM and SUN. SOAP [52], WSDL [53] and WS-Inspection [54] are the most important standards.

The Simple Object Access Protocol (SOAP) : SOAP provides a communication method between a service request and provider. SOAP envelop can be delivered on HTTP, FTP or HTTPS.

The Web Services Description Language: WSDL is an XML document to describe the web services, such as the service type, the functions of the service.

WS-Inspection: WS-Inspection is composed of a simple XML language and related service descriptions, which is a URL to a WSDL document.

The four main layers comprise the OGSA architecture are the following:

- Grid applications
- OGSA architected services
- Web services, plus the OGSI extensions that define grid services
- Physical and logical resources layer

3. Commercial Grid Productions

From business point of view, Grid computing is becoming a new IT architecture that produces high performance and lower cost enterprise information systems. With the helping of grid computing, more independent enterprise resources, such as hardware and software components, can be rejoined and connected dynamically to meet the requirements of businesses. Grid style systems can deliver a higher quality of service, a lower cost, with great flexibility and higher performance. We claim some of prominent companies are investing in grid computing.

3.1 Oracle

Oracle's grid [9, 10, 11, 12, 13 and 14] is composed by Oracle infrastructure, database, and application products. Its goal is to provide the customers with one clustered grid architecture. Oracle builds many components into its grid architecture, such as enterprise resource planning (ERP), customer relationship management (CRM), and supply chain management (SCM) business applications, a database, J2EE-based middleware, development tools, as well as various application/database/systems/storage management products.

Oracle's 10g grid infrastructure is composed of all of its software products. Oracle Database 10g, Oracle Application Server 10g, and Oracle Enterprise Manager 10g together provide the first complete grid infrastructure software.

Oracle Database 10g is the first database designed for Grid Computing. It builds on the success of Oracle9i Database, and adds some new specific grid capabilities, such as Real Application Clusters, Automatic Storage Management, Information Provisioning and Self-Managing Database.

Oracle Application Server 10g provides a complete grid infrastructure framework to develop and deploy enterprise applications, such as web services and EJB. The enterprise applications can run on low cost storage devices and computer server with very scalability, availability and high performance [10, 12].

Oracle Enterprise Manager 10g Grid Control is the central management system and can automatically manages computer tasks across sets of systems in the distributed environments. Grid Control system can reduce administration costs by automatic management and policy-based standardization [10, 12, and 14].

3.2 SUN

Sun and Oracle are working partners and they are establishing a complete grid solution. For example, Solaris 10 operating system is used in the Oracle Database 10g. Sun provides scalable and dynamic grid infrastructure, which virtualizes computational hardware and software resources.

With the helping of Sun Grid Solution, enterprise customers can rapidly develop and deploy a proven grid architecture, which can address some specific business [15, 16, 17 and18].

The Grid reference architecture from Sun provides a framework. All computer resources can be integrated into a powerful grid system. The reference architecture includes following hardware and software components [17 and 19]:

Hardware components:

- Sun Fire V20z Compute Grid Rack System
- Sun Fire V440 server (File Server)
- Sun StorEdge 3510
- Sun StorEdge 6120 Array
- Myrinet, Foundry switches
- Sun StorEdge 5310 NAS
- Software components:
- Solaris 9 Operating System
- Red Hat Linux EL3.0
- Suse Linux Enterprise Server 8
- Pro9
- Sun N1 Grid Engine 6
- Sun StorEdge Performance Suite (Sun QFS 4.0)
- Sun Control Station (SCS) w/SGEEE Grid Manager
- ROCKS cluster deployment tools
- Ganglia cluster monitoring tools

- N1 System Manager

3.3 IBM

IBM is one of the major contributors to key technical standards being developed in the Global Grid Forum (GGF). IBM mainly focuses on standards, packaging, homogeneity , compute and data grids, and professional services delivery. By contrast, Oracles does not focus on heterogeneous grid.

The grid solution from IBM is to comprise all of its virtualization capabilities (such as those available on IBM servers and storage) into a dynamic services-oriented infrastructure (SOI) to support service-oriented architecture (SOA) application environments [20, 21 and 22]

Both Oracle and IBM have the same design goal to provide a SOA-based infrastructure that can access virtualized resources from different domains. Oracle focuses on Applications and Database Management, however IBM mainly focuses on the Management of Grid Infrastructure and Resources.

3.4 Microsoft

Microsoft provides the service Grid and the whole structure is based one Web services. Web services concepts and technologies is used to implement a services-oriented architecture. The DataGrid is congruent to Microsoft's .NET strategy.

3.5 Grid Testbed

One of the most challenging tasks of Grid computing is to create a real prototype or even production Grid, which requires maintenance of both the technological and political architecture. After setting up a new grid, we need a testbed to test and monitor the status of computing resources. In following section, we will introduce some Grid testbeds.

3.5 DOE Science Grid Testbed

The Grid project of DOE Science [27] started since August 2001. Its goal is to define, integrate, deploy, and develop Grid services in a large scalable, robust, lower cost, high performance Grid infrastructure.

The DOE Science Grid (DOE SG) provides software services for job management, security, fault tolerance, resource discovery, resource access, system monitoring to advanced scientific applications and problem solving frameworks [27 and 28]. The DOE Science Grid is developing following software Services.

- Science Portals
- Grid Information Services (GIS)
- Certification Authority (CA)
- Deploy Globus on Computing Resources.
- Grid Tertiary Storage
- Security Infrastructure
- Auditing and Fault Monitoring
- User Services
- Grid System Administration Tools
- Grid User Access to Resources

3.6 GridLab Testbed

The GridLab Testbed is a European grid architecture, which is comprised of heterogeneous machines from many academic and research institutions [24]. The GridLab Testbed includes following components:

Grid Portals: The portal of GridLab is called GridSphere. GridSphere is an open source Web portal.

Grid Resource Management System (GRMS): GRMS allows developers to build and deploy resource management systems for large scale distributed computing infrastructures

Grid Security: The GridLab Testbed mainly focuses on the definition of common security policies. Currently, most of Grid system is lack the mechanism to define and implement the security policies. This shortcoming results the complexity of open Grid security architecture. GridLab provides GridLab Authorization Service (GAS) to handle the security issues.

Grid Monitoring: Grid monitoring services can satisfy the all requirements of performance monitoring.

Grid Data and Visualization Services: These services provide a frame work to visualize the dynamic streaming data in Grid environment.

Grid Information Services (iGrid): The iGrid has both system information providers and user information providers.

3.7 ATLAS Grid Testbed

The goal of U.S. ATLAS Grid testbed is to let every collaborator of distributed computing infrastructure for ATLAS can access distributed resources and perform data analysis from their home institution, no matter how far they are from CERN geographically [25].
ATLAS Grid is an important Grid testbed in U.S. The membership of ATLAS includes Argonne National Laboratory, Boston University, Brookhaven National Laboratory, Indiana University, Lawrence Berkeley National Laboratory, University of Michigan, Oklahoma University and .University of Texas at Arlington.

The ATLAS Grid Testbed is composed by following software services:

- GRIDView - web based grid monitoring tool
- Magda - grid data manager
- Pacman - package manager
- Pippy - pacman information provider for MDS
- Grappa - grid portal
- GridExpert - expert knowledge database

4. Grid Testbed

One of the most challenging tasks of Grid computing is to create a real prototype or even production Grid, which requires maintenance of both the technological and political architecture. After setting up a new grid, we need a testbed to test and monitor the status of computing resources. In following section, we will introduce some Grid testbeds.

4.1 DOE Science Grid Testbed

The Grid project of DOE Science [27] started since August 2001. Its goal is to define, integrate, deploy, and develop Grid services in a large scalable, robust, lower cost, high performance Grid infrastructure.

The DOE Science Grid (DOE SG) provides software services for job management, security, fault tolerance, resource discovery, resource access, system monitoring to advanced scientific applications and problem solving frameworks [27 and 28]. The DOE Science Grid is developing following software Services.

- Science Portals
- Grid Information Services (GIS)
- Certification Authority (CA)
- Deploy Globus on Computing Resources.
- Grid Tertiary Storage
- Security Infrastructure
- Auditing and Fault Monitoring
- User Services
- Grid System Administration Tools
- Grid User Access to Resources

4.2 GridLab Testbed

The GridLab Testbed is a European grid architecture, which is comprised of heterogeneous machines from many academic and research institutions [24]. The GridLab Testbed includes following components:
Grid Portals: The portal of GridLab is called GridSphere. GridSphere is an open source Web portal.
Grid Resource Management System (GRMS): GRMS allows developers to build and deploy resource management systems for large scale distributed computing infrastructures
Grid Security: The GridLab Testbed mainly focuses on the definition of common security policies. Currently, most of Grid system is lack the mechanism to define and implement the security policies. This shortcoming results the complexity of open Grid security architecture. GridLab provides GridLab Authorization Service (GAS) to handle the security issues.
Grid Monitoring: Grid monitoring services can satisfy the all requirements of performance monitoring.
Grid Data and Visualization Services: These services provide a frame work to visualize the dynamic streaming data in Grid environment.
Grid Information Services (iGrid): The iGrid has both system information providers and user information providers.

4.3 ATLAS Grid Testbed

The goal of U.S. ATLAS Grid testbed is to let every collaborator of distributed computing infrastructure for ATLAS can access distributed resources and perform data analysis from their home institution, no matter how far they are from CERN geographically [25].
ATLAS Grid is an important Grid testbed in U.S. The membership of ATLAS includes Argonne National Laboratory, Boston University, Brookhaven National Laboratory, Indiana University, Lawrence Berkeley National Laboratory, University of Michigan, Oklahoma University and .University of Texas at Arlington.

The ATLAS Grid Testbed is composed by following software services:

- GRIDView - web based grid monitoring tool
- Magda - grid data manager
- Pacman - package manager
- Pippy - pacman information provider for MDS
- Grappa - grid portal
- GridExpert - expert knowledge database

5. Grid Security

5.1 Overview

Security is one of the important issues that usually arise when considering a grid environment. Since the goal of grid is resources sharing, computer resources will be accessed by a lot of different virtual organizations (VOs). The security requirements are fundamental to the Grid security design. The high level grid security requirements include following aspects [5 and 21]:
Authentication: Providing interfaces to plug-in different authentication mechanisms and means to convey the mechanism used
Authorization: Ability to control access to grid components based on authorization policies.
Delegation: Providing mechanisms to allow delegation of access rights from requesters to services while ensuring that the access rights delegated are restricted to the tasks intended to be performed within policy restrictions.
Message integrity: Ensuring unauthorized changes made to message content or data can be detected at the recipient end.
Single logon: This refers to relieving an authenticated entity from re-authentication for a certain period of time when subsequent access to grid resources are requested while taking multiple security domains and identity mappings into account.
Confidentiality: Protecting confidentiality of underlying transport and message content and between OGSA-compliant components in either point-to-point or store and forward mechanisms
Privacy: Allowing both a service requester and a service provider to define and enforce privacy policies.
Policy exchange: Allowing security context negotiation mechanism between service requesters and service providers based on security policy information
Credential life span and renewal: Ability to refresh requester credentials if a grid application operation takes longer to complete than the life-span of a delegated credential
Secure logging: Providing a foundation for non-repudiation and auditing that enables all services to time-stamp and log various types of information without interruption or information alteration by adverse agents.
Assurance: Providing means to qualify the security assurance level that can be expected of a hosting environment. The security assurance level indicates the types of security services provided by an environment. This information is useful in deciding whether to deploy a service in the environment.
Manageability: This requirement mainly deals with various security service management issues such as identity management, policy management, and so on.
Firewall traversal: Ability to traverse firewalls without compromising local control of firewall policy to enable cross-domain grid computing environment
Securing the OGSA infrastructure: This refers to securing core OGSA components. The security challenges faced in a Grid environment can be grouped into three categories [5, 21 and 33]:

Integration with existing systems and technologies: Interoperability with different "hosting environments"
Trust relationships among interacting hosting environments. Because of the dynamic and multi-institutional nature of the grid environments, we need new technical approaches to solve security problem.

5.2 Grid Security Infrastructure (GSI)

In order to overcome the security challenges, Globus proposes the Grid Security Infrastructure (GSI) [29]. GSI is composed of a set of command-line tools to manage certificates, and a set of GSS-API to easily integrate security into other web services. GSI offers the following functions [33];
Transport-level Security
Message-level security (WS-Security and WS-Secure Conversation)
Authentication through X.509 digital certificates
Several authorization schemes
Credential delegation and single sign-on
Different levels of security: container, service, and resource
GSI has satisfied basic Grid security requirements, such as authorization, delegation, authentication and message protection.
GSI provides two levels security: Message-level Security and Transport-level Security. Our research survey will focus on message-level interoperability. For both levels, we have server and client side security.

5.3 Security architecture for the Open Grid Services Architecture (OGSA)

By delivering integrated and interoperable solutions, web services architecture has the ability to overcome security challenges. When designing the security of a Grid environment, we take into account the lots of security aspects involved in a Grid service invocation. The Grid security model includes all following components [5 and 33]:
Application-specific components

- Secure conversation
- Credential and identity translation
- Access control enforcement
- Audit and non-repudiation

Policies and rules components

- Authorization
- Privacy
- Identity/credential mapping
- Service/end-point.

In OGSA architecture, application-specific components depend on policies and rules components. In order to use and manage these services policies, a service language is needed to express and exchange policies. Some methods also are needed to execute secure communication through transport protocols binding. These management components are also subject to policy enforcement [31].
In order to establish the trust relationship between requestor's domain and service provider's domain, they should open a secure conversation channel.
Based on above Grid security model, IBM and Microsoft proposed a WS-Security specification [35]. The security specification will make customers to easily build interoperable secure Web services. The proposed specification is summarized below [35]:
Initial Specifications include WS Security, Policy and Trust.

WS-Security: To describes how to attach signature and encryption headers to SOAP messages.
WS-Policy: To describes the capabilities and constraints of the security (and other business) policies on intermediaries and endpoints
WS-Trust: To describes a framework for trust models that enables Web services to securely interoperate.
WS-Privacy: To describes a model for how Web services and requesters state subject privacy preferences and organizational privacy practice statements.
Follow-On Specifications
WS-SecureConversation: To describes how to manage and authenticate message exchanges between parties.
WS-Federation: To describes how to manage and broker the trust relationships in a heterogeneous federated environment.
WS-Authorization: To describes how to manage authorization data and authorization policies.

5.4 Grid Security Research

In pervious section, we have introduced three challenges of Grid security. We will mainly focus on interoperability challenge in this survey. If we consider a VO as a policy domain overlay, we can address part of challenges [33]. A virtual organization policy domain overlay pulls together participants from disparate domains into a common trust domain. Some VOs are separate domain and they have different security policies. By considering a VO as a policy domain overlay, those domains can become one trust domain.

In order to implement dynamic policy overlays, a Grid security model should provide three key functions:

Multiple Security Mechanisms

One organization can join a VO. The security mechanism of the organization may be different with the security mechanism of VO. Grid security model should interoperate with exist security mechanisms and infrastructure rather than replace them.

Dynamic Creation of Services

Because of the dynamic nature of Grid system, users should have the ability to create a new software service dynamically.

Dynamic Establishment of Trust Domains

The trust relationship management is a very important service in Grid system. The trust relationship is dynamically established. At time t, the two VOs have trust relationship, but they are not trustable in time k. These trust domains can have multiple organizations and must can dynamically join and leave the VO.

In Grid computing, Grid services traverse among multiple domains and hosting environments. In order to make them interact with each other, we should provide interoperability. Usually, Grid Architecture provides interoperability in three levels: protocol level, policy level and identity level. Here we list detail information about those levels [5]:

- Protocol level allows domains to exchange messages. We can use SOAP/HTTP.
- Policy level to make conversation policy among different domains
- Identity level to identify a user from one domain in another domain

5.4.1 Grid portal for interoperability

Computing portal [39] is a very important service to make that web services can seamless access the resources of other domains. Through the portal service, we can uniform access to remote domain's computational resources, such as hardware, software and data.

A full functional computing portal should provide following services [37 and 42]:

Security service will provide all security functions relative with accessing the grid resources.

Discovery service will lookup and registration the usable and available service, we can use UDDI to implementation.

Account management service to manage user's account and track action of users.

Job management service allows a user to monitor and track the job status from the batch scheduler. It also provides function to compose the core services.

File management service to manage the user directory such as copy, rename, move, and so on.

Command execution Service will execute commands of users

Security service in computing portal is critical, since it often involves directly accessing remote resources through delegation to a middle tier proxy. We can use web service mechanism to implement the security of portal service. But how to add security to web service message is a big research problem.

Indiana University [37] proposed a secure web services for computational portal. They implemented a message-level security system using web services security language and WS-Security, which based on GSI from Globus Grid Forum. Because we use SOAP message to communicate between client and server, we can use the assertion based security such as SAML, WS-security into SOAP messages. The assertion based security is adding additional metadata to the correct handler classes.

The most important job of security service is the communication between client and server. After receiving SOAP message from client, the server

of portal service will do security check, such as authentication and authorization. The prototype authentication system of Indian University has both client-side and server-side process.

When considering the problem of inter-grid interoperability, the basic idea is how to define the minimal set of Grid services. If we get the minimal set, any resource requests can be expressed by those elements in the minimal set. Based on the minimal set, we can do some translation between different resource description mechanisms. For examples, Grid A sends request to Grid B. Even Grid B does not understand this content of description, because they come from the same set, we can translate the request to understandable one.

GRIP [44] project solves the interoperability problem between UNICORE Grid and Globus. Users can submit jobs to Globus from UNICORE. They define the minimal set of Grid services:

- Authentication Services
- Ability to compose an object in RR (Resource Request) space
- Ability to instantiate an RR object as an object in RP (Resource Provider) space
- Ability to discover resources
- Notification of commencement

Many universities and research institutes have developed lots of portal service for grid computing, such as GENIUS [43], World Grid [45], NPACI [46], Pegasus [47], PortalLab [48] and GridSpeed [49]. Although all of them provide security service, the security only includes very basic authentication, authorization and Delegation. They only focus on protocol level (SOAP/HTTPS) and identity level (Kerberos and X.509), but put little effort on policy level, which is very important to solve interoperability challenge.

5.4.2 Policy Expression and Policy Exchange

How to implementation an adaptive grid portal, which can exchange policy among different domains is another research problem. Security interoperability requires that each domain can be able to specify any policy when it wants to set up a secure conversation with other domains.

IBM security roadmap [35] describes a web services specification that WS-security, WS-Policy, WS-Trust and WS-privacy together provide the foundation capability to create a secure interoperable web services across trust domain. WS-Policy can describe how clients and servers can specify the security policy to establish a conversation. The most attractive feature is WS-Policy is fully extensible and does not have any limitation on the requirements and capabilities of security policy.

Here has an example in enforcing policies: Joshua is a parts provider and Nicholas is a dealer. Nicholas put its business policy into the WS-Policy, after that Joshua will require Manufacturer Token from manufacture and then confirms that token with Nicholas. In this scenario, it can dynamically exchange policy between dealers and providers. WS-Policy also can specify constrains on the information can be stored in order to ensure compliance with privacy policy.

Complex relationships among these different domains will have different types of policies, such as VO-oriented, local systems-oriented, and a mix of the two. G- PBOX [50] provides a policy exchanging framework for Grid environments and it is an approach to the representation and management of those policies. G-PBOX has various levels, such as VO, Domain, Site, Farm level, which will clearly limit the scope of specified group of policies.

Every PBox has a Policy Enforcement Point, which can decide if policy enforcing is required. The PBox of different layer can send or receive new policies. During the exchanging policy, the PBox has been considered as peer or master:

Peer: PBox will put received policies into a waiting queue until the local administrator reviews them and decides whether to accept or refuse them.

Master: PBox will immediately accept received policies are

Stephen.Yao [51] describes an adaptive security framework that various domains can rapidly specify, update and enforce security policies. Although the framework is not designed for Grid, it is suitable for service-oriented architecture. By providing unambiguous logical policy specification, the system can specify and enforce flexible security policies.

5.4.3 Access Control

The Web services in Grid environment can access and share of distributed resources crossing multi administration domains. It is very important to provide a uniform access mechanism. During those years, researchers have come out many research ideas about it, such as role-based access control, Trust-based access control, policy-based access control and Agent-based access control, etc. Usually, the simple way to realize secure access control in computational grids is just to provide an identity certificates for requester. When requester send service request to service provider, it should send the identity certificates to service provider. After getting the certificates, the service provider will check if it is correct or not.

However this basic approach can not be scale when requesters and resources are increasing. University of Southern California [56] proposed an adaptive trust negotiation access control approach for Grid system. This approach addresses the problem of secure access control. When resources get access request, they should protect them from adversaries who may want to misuse, exhaust or deny service to resources. The trust negotiation access control frame allows requester and provider to establish trust relationship based on attributes other than identity. It is scalable and efficient. Because even resources and requesters become larger, they still can establish trust relationship by negotiation each other.

Because the size of the distributed clients and resources is becoming larger and larger, the access control system should get security relevant contextual information from Web services, such as time, location, security policy and current environment state. These contexts can dynamically catch the changing of access requirement, thus it is very important to effectively access relative resources. From security point of view, the context will affect the trust relationship between two distraction domains. In order to handle those issues, Dr Elisa. Bertino [55] proposed a trust-based context-Aware access control framework in 2004.

As we all know, the express and exchange of security policy is a big issue in Grid security. Because when we access resource crossing different administrative domains, the security may be totally different. We need a uniform policy mechanism couple with fine-grain usage policies for enforcing authorization. Louisiana Tech University [57] presents a policy-based access control framework to address the policy express and enforce problem for Grid computing. They use Policy Markup Language (PML) to describe the site policy. The core part of their approach is the policy engine. The client sends the identified certification to Grid Resource and Allocation management (GRAM) gatekeeper. The policy engine will start up after GRAM authenticated to the requester. The policy engine automatically resolves attribute values based one the user information and profile. After get those values, the engine will check it with LDAP server and then return accepting or rejecting decision to the requester.

5.4.5 Security Environment

Grid system shares resources crossing multi administrative domains. Users can submit their jobs to some node in Grid system. After getting these jobs, the node computes them and sends result to users. During this process, two scenarios will occur. First, the node with shared resources may be malicious and affects the result. Second, the jobs may be malicious and affects the node action.

Those two cases are very important for security environment. For case one, if node is malicious and return the incorrect result to users, the users need resubmit the jobs. Even worse, if the returned result is with virus, the users can be infected and crashed. Few research works focus on case one, because it is difficult for users to know if the result is good or not.

Ali Raza Butt [58] proposed a method to address the problem of malicious code of users. They thought two aspects of security should be addressed to achieve an ideal grid environment: decoupling of grid user management from the physical entities and guaranteeing safe grid usage in the absence of user accountability. They used two methods to overcome grid security issues in their approach:

Providing a private and anonymous account for every job submitter.

The standard user account is a unique numeric identifier. They only maintain Grid user data in

logical use accounts. Because the account is not tied to a specific numeric identifier, the system could dynamically manager the permission of the account.

When system gets a job from users, it will put the job into one queue with security priority. Based on the security priority, the system knows the application should run on which security level.

Two-level motioning process approach

Runtime process monitoring can provide control allover the system, but this method has extra overheads. System can not monitor every process from starting to ending.

In the first level, they provide a restricted shell to enforce the host security policy. The restricted shell decides whether the application should be monitored.

Before execution an application, the restricted shell informs level-two to monitor the applicant at runtime.

6. Summary and Directions

Grid computing is a very hot research topic and security issue is very important in Grid. Several Grid architectures have been proposed in last ten years. The Grid systems have three security challenges: integration with existing system, interoperability with different environment and trust relationship among domains.

Our survey mainly focuses on interoperability challenge. Several research topics have been described, such as inter-Grid interoperability, policy express, access control for Web services and security Grid environment.

Grid computing already has history for more than ten years. Although, many grid software services have been developed very well, for example, Resource management, resource discovery and fault tolerance, the research of Grid security just starts. Grid security still is one of the most crucial and difficult research topics. Based on the analysis to research of Grid computing in research institutes, Universities and industry companies, we should focus on following research topics in Grid security:

- VO-based interoperability
- Message-based security service in Protocol level and policy level
- Policy express and exchange in inter-Grid and intra-Grid
- Policy based Access control
- Security Grid environment
- Dynamically secure delegation for service requester.
- Adaptive trust relationship management

References

[1] Foster, I. and Kesselman, C, “The Grid: Blueprint for a New Computing Infrastructure,” Morgan Kaufmann, 1999.

[2] Foster, I., Kesselman, C. and Tuecke, S, “The Anatomy of the Grid: Enabling Scalable Virtual Organizations,” International Journal of High Performance Computing Applications, 15 (3). 200-222. 2001.

[3]Ian Foster, “What is the Grid? A Three Point Checklis,” 2002

[4] Jay Unger, Matt Haynos. “A visual tour of Open Grid Services Architecture,” 2003

[5] Nataraj Nagaratnam, Philippe Janson, John Dayka. Anthony Nadalin, Frank Siebenlist, Von Welch, Ian Foster, Steve Tuecke, “The Security Architecture for Open Grid Services,” 2002

[6]Ali Raza Butt,Sumalatha Adabala,Nirav H. Kapadia,Renato J. Figueiredo,and Jose A.B. Fortes, “Grid-computing portals and security issues,”, 2003

[7] Oracle Company, “Grid Computing with Oracle: An Oracle Technical White Paper,” March 2005

[8] Dan Kusnetzky, Carl W. Olofson, “WHITE PAPER Oracle 10g: Putting Grids to Work,” April 2004

[9] Oracle Company, “Oracle Database 10g: The Database for the Grid An Oracle White Paper,” January 2005

[10] Oracle Company, “Oracle Application Server 10g -Grid Computing,” July 2005

[11] Christian Antognini, “Is Oracle Database Moving Toward Grid Computing?,” 2004

[12] Oracle Company, “Oracle 10g: Infrastructure for GridComputing An Oracle White Paper,” September 2003

[13] CALBBY Analytics, “Oracle vs. IBM in Grid Computing: A Comparative Overview of Each Company’s Grid Strategies, Products, and Services,” March, 2006

[14] Oracle Company, "Oracle Enterprise Manager Grid Control Installation and Basic Configuration," May 2006
[15] Sun Microsystems, Inc,, "The Sun Grid Solution: Competitive Advantage for Organizations That Depend Upon Computational Power," 2005
[16] Sun Microsystems, Inc,, "The Sun Grid Solution: Deploying Grid Computing for Competitive Advantage," 2005
[17] Sun Microsystems, Inc,, "An Overview of Grid Computing From Sun," 2002
[18] Jean S. Bozman John Humphreys, "Sun Server and Oracle Grid," November 2005
[19] Sun Microsystems, Inc, "Grid Infrastructure Reference Architecture," 2006
[20] IBM, "The Grid Report of IBM," 2004 Edition
[21] Bart Jacob, Michael Brown, Kentaro Fukui, Nihar Trivedi, "Introduction to Grid Computing," December 2005
[22] Clabby Analytics, "Competitive Positioning: IBM in Grid Computing," 2003
[23] IBM, "The Era of Grid Computing: A new standard for successful IT strategies, " January 2004
[24] GridLab project, http://www.gridlab.org/ (June.2006)
[25] ATLAS project, http://heppc1.uta.edu/atlas/grid-testbed/index.htm (June.2006)
[26] DOE Science, "Introduction to the DOE Science Grid," 2005
[27] DOE Science, "DOE Science Grid: Enabling and Deploying the SciDAC Collaboratory Software Environment," 2005
[28] DOE Science, "The DOE Science Grid-Phase 2," 2005
[29] Globus project, http://www.globus.org/ (June.2006)
[30] Foster, C. Kesselman, G. Tsudik, S. Tuecke, "A Security Architecture for Computational Grids," Proc. 5th ACM Conference on Computer and Communications Security Conference, pp. 83-92, 1998.
[31] Frank Siebenlist, Von Welch, Steven Tuecke, Ian Foster, Nataraj Nagaratnam, Philippe Janson, John Dayka, Anthony Nadalin, "Global Grid Forum Specification Roadmap towards a Secure OGSA," 2002
[32] Mike Jones, "An overview of the methods used to create a secure grid," April 2004
[33] The Globus Security Team, "Globus Toolkit Version 4 Grid Security Infrastructure: A Standards Perspective," 2005
[34] Kaizar Amin, Gregor von Laszewski, Mikhail Sosonkin,Armin R. Mikler, Mihael Hategan, "Ad Hoc Grid Security Infrastructure, Grid Computing," The 6th IEEE/ACM International Workshop, 2005
[35] IBM Corporation and Microsoft Corporation, "Security in a Web Services World: a Proposed Architecture and Roadmap, A joint security whitepaper," April 7, 2002
[36] Geoffrey Fox, "Grid Computing Environments," 2003
[37] Choonhan Youn, Marlon Pierce and Geoffrey Fox , "Developing Secure Web Services for Computational Portals", 2004
[38] Marlon Pierce and Geoffrey Fox, "Interoperable Web Services for Computational Portals", 2005
[39] Fox, G. C., Gannon, D., and Thomas, M, "A Summary of Grid Computing Environments. Concurrency and Computation: Practice and Experience," Vol. 14, No. 13-15, pp 1035-1044.
[40] Globus Project, "GT4 Message & Transport Level Security Developer's Guide", 2006
[41] GRIP (GRID INTEROPERABILITY PROJECT), http://www.grid-interoperability.org/grip-papers.htm (June.2006)
[42] Ge He Zhiwei Xu, "Design and Implementation of a web-based computation Grid Portal," Proceedings of the IEEE/WIC International Conference on Web Intelligence , 2003
[43] R. Barbera, "The GENIUS Grid Portal," 2003
[44] Unicore, "Unicore -Globus Interoperability of Grid Infrastructures," 2005
[45] F. Donno, V. Ciaschini, D. Rebatto, L. Vaccarossa,M. Verlato, "The WorldGRID transatlantic testbed: a successful example of Grid interoperability across EU and US domains," Computing in High Energy and Nuclear Physics, 24-28 March 2003
[46] M. Thomas, J. Boisseau, "Development of NPACI Grid Application Portals and Portal Web Services," Cluster Computing, 2003

[47] Gurmeet Singh, Ewa Deelman, Gaurang Mehta, Karan Vahi, Mei-Hui Su, “The Pegasus Portal: Web Based Grid Computing,” ACM Symposium on Applied Computing, 2005
[48] M. Li, P. van Santen, D.W.Walker, O.F.Rana, M.A.Baker, “PortalLab: A Web Services Toolkit for Building Semantic Grid Portals,” Proceedings of the 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid , 200
[49] Toyotaro Suzumura, Hidemoto Nakada, Satoshi Matsuoka, Henri Casanova, “GridSpeed: A Web-based Grid Portal Generation Server,” Proceedings of the Seventh International Conference on High Performance Computing and Grid in Asia Pacific Region, 2004
[50] V. Ciaschini, A. Ferraro, A. Ghiselli, G. Rubini, R. Zappi, “G-PBox: A Policy Framework for Grid Environments,” The 2004 Conference of High Energy Physics, 2004
[51] Stephen S. Yau, Yisheng Yao, Zhaoji Chen, Luping Zhu, “An adaptable security Framework for Service-based systems,” 10th IEEE International Workshop on Object-Oriented Real-Time Dependable Systems, 2005
[52] W3C, “Simple Object Access Protocol (SOAP) 1.1. W3C, Note 8,” 2000.
[53] Christensen, E., Curbera, F., Meredith, G. and Weerawarana., “Web Services Description Language (WSDL) 1.1. W3C, Note 15,” 2001,.
[54] Brittenham, P, “An Overview of the Web Services Inspection Language,” 2001
[55] Rafae Bhatti, Elisa Bertino, Arif Ghafoor, "A Trust-based Context-Aware Access Control Model for Web-Services,", Proceedings of the IEEE International Conference on Web Services, 2004
[56] Tatyana Ryutov, Li Zhou, Clifford Neuman, Noria Foukia, Travis Leithead, Kent E. Seamons, "Adaptive Trust Negotiation and Access Control for Grids," IEEE Grid Computing Workshop, 2005
[57] Jin Wu, Chokchai Box Leangsuksun, Vishal Rampure, "Policy-based Access Control Framework for Grid Computing," Proceedings of the Sixth IEEE International Symposium on Cluster Computing and the Grid, 2006
[58] Ali Raza Butt, Sumalatha Adabala, Nirav H.Kapadia,Renato J.Figueiredo, and Jose A.B.Fortes, "Grid computing portals and security issues," Parallel and Distributed computing,2003http://www.ieice.org/eng/shiori/mokuji.html

Jianmin Zhu is a PhD student in the Erik Jonsson School of Engineering and Computer Science of the University of Texas at Dallas since Aug.2005. He got Bachelor degree of computer science in China. In 2003, he got master degree of computer science from Chinese Academy of Sciences. Now he is doing research in Grid security under supervision of Dr. Bhavani Thuraisingham.
His research interesting is secure Grid computing, data security, sensor computing, mobile/wireless technologies and the middleware of distributed system. In the past five years, he has published several papers in Grid computing and the middleware of distributed system.. He has internship experience in Wireless Network Group of Microsoft Research Asia and working experience in the R&D department of Shanghai Alcatel.

Prof. Bhavani Thuraisingham joined The University of Texas at Dallas (UTD) in October 2004 as a Professor of Computer Science and Director of the Cyber Security Research Center in the Erik Jonsson School of Engineering and Computer Science. She is an elected Fellow of three professional organizations: the IEEE (Institute for Electrical and Electronics Engineers), the AAAS (American Association for the Advancement of Science) and the BCS (British Computer Society) for her work in data security. She received the IEEE Computer Society’s prestigious 1997 Technical Achievement Award for “outstanding and innovative contributions to secure data management.”

Dr. Thuraisingham’s work in information security and information management has resulted in over 70 journal

articles, over 200 refereed conference papers and workshops, and three US patents. She is the author of seven books in data management, data mining and data security including one on data mining for counter-terrorism and another on Database and Applications Security and is completing her eighth book on Trustworthy Semantic Web. She has given over 30 keynote presentations at various technical conferences and has also given invited talks at the White House Office of Science and Technology Policy and at the United Nations on Data Mining for counter-terrorism. She serves (or has served) on editorial boards of leading research and industry journals including several IEEE and ACM Transactions and currently serves as the Editor in Chief of Computer Standards and Interfaces Journal. She is also an Instructor at AFCEA's (Armed Forces Communications and Electronics Association) Professional Development Center since 1998 and has served on panels for the Air Force Scientific Advisory Board and the National Academy of Sciences.

Prior to joining UTD, Thuraisingham was an IPA (Intergovernmental Personnel Act) at the National Science Foundation from the MITRE Corporation. At NSF she established the Data and Applications Security Program and co-founded the Cyber Trust theme and was involved in inter-agency activities in data mining for counter-terrorism. She has been at MITRE since January 1989 and has worked in MITRE's Information Security Center and was later a department head in Data and Information Management as well as Chief Scientist in Data Management. She has served as an expert consultant in information security and data management to the Department of Defense, the Department of Treasury and the Intelligence Community for over 10 years. Thuraisingham's industry experience includes six years of research and development at Control Data Corporation and Honeywell Inc.

Dr Thuraisingham is the Founding President of "Bhavani Security Consulting" - a company providing services in consulting and training in Cyber Security and Information Technology. She was educated in the United Kingdom both at the University of Bristol and at the University of Wales.