A BROKERED APPROACH TO INTEROPERABLE SECURITY IN OGSA-BASED GRID SYSTEMS

Demetrios Loutsios[a] and Maree Pather[b]

[a, b] Nelson Mandela Metropolitan University

[a, b] Faculty of Engineering, Department of Computer Studies, PO Box 77000,
Nelson Mandela Metropolitan University, Port Elizabeth, 6013
[a] demetrios@nmmu.ac.za
[b]maree.pather@nmmu.ac.za

ABSTRACT

The need for organisations to share data and collaborate on a large scale with geographically dispersed parties has increased dramatically in recent years. Grid Services allow for large scale collaboration between geographically-dispersed parties running diverse hardware and software platforms, over public networks such as the Internet. Grid Services are an evolution of Web Service technology and other open, platform-independent standards. Current research efforts have been undertaken to standardize grid implementations. With the efforts of the Global Grid Forum (GGF) and other interested parties, the Globus Toolkit has been developed. The focus of this paper is to define a holistic security strategy for implementing Globus-based Grids.

The Globus Toolkit is an open source software initiative, providing a set of tools and a platform for grid developers to build onto. The Toolkit is currently the de facto standard for Grid Service implementations, and is in its fourth major revision GT4 (Globus Toolkit version 4). The Globus Toolkit consists of a number of core components for implementing grids; the component of interest to this research is the Globus Security Infrastructure (GSI). This research looks at a layered approach to securing grids, making use of a defence-in-depth approach. The focus is on the Globus Toolkit and GSI, local hardware and software configurations for remote sites, and communications (i.e. TCP/IP stack, RMI, RPC, etc). The STRIDE model will be used to provide a base for understanding hackers attack methodologies and threats faced by modern Grids.

KEY WORDS

Grid Security, Globus, Brokered Grids, STRIDE, Grid interoperability

A BROKERED APPROACH TO INTEROPERABLE SECURITY IN OGSA-BASED GRID SYSTEMS

1 INTRODUCTION

Due to a number of factors, grid computing has gained in popularity and application. To date many distributed computing paradigms exist, such as Common Object Request Broker Architecture (CORBA), Java's' Remote Method Invocation (RMI), Common Object Model (COM), Web services, etc. Grid Services are an evolution on existing paradigms (Foster, C. Kesselman, S. Tuecke; 2001). The use of open standards such as Open Grid Service Infrastructure (OGSI), extensible Mark-up Language (XML) and Simple Object Access Protocol (SOAP) easily allows for heterogeneous platforms to communicate and share computing resources within a virtual organisation (VO) context.

According to Foster, the goal of Grid Computing is to "coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations". Sharing is not just denoted as file exchange or data sharing, but rather direct access to computers, software, data, and other resources, as is required by a range of collaborative problem-solving and resource-brokering strategies emerging in industry, science, and engineering (Foster, et al;2001). Furthermore, a set of individuals and/or institutions defined by such sharing rules form what is known as a virtual organization (VO). Current implementations of grid computing models have had great success in a variety of contexts, from the monitoring of natural phenomena, to the prediction of market trends among consumers, to name a few. However, as the face of modern computing evolves, so does the challenges to the underlying technologies that drive it.

These challenges include: location, connectivity and platform configurations. Implicit in these challenges are issues of interoperability, ownership and responsibility, security, performance, and reliability.

Grid Services is a distributed computing paradigm, built on Web-Services and SOAP. The use of XML Web-Services as an underpinning technology – notably the WS-* set of specifications for extending SOAP functionality - makes it possible for most of these issues to be addressed (Foster, et al; 2001). A Grid is a collection of Grid Services, or other Grids logically grouped into a Virtual Organization (VO). Grid Services provide a number of services, including processing or computational power, database housing, and application hosting and sharing.

The primary focus of this research is to provide a generic and coherent security framework, to protect Grid Computing resources and users from hackers and intrusion attacks. The process of defining a detailed security strategy for all known vulnerabilities, attacks, possible variants on know attacks, and new or unknown attacks can be a daunting task; almost impossible, at the rapid rate of availability of new hacking tools. It might be more economical to typify the hackers' intentions and generic goals when attacking a system, with a view to defining a threat model that can be applied to Grids and Grid services.

A two-level strategy will be discussed in implementing a defence-in-depth strategy for protecting Grids, within the Globus Context (The Globus Toolkit is used for developing Grid Service solutions; see http://www.globus.org/toolkit/). The first level is concerned with Grid Services (the lowest level in a Grid), while the second level will look at the Grid as a whole and the particular challenges faced by Grid designers when implementing them. The STRIDE model (Meier et al; 2003) for hacker behaviour will be investigated and applied to a risk assessment methodology, to provide Grid designers with a framework for developing security policies to protect their Grid

Services. Additional information will be provided on Grids and the Globus Toolkit, as well as a threat-modelling strategy that can be applied to Grid computing.

To summarise, this paper will:

- Discuss Grid Computing and supporting technologies, such as the Globus Toolkit,
- Provide an introduction to the STRIDE model for threat-modelling, and
- Describe a holistic security framework for defining security strategies in a Grid environment.

2 GRID COMPUTING

Grid Computing allows large heterogeneous groups to share computing processing power, as well as other computing resources. Foster defines a grid as follows: "A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities" (Foster; 2002). Foster, furthermore, proposes a three-point checklist, to which grid systems must comply to:

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

Grid research is currently focused on standards to facilitate resource virtualization and to accommodate intrinsic heterogeneity of resources in distributed environments (Stuer, V. Sunderam, J. Broeckhove; 2004). The concept of Grid Services is a natural evolution on Grid Computing.

Open Grid Services Infrastructure (OSGI) is a specification which defines basic mechanisms and interfaces which can be used to build Grid functionality. Open Grid Services Architecture (OGSA), is an open standard for Grid Services implementation. Standard frameworks, based on XML, are being used to describe standard service specifications, to allow clients to discover and use services across platform, and domain contexts (Ibid). OGSA defines a best practice for implementing grid-enabled services.

The Globus Toolkit (http://www.globus.org/Toolkit/), now in version 4, is an open source software framework, designed to implement grid services. Its goal is to develop and promote standard grid protocols to enable interoperability and shared infrastructure. A lot of the work done on the Globus project is through the Global Grid Forum (GGF).

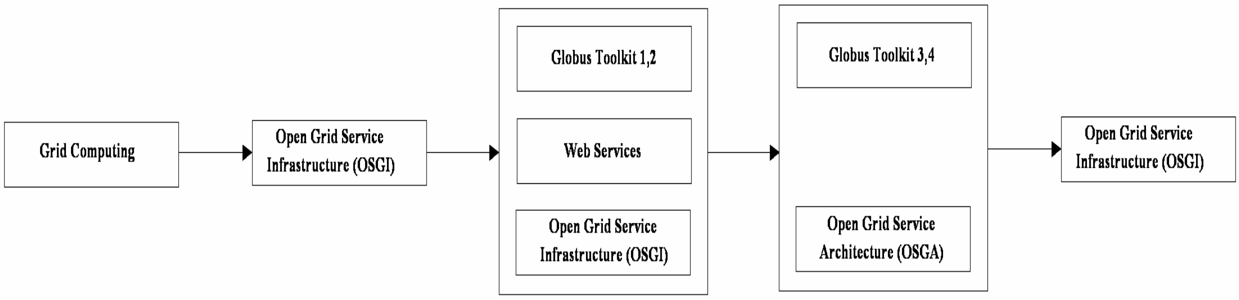
Grid Computing
Open Grid Service Infrastructure (OSGI)
Globus Toolkit 1,2
Web Services
Open Grid Service Infrastructure (OSGI)
Globus Toolkit 3,4
Open Grid Service Architecture (OSGA)
Open Grid Service Infrastructure (OSGI)

Figure 1: Timeline diagram of Grid Services, concepts and related standards and technologies

The above diagram shows a logical timeline of standards and technologies that support or make up modern Grid Services, the following section will discuss the Globus Toolkit in more detail.

2.1 Globus Toolkit

The Globus Toolkit facilitates an open source implementation of all the protocols and primitives defined by Open Grid Service Infrastructure (OSGI), for implementing grid services (Sandholm; 2003). The Toolkit consists of a number of components, allowing one to develop and implement a grid service. This section will introduce these components and briefly discuss them.

The Globus Toolkit has a layered architecture; high level global services are built on a core set of lower level services. At the bottom of the hierarchy, and possibly one of the most important services, is the resource management service, Globus Resource Allocation Manager, or GRAM; this is responsible for assigning as well as de-allocating resources to services (Foster; 1998).

In most distributed system architectures, communication plays a key role. The Globus Toolkit provides a communication component, NEXUS. NEXUS is a library of lower level communication APIs that provide support for higher level communication (Foster; 1998).

Security is also a major concern in grid implementations. Security needs in grids are diverse, including authentication, access control and privacy. Globus Security Infrastructure (GSI) is the component within the Toolkit that implements security. GSI primarily looks at the problem of authentication, and therefore leaves open a large area for future research in the security space (Ibid).

In a dynamic environment such as in a grid system, the need to be able to easily access information about services, components, and applications, in a timely fashion, is important. This is in order to allow for adaptation to changes in system structure and state. Globus Meta-Computing Directory Service (MDS) stores and makes accessible information such as the architecture type; operating system version and amount of memory on a computer; network bandwidth and latency; available communication; Protocols; and the mapping between IP addresses and network devices (Foster; 1998). MDS provides tools and APIs to allow for discovery, publishing and access information about the structure and state of a grid.

Health Beat Monitor (HBM) provides simple management services for monitoring the health and status of sets of remote processes. The HBM consists of several client APIs. A process can register with the HBM, which then acts as a data-collection base, periodically receiving “heart-beat” information about a process. Other processes can query the HBM for the status of another process.

Globus also provides Global Access to Secondary Storage (GASS), a component that allows programs with access to simple C I\O libraries the ability to open, edit and save files on remote computers.

The final core service in the Globus Toolkit is Globus Executable Management (GEM). GEM supports the remote identification, creation and location of executables in heterogeneous environments.

Grid Concepts and the Globus Toolkit were discussed in this section, the following section will discuss threat modelling and hacker behaviour.

3. Threats

To understand the importance of securing one’s information, it is important to understand what are the threats and impact associated with insufficient security (Whitman; 2003). A wide range of threats exist. These threats are unique for the various parts of a grid, although the attacker’s (generic) goals might be the same (Meier, Mackman, Vasireddy, Dunner, Escamilla, Murukan; 2003). Knowing how and why a hacker can attack an information system is a good starting point to identifying threats to an organization’s information assets.

3.1 Attackers goals

There are a wide range of possible attacks, and further fine-grained variations on these attacks. The best method to classify threats to one's system is to identify the hacker's goals when performing an attack. STRIDE is the acronym for an approach to categorize different threat types (Ibid):

- **Spoofing** - The hacker's goal when spoofing is to try gain access to the system by mimicking legitimate user-credentials or network traffic.
- **Tampering** – This is the unauthorized altering of information, while it is in transit between two computers.
- **Repudiation** – Prevents administrators from knowing if users (legitimate or not), have performed an action.
- **Information disclosure** – This is the unwanted exposure of private information.
- **Denial of Service** – This is the process of making services un-available to users.
- **Escalation of privileges –** This attack occurs when a user of limited privileges assumes the roll of a privileged user, in order to steal, corrupt, or deny access to information asset.

3.2 Hacker's methodology

Microsoft (Ibid) identifies the basic attack approach adopted by hackers, this approach defines generic steps a hacker will need to perform in order to complete a successful attack; not all steps are required in every instance. The steps in the hacker's methodology are listed below:

1. **Survey and assess** - Survey and assess is the initial stage of the hacking process. The hacker will try to learn of possible servers and services on the network. The hacker will then try to find possible weakness and exploits, to try and gain access to the target machine.
2. **Exploit and penetrate** - Once the hacker completes the survey phase, the next step is to exploit and penetrate the target.
3. **Escalate privileges** - Upon completing the attack and delivering the payload, the hacker will then attempt to create a backdoor to access the desired server. Immediately an attempt will be made to escalade privileges, specifically to administrator.
4. **Maintain access** - Once the attacker has administrative privileges, they will try and make further access easier and try to hide his or her tracks. A common method of making back door access possible is to plant back-door applications. Hackers will often attempt to clear event logs at this stage.
5. **Deny service** - If the attacker is not successful in his or her attack, they will try launch a Denial of Service attack (DoS), to deny others use of the service.

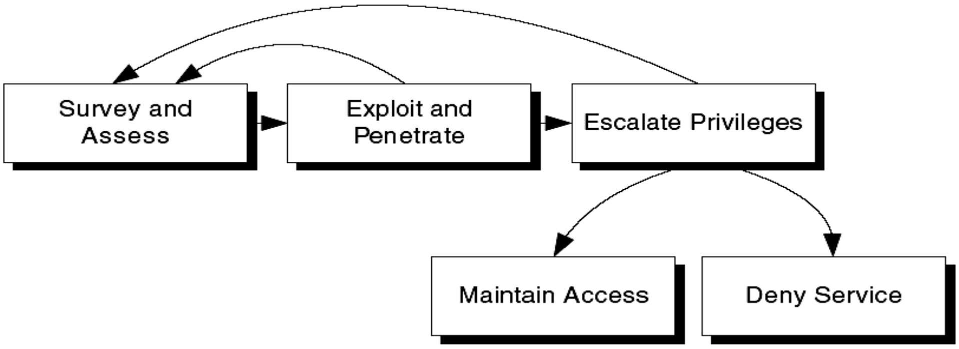

Figure 2. Steps in a typical attack (Meier et al, 2003)

In a Grid Community, nodes (clients, servers, brokers), messages and message pathways are exposed to a range of threats.

4. Holistic Grid Security Framework

Grids can be logically divided into two levels, based on security needs and challenges. The first layer is the local grid service layer, which is concerned with a data or computational grid service as a separate local entity below the common grid infrastructure in the VO (Virtual Organization) context. The second layer is the Common Grid layer. This layer consists of all GT grid services. A single VO can span countries, or the globe.

One of the biggest problems faced by Grid designers is implementing authentication and authorization between Grid Services or sites. Each site may have its own local security policy, and will make use of a different set of technologies (Foster; 1998b). This includes security issues when crossing trust domains and grid-to-grid security issues, such as single-sign on authentication and authorization. We will first look at the lowest level, the local grid services that make up a grid. Consider the following diagram:

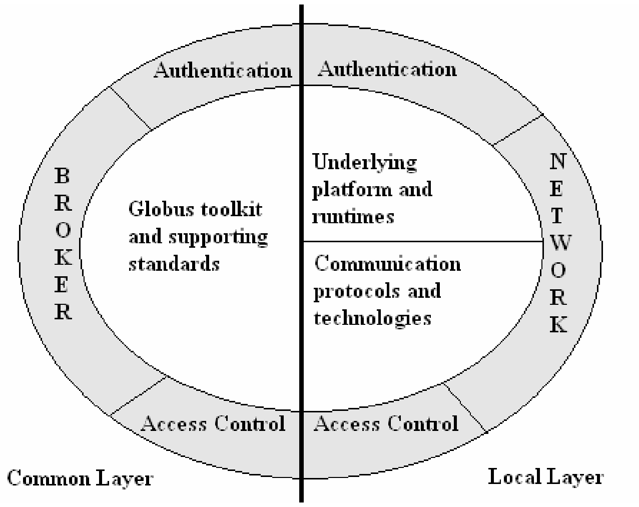
Authentication
Authentication
BROKER
Globus toolkit and supporting standards
Underlying platform and runtimes
Communication protocols and technologies
NETWORK
Access Control
Access Control
Common Layer
Local Layer

Figure 3: Logical view of a Grid

In the above diagram, the core represents a local Grid Service. The outer layer represents aspects and challenges of a Grid in a VO (virtual organization) context. Local Grid services can be divided into four logical security layers:

Local (Grid Service) level:

- **Underlying platforms and runtimes:** Platform security is based on the hardware and software platforms the Grid Service is hosted on, i.e. INTEL x86, SPARC, etc. running LINUX, UNIX or Windows operating systems. Runtimes dictate the security runtime environment; the grid software is typically hosted in e.g. JAVA or .NET.
- **Communications protocols and technologies:** Service components can communicate over a variety of mediums and protocols. The most commonly used communication protocol is TCP/IP, over variety of mediums, broadband, wireless, VPNs, etc. Each communication mechanism involves an appropriate local security implementation, e.g. IPSec.
- **Network:** the network “cloud” between (local and) grid participants, and (local and) grid services in a virtual organization is, fortunately, commonly based on TCP/IP, with interoperability on LAN and WAN interfaces being provided for through hardware and software gateways.

- Implicit in the **actual service components** will be additional security measures such as .NET strong names (with embedded credentials) and role-based security at the component/class/method level.

The Common Grid Layer Challenges:

- **Authentication:** A variety of authentication challenges are presented when multiple sites or grid participants have varying security policies and authentication implementations. Identification and authentication credentials have to be persisted from the common layer to local layer.
- **Access Control (Authorization):** The challenge of maintaining access control assertions down to the local components is obviously great as common policies and interoperable implementations are critical factors. Hence, Web Services standards, such as WS-Security and WS-Policy are crucial to the GT framework.
- **Broker:** An abstracted software component, acting as an intermediary between parties, is the backbone of Grid Services integration. It deploys GT (or equivalent) middleware, common security and interoperability policies and implementations. This layer provides access to the local grid service and associated services (database, application hosting, processing, etc). It uses interoperable standards, such as XML and SOAP. It uses GT mechanisms for mapping security credentials and interoperability mechanisms from the Common Grid Layer to the Local Grid Layer.

Next, Grid Services security will be considered in terms of the STRIDE approach discussed earlier.

4.1 Grid Services Security and STRIDE

The STRIDE model for threat modelling was introduced previously, as well as a hacker methodology for attacking information systems.

- Grid services have unique security needs, largely due to their open nature and interconnectivity. (Grid services are largely un-standardised, in terms of underlying platforms and communications technology. As discussed, they are often built on a variety of hardware, software, and operating system platforms, as well as a range of possible communication protocols and technologies (Baker, et al; 2000). However, the common layer is, typically, standardized in terms of using open standards, such as XML and SOAP, and the Globus Toolkit (a de facto standard for building Grid services). This layer is standardized in order to facilitate integration of underlying heterogeneous platforms and technologies.

In the previous section it was determined that there are several generic hacker goals, as well as a set of generic steps a hacker will follow to attack a system. The following table shows what goals are typically applicable to each particular layer, defined above, in a Grid service VO.

	Spoofing	Tampering	Repudiation	Information Disclosure	Denial of Service (DoS)	Escalation of privileges
Underlying platform and runtimes	X		X		X	X
Communication protocols and technologies	X	X			X	
Local Grid Service	X			X	X	X
Broker: Globus Toolkit and supporting standards	X		X	X	X	X

Figure 4: table of typical goals of a hacker when attacking each layer of a grid

The above table can be used as a generic basis for developing a local security strategy to protect a Grid Services Deployment. The details of implementation are beyond the scope of this paper. However consider the following scenario:

Tampering is a risk associated with the communications layer of a grid. A typical method of tampering with network traffic is a “man-in-the middle” attack, in which a hacker will intercept traffic in transit from one node, read the contents and alter it, then pass it on to the intended recipient. A number of controls can be implemented to combat this threat, such as encryption. On a high level, we have determined that the grid implementation will require encryption to protect information in transit. When deciding on a Grid-wide encryption strategy for information in transit, we can determine if IPSec will be used, or more commonly in this instance, encrypted SOAP packets at layer 7 (of the OSI reference model).

4.2 Grid Security Implementation Scenario

Grids require standard security functions, such as, authentication, access control, integrity, privacy, and non-repudiation (Foster; 1998b). This is difficult to implement in a Grid-wide Community, due to a number of factors. VOs (Virtual Organizations) can be made up of a number of diverse geographically disperse sites, implementing non-compatible local security policies\technologies. Consider the following scenario:

User-A at Site-A starts an analysis program that sends code to be executed on Site-B, but Site-B requires a dataset on Site-C to perform the analysis. The application at Site-A contacts a broker at Site-D to obtain idle resources needed to process the task at hand. The Broker then initiates communications with sites E,F,G in order to complete the task at hand. These sites will need to maintain communication between them (possibly using a multicast protocol), as well as the broker, the original site (requesting site), and the user.

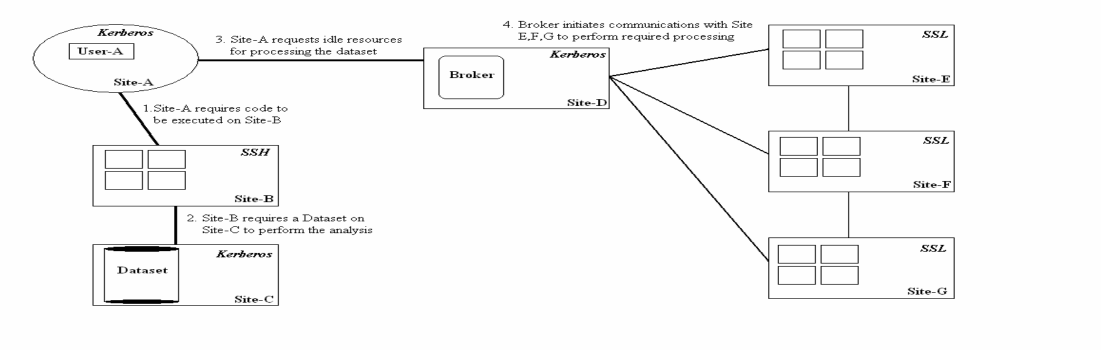
Kerberos
User-A
Site-A
3. Site-A requests idle resources for processing the dataset
4. Broker initiates communications with Site E,F,G to perform required processing
Broker
Kerberos
Site-D
1.Site-A requires code to be executed on Site-B
SSH
Site-B
2. Site-B requires a Dataset on Site-C to perform the analysis
Dataset
Kerberos
Site-C
SSL
Site-E
SSL
Site-F
SSL
Site-G

Figure 5: Example of large scale distributed computing environment

The above scenario depicts many distinctive characteristics of the Grid Computing environment (Foster; 199b):

- The user population is large and dynamic.
- The resource pool is large and dynamic.
- A computation (or processes created by a computation) may acquire, start processes on, and release resources dynamically during its execution.
- The processes constituting a computation may communicate by using a variety of mechanisms. Low level communications (e.g. TCP/IP sockets) can be created and destroyed dynamically during program execution.
- Resources may require different authentication and authorization mechanisms and policies, which we will have limited ability to change. In the above example, this was illustrated this by showing the local access control policies that apply at different sites. These include Kerberos, Secure Socket Library (SSL) and Secure Shell (SSH).
- An individual user will be associated with different local name spaces, credentials, or accounts, at different sites, for the purposes of accounting and access control.
- Resources and users may be located in different countries.

There is need to provide security solutions to grid users that can allow computations, such as in the above described scenario. These solutions must allow for the co-ordination of diverse access control policies and to allow them to operate securely in heterogeneous environments (Foster, 199b).

In order to achieve a global security infrastructure within a VO, a broker can be used to facilitate communications, authentication and authorization at a central site. The implementation of various services and middleware can allow for this.

Grid users are provided two sets of credentials, one applicable to their local security policy and another to a global Grid security policy. A broker service can be used to maintain a table of mappings for user credentials, which allows for comparison to a global security policy for access to resources (Foster; 199b). This mapping of user credentials can provide a transparent single sign-on to the user when interacting with the grid.

1. A user provides the credentials needed to log onto the grid.
2. The user initiates a process that requires remote grid resources.
3. The user's grid credentials are tested against a global access-control policy.
4. The user's rights to that resource are determined.
5. If they have sufficient rights, the grid initiates the communication and provides that grid service with the correct level authentication.

This sort of policy can be implemented using a group policy structure. The structure consists of several elements:

- Resource
- Users
- Groups
- Privilege

A resource is defined, groups are linked to a resource, and one group can contain many users. A group then has a privilege to access the resource defined, one resource can have many groups linked to it.

This simple paradigm can allow for complex authorization structures though out the grid and can cater for scalability. However there are some complexities involved in the implementation and maintenance of the proposed structure. Middleware can be used to reduce the complexities of maintaining the proposed structure, however this falls outside the scope of this research.

A holistic grid security structure was investigated, and it was found grids can be divided into two logical layers, the grid layer and grid service layer. Each of these defined layers has their own security needs. A framework to implement a security strategy was described.

5. Conclusion

A layered approach to securing grids was introduced in this paper. Grids provide a powerful mechanism for collaboration and sharing data and processing resources. The Globus Toolkit was briefly discussed, the Toolkit provides Grid designers a standardised set of software tools and libraries for implementing grid services, and is considered the de facto standard for implementing grid services. STRIDE was discussed as a threat model for categorizing hacker's action and behaviours, based on the outcome of the attack performed or the hacker's goal in attacking a Grid.

The anatomy of a Grid and Grid Services were discussed. It was suggested that Grids can be divided into two logical layers, the common and local layers, each with its own security needs. The lower of the two layers the local layer is concerned with security at a single site. The higher level, or the common layer, is concerned with "global" Grid security issues, including authentication and authorization between Grid Services sites.

A security strategy taking into account all possible attacks and hacks against a Grid is a daunting task. It was proposed here that STRIDE be used by Grid designers as a basis to develop security strategies to protect Grid Services. Each category of STRIDE was found to be applicable to aspects of a Grid Service, providing Grid designers a suitable framework for developing tailored Grid security strategies.

A brokered approach for providing authentication and authorization services on a common grid layer was discussed. Although this approach provides a means to solve the problem of single-sign authentication, grid-wide authorization, etc. It does require the use of Globus (or other standard) middleware. The complexities of implementing and maintaining a brokered approach provide an area for further research.

6. REFERENCES

- I. Foster, C. Kesselman, “*The Globus Project: A status report*”, (1998a)
- I. Foster, C. Kesselman, G. Tsudik, S. Tuecke,*“A Security Architecture for Computational Grids”*,(1998b)
- M.Baker, R. Buyya, D. Laforenza, “*The Grid: International Efforts in Global Computing*”, (2000)
- I.Foster, C. Kesselman, S. Tuecke, “*The anatomy of the Grid*”, (2001)
- I. Foster, “*What is the Grid? Three point checklist*”, (2002)
- M.E. Whitman, “*Enemy at the Gate: Threats to Information Security*”, (2003). Communications of the ACM.
- J.D. Meier, A. Mackman, S.Vasireddy, M.Dunner, R. Escamilla and A. Murukan, “*Improving web security, threats and countermeasures*”, (2003). Microsoft Pattern and Practices.
- T. Sandholm, J. Gawer, “*Globus Toolkit 3 core – A Grid Service Container Framework*”, (2003)
- G. Steur, V. Sunderam, J. Broeckhove, “*Towards OGSA Compatibility in Alternative Metacomputing Frameworks”*, (2004)