Improving Security Measures on Grid Computing

Dominic Damoah[#1] , Edem Kporha[#2], Edward D. Ansong[#3], Ronky Doh, J. Abandoh-Sam[#4], Brighter Agyeman[#5]
Faculty of Science
Valley View University
P.O.Box AF 595
Oyibi-Accra, Ghana

ABSTRACT
The purpose of this project was to come out with a theoretical framework to standardize the solutions to the existing security challenges. The principal objectives were to contribute to the ongoing research to help improve the security measures on grid computing. Hypervisor security model, Host/Platform security model, Security Communication model and Security between Guests were developed and implemented. The major finding derived from this study suggests that Grid Computing Security can have challenges in the following areas: Authentication and Authorization of users, integration to other exiting systems and enforcing trust policies within dynamic Grid environment. Based on the above mentioned challenges with Grid Computing Security a proposed conceptual model has been designed to solve these issues. The various models were implemented and tested to evaluate their performance.

Keywords
Authentication, Grid, Hypervisor, Security, Virtualization

1. INTRODUCTION
The computer industry is a rapidly changing environment and one of the existing trends and growing fields is Grid Computing. According to [1], grid computing is an enabling technology harnessing many computers in a network to solve problems requiring a large number of processing cycles and involving huge amounts of data. Also it is an advanced form of distributed networking. Grid computing links servers, databases, and applications into a "single" large system via specialized software. These networked components which may run different operating systems on multiple hardware platforms, may be physically located in the same building or in various locations around the world. By combining distributed computing resources into a single system, grid computing creates a virtual computer from which users can access data and resources as required.

1.1 In most business industries, it is very expensive and time consuming for small and large businesses to set up I.T. services or even expand their I.T. systems to meet the demand

1.2 of their business environment and day to day transactions. Security is one of the key issues that should be considered when heterogeneous set of computers on a distributed system share their idle resources. According to [15] grid security is becoming a more and more important topic, hence a number of problems remain a nightmare by the existing grid security solutions. This research is aimed at improving the security measures concerned with Grid computing.

2. RESEARCH OBJECTIVES
This research is aimed at finding feasible solutions knowledge gabs on Grid Computing. It is purposed to achieve the following:

- To develop a theoretical framework to standardize the solutions to the existing security challenges.
- To examine and improve on the known security challenges and provide solutions to Grid computing [15]
- To investigate into the reasons for which a solution to an existing challenge does not meet other future occurrences.
- To establish new security solution paradigms to existing grid security policies [15]
- To solve Grid security problems making use of the behavior conformation security policy [37].

3. CONTEXTUAL BACKGROUND
The term grid computing originated in the early 1990s as a metaphor for making computer power as easy to access [20]. CPU scavenging and volunteer computing was popularized beginning in 1997 by Distributed.net and later in 1999 [2] to harness the power of networked PCs worldwide in order to solve CPU-intensive research problems. The idea of the grid (including those from distributed computing, object-oriented programming, and Web services) was brought together by Ian Foster, Carl Kesselman, and Steve Tuecke (Fathers of the Grid).

Companies and businesses are now moving towards the paradigm of having a particular I.T. service without worrying too much about how these services will be implemented or the details behind how these services actually work. This has moved I.T. towards the provision of this badly needed service. Over the years, the computer industry has tried to improve upon the way of delivery of services to make it simpler and easier for non I.T. personnel. This has made the I.T. Industry move towards the paradigm of Software as a Service (S.A.A.S), where I.T. users can have access to particular services they need for as long as they pay for only the amount of time they use it [25].

4. GRID COMPUTING
The term Grid was coined in the late 90s [21] to describe a set of resources distributed over wide area networks that can support large-scale distributed applications. The concept of Grid computing was started by [12] as a project to link geographically dispersed supercomputers but now it has grown far beyond its original intent. Due to the rapid growth of the Internet and Web, there has been a growing interest in Web-based distributed computing and many projects have been started and aimed at exploiting the Web as an infrastructure for running coarse-grained distributed and parallel applications.

Grid computing is emerging as a viable option for high-performance computing as the sharing of resources provides improved performance at a lower cost than if each organization was to own its own closed-box resources. Grid computing is defined in literature as; systems and applications that integrate

and manage resources and services distributed across multiple control domains. According to [12] a grid is a system that conforms to three specific categories: It coordinates resources that are not subjects to centralized control, uses standard, open, general-purpose protocols and interfaces and delivers nontrivial quality of services. Kon et Al define grid computing as coordinated resource sharing and problem solving in dynamic, multi-institution virtual organizations [27].

A. Standards for Grid Environment

Grid computing requires technologies that include: support for executing programs on a variety of platforms, a secure infrastructure, data movement or replication or federation, resource discovery, and resource management.

5. COMPARATIVE STUDY OF GRID SYSTEMS SECURITY CHALLENGES

Grid Computing has remained the most vibrant and useful forms of distributed computing all over the world. Their application is such that it cuts across both academia and industry. In this research, we take a critical look at comparative study of existing grid computing technologies with the aim of understanding their clear security challenges and solutions. To have a good comparison, we start by giving a close look at the existing systems and their functions. Also, we explore security issues and solutions to the existing computing systems. Security is the basis for grid systems and key requirements for ensuring a secure connection involves:

- Data Confidentiality - protection of information exchange against spying
- Authentication - proof of identity
- Data Integrity - protection of message modified in transit (intentionally or by accident)
- Non-repudiation - guarantee that the sender cannot deny that he/she sent the message similarly; a receiver cannot deny receiving the message.

A. Globus Toolkit

In GT4 system User's private keys are stored in a file in the local computer's storage hence, very accessible to be obtained by other users who can gain access to resources. Authentication and authorization of users are key measures to be solved on the GT4. To prevent other users of the computer from stealing, the private key; a single sign-on; the process of enabling user and its agents to acquire additional resources without repeated authentication, (passwords) was achieved with proxies. It consists of a new certificate with new public, private keys and owners identify. This certificate is then signed by the owner and not by the CA and it is given limited lifetime (generally 24 hours) against unauthorized users. This has still not solved issues on authentication and authorization of the GT4 hence with security breaches such as spoofing and eavesdropping [8].

B. Condor

Condor by design allows remote users (who do not have an account on the system) to have access to the system. Also, it allows users to run arbitrary codes, enough for attackers to compromise the system. Related processes on Condor systems consume lots of CPU cycles (perfect for a DoS attack). If an attacker manages to perform Dos attacks against Condor, it causes significant problems if Condor users make serious use of the system. Condor by default runs as root, so vulnerability in Condor leads to a root compromise. Condor in solution supports the strong authentication which is not supported on all platforms. Since it is heterogeneous the Condor pool has problems here [13].

C. UNICORE

UNICORE‘s system security challenges are related to authentication and authorization mechanisms. Also credential management, where member installations of a grid accept only certificates issued by their organization. In that case all partners have to accept two different certificate authorities to allow multi-site jobs. If more than one has the same demand, an additional administrative burden is put on all partners.

Moreover, multi-site jobs involving these two partners become impossible. Even if technical security issues are resolved, administrative ones are still a major obstacle [19].

D. Portable Batch System

Security in PBS is composed of two facets, authorization and authentication. Any user making a request of a batch server must be authorized to request that service. To provide the basis for authorization, each request contains the identity of the user making the request. To insure the user identity is correct, the identity must be authenticated in some manner. In the basic method provided with PBS, an Authentication Batch Request is sent to the server from a privileged port. This request contains the user‘s true identity, the name of the host on which the client is running and the port name to which the client is bound and will be communicating with the server [9].

6. GRID SECURITY THROUGH VIRTUALIZATION

Virtualization has become a common technique for IT systems in many application environments. The idea dates back to the days of mainframe computing [3] but has gained increased popularity in recent years. Grids computing is concerned with enabling coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations. The sharing basically discussed, is not the primarily file exchange as supported by the Web or peer-to-peer systems, but rather, direct access to computers, software, data, services and other resources as required by a range of collaborative problem-solving and resource-brokering strategies emerging in industry, science and engineering. This sharing is, necessary, highly controlled, with resource providers and consumers defining clearly and carefully just what is shared, who is allowed to share and the conditions under which sharing occurs.

A set of individuals or institutions defined by such sharing rules form what we call a Virtual organization (VO) [23]. What basically distinguishes a VO from a classical organization is that it may gather individuals and/or institutions that have agreed to share resources and otherwise collaborate on an ad-hoc, dynamic basis, while they continue to belong to different real organizations, each governed by their own set of internal rules and policies. This poses a challenge when combined with the fact that an individual or institution may be a member of several VOs simultaneously. From a security point of view, one is thus confronted with protection domains that may superpose, straddle, and intersect one another in many different ways. Within this context, we require interoperability among domains while maintaining a clear separation of the security policies and mechanisms deployed by both virtual and real organizations [23].

A. Virtualization

Virtualization is often associated with virtual machines and corresponding CPU abstraction. However, the idea and current trends show a broader meaning that also includes different kinds of resources. Majority of current applications are in the area of CPU virtualization, storage virtualization and network virtualization. Generally virtualization hides the physical characteristics of resources from the resource consumers which can be end users or applications. Virtualization is used in many different contexts which can be grouped into two main types:

- Platform virtualization: involving the simulation of whole computers and
- Resource virtualization: involving the simulation of combined, fragmented or simplified resources [29].

B. Virtualization in Grid Computing

In the area of Grid Computing, virtualization is gaining more and more interest. Virtualization is allowing the addressing of multiple problems in Grid systems such as coping with the heterogeneity of Grid resources, the differences in software stacks, enhanced features in resource management like a more general check pointing or migration models. Adopting virtualization in smart ways gets us closer to real Grid computing with more flexibility in the type of applications and the resources to use.

Virtualization in Grids introduces an abstraction layer in the hard and software stack. Within this layer resources decompose to multiple smaller entities or aggregation of smaller resources to a single entity. For a user who is situated above the virtualization layer the decomposition or aggregation is fully transparent. Examples for resource decomposition are VLANs (Virtual LANs. IEEE Computer Society) which allows running several subnets over one network port? As seen in the example virtualization is already used for networks, but also complete compute platforms can be virtualized. The primary focus in Grid Computing lies in secure resource sharing in terms of access to computers, software and data in a dynamic environment. Sharing of these resources has to be fine grained and highly controlled [20]. Moreover, [20] proposed a three point checklist which characterizes a Grid more in detail:

- delivery of nontrivial qualities of service;
- usage of standard, open, general-purpose protocols and interfaces e.g. for inter-communication;
- Coordination of resources that are not subject to centralized control.

C. Grid Virtual Communities

As discussed earlier, a VO‘s purpose is to provide access control for different users belonging to a particular community. The community should grant privileges to users based on their role, membership and so on. Two examples are the CAS and VOMS.

1) CAS: CAS was developed by the Globus community, which allows resource providers specify course-grained access control policies, while letting the community specify fine-grained policies. The CAS server handles different administrative jobs, such as adding and deleting users, resources, policies and so on, to and from the database. When a user tries to access a resource, authentication is done with the CAS server, using proxy credentials. The CAS server checks for the user‘s identity and rights in the database. It then issues the user a signed policy assertion that contains information about its identity and rights. A CAS client generates a proxy certificate after embedding the policy assertion as a noncritical extension. The user uses this new certificate to request a resource, which first authenticates that user, then parses the assertion information to extract the VO-specific policies or rights. Finally, it combines its local policies with the community policies and decides whether to grant or reject the request.

2) VOMS: VOMS was developed as part of a joint effort between the European Data Grid (EDG) and Data TAG projects to cater the authorization requirement of multiple users distributed across many sites. It provides support for roles, memberships, groups, and capabilities. As with CAS, when a user wants to access a resource in a VO, that user first contacts the VOMS server and authenticates itself. Then, the user sends the request to the VOMS server. After checking for request correctness, the VOMS server issues a signed attribute certificate to the user, who then creates a proxy certificate after adding the attribute certificate as a noncritical extension. The user utilizes this certificate to place a request to access a resource.

VOs with different authorization mechanisms can become interoperable by standardizing the format for policy assertions [30]. In most trusted environment, Grid Security Infrastructure covers nearly all security aspects to protect the Grid from outside threats unlike the traditional view which suffers from several security issues, especially if users may install software autonomously.

D. Benefits through Virtualization

- Complete isolated execution environments
- No disclosure of meta-data
- Enhanced security of sensitive data
- Secure and individual software installation is possible
- Usage of shared resource possible, Dynamic and fine grained network control through VO based firewalling,
- Enhanced OS security using TPM hardware

E. Enhancing Grid Security Using Trusted Virtualization

Currently security built into grid toolkits (e.g. the Globus toolkit) are used at the provider sites (parties that offer resources for use in the grid). Secure channels, authentication, unsupervised login, delegation and resource usage are all handled by the toolkit. These mechanisms usually do not protect the grid user (the person or entity wishing to utilize resources). The user is forced to trust the provider often without the possibility of verifying whether that trust is justified. Current literature on grid security ensures that the user is not regarded as trustworthy. This trust asymmetry could potentially lead to a situation in which the grid provider causes large damage to the user with little risk of detection or penalty. These problems are most evident in computational grids, other such as storage or sensor grids also suffer from the negative consequences of this trust asymmetry. Because of this problem companies are reluctant to utilize available grid resources for critical tasks [18].

A trustworthy grid environment that enforces multilateral security would offer a number of benefits and even sensitive computations could be performed on untrusted hosts. Most

personal computers used today possess computing abilities in excess of what is required for casual or office use. A large percentage of the platforms in large-scale grids are built using general-purpose hardware and software. However, it is easy and cheap for existing platforms to incorporate a Trusted Platform Module (TPM), based on specifications of the Trusted Computing Group (TCG). The module provides a trusted component, usually in the form of a dedicated hardware chip. The chip is already incorporated into many newly-shipped general-purpose computers. The TPM chip is tamper-evident (and ideally, tamper-resistant) hardware that provides cryptographic primitives, measurement facilities and a globally unique identity. One approach to securing computing systems that process potentially malicious code (such as in many number-crunching grid applications) is to provide a virtualized environment. This technique is widely used for providing V-Servers (servers running several virtual machines that may be rented to one or several users).

Although the users would have full control over the virtual environment, they cannot cause damage outside that environment. Since virtualization offers abstraction from physical hardware and some control over process interaction, there still are problems to be solved, such as the x86 architecture, direct memory access (DMA) devices can access arbitrary physical memory locations. However, hardware innovations such as Intel‘s Trusted Execution Technology (formerly known as LaGrande) and AMD‘s Virtualization Technology (formerly code-named Pacifica) aim to address these problems and could eventually lead to secure isolation among virtual machines. Virtualization technology can be leveraged for building a trustworthy grid environment, especially because several works, such as [33] have already begun to consider architectures that feature policy enforcement in the virtualization framework.

F. Virtualization Vulnerabilities

Virtualization platforms are software and all software have flaws. The major virtualization platform vendors, VMware, Xen (now Citrix) and Microsoft all have several vulnerabilities. The following are major areas of concern for security professionals.

1) Hypervisor security - The hypervisor also called a virtual machine manager is a program that allows multiple operating systems to share a single hardware host. The hypervisor controls host processor and resources, allocating what is needed to each operating system in turn and making sure that the guest operating systems (called virtual machines) cannot disrupt each other. The security implications of these actions, access to the virtualization management system should be restricted to authorized administrators only. Most hypervisor software currently only use passwords for access control hence easy for attackers to penetrate the software that orchestrates the whole virtual environment (i.e. Hypervisor) and take control of every virtual machine under its control and all the data stored on them.

2) Host/Platform Security – The biggest security pitfalls with virtualization today on Intel/AMD/Compatible server platforms is that, a VM product has to be able to (a) Support a number of differing operating systems (OS) (e.g. Windows, Linux & UNIX) and differing OS distributions (e.g. Linux distributions), and (b) Interwork with a number of differing third party security products.

3) Securing Communications - Securing of Data security concerns is regarding the data that is being or will be transferred from one server to the other. This data transfer is very difficult to track and thereby to protect. If the firewalls and anti-theft software programs are not kept in place there is a very good chance that the servers can be hacked. Since there is no way to detect the data transfer, the servers are particularly vulnerable. To combat this problem there are certain antivirus programs that are designed especially for protecting against server data theft and also platforms that can support this communication such as SSH, SSL and IPSec for any communications.

4) Security Between Guests - One of the biggest security issues facing the virtualized enterprise revolves around the lack of visibility into traffic between guests. In a virtualized environment, lots of traffic might occur within the hypervisor without ever making it being protected and securing the physical network with intrusion detection systems, malware scanners, etc. This is dangerous unless these security systems are configured to protect the virtual networks. Other issues concern possible data breaches, whereby data which once resided in secure environment could be inadvertently moved into an insecure environment by the touch of a button; this is usually due to a lack of education, segregation of duties and lack of processes and controls [14].

In summary, security challenges in a Grid environment can be addressed by categorizing the solution areas:

- Enhancing Grid Security Using Trusted Virtualization [34].
- Interoperability solutions so that services hosted in different virtual organizations that have different security mechanisms and policies will be able to invoke each other; and
- Solutions to define manage and enforce trust policies within a dynamic Grid environment.

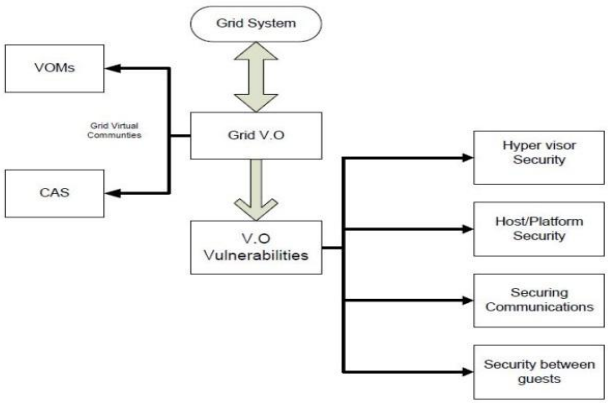

Fig. 1 Model of Grid Virtualization Vulnerabilities

7. DESIGN OF PROPOSED MODELS

The following are proposed solutions to each vulnerability as illustrated in fig. 1 above.

A. Hypervisor Security

To ensure that the hypervisor does not disrupt with other operating systems a security services should be employed. Security is an important requirement while accessing or transferring files to ensure proper authentication of users, file integrity and confidentiality. Security can be divided into two main categories: authentication of users and encryption of data transfer. Authentication can be based on either passwords or symmetric or asymmetric public key `cryptographic protocols such as Kerberos. Data encryption may be present or absent

within a transfer mechanism. The most prevalent form of data encryption is through SSL (Secure Sockets Layer).

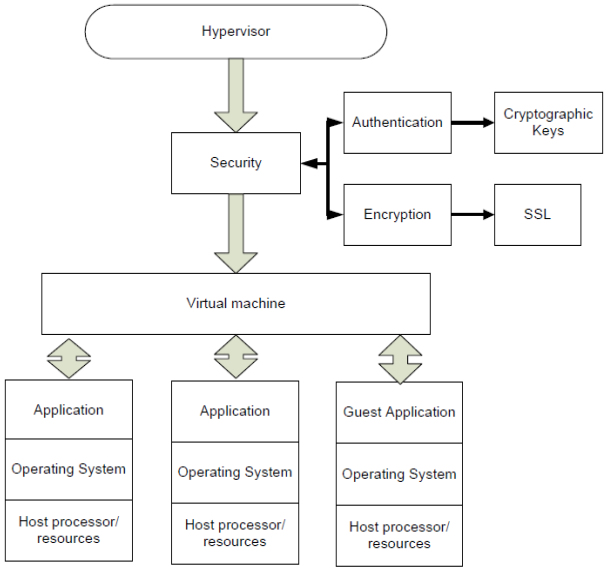

Fig. 2 Model of proposed hypervisor security

B. Host/Platform Security

Host/ platform security can be solved when the virtual machine can support differing operating systems, host processor and resources and interwork with third party service.

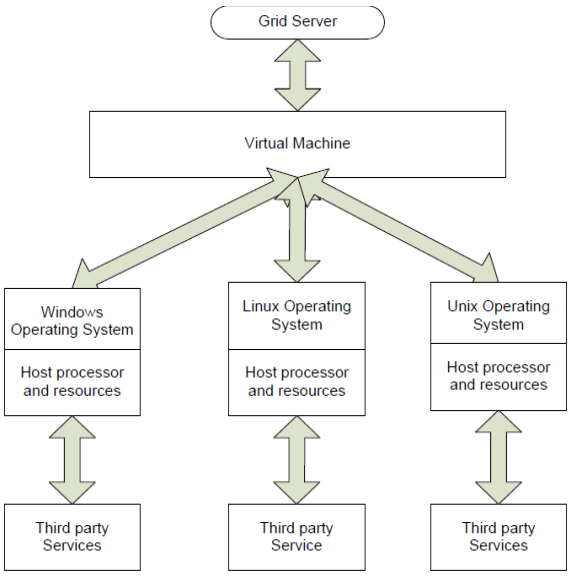

Fig. 3 Model of proposed Host/Platform Security

C. Securing Communications

Securing communications between two servers can be made possible if Firewall and antivirus programs are designed especially for protecting against server data theft and also platforms that can support this communication such as SSH, SSL and IPSec for any communications.

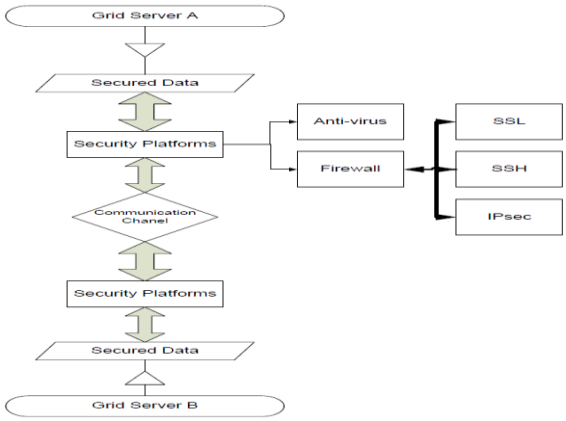

Fig. 4 Proposed methods of Securing Communications

D. Security between guests

Security between hypervisor and guest can be solved when security systems such as intrusion systems and malware scanners are configured to detect the traffic between guests.

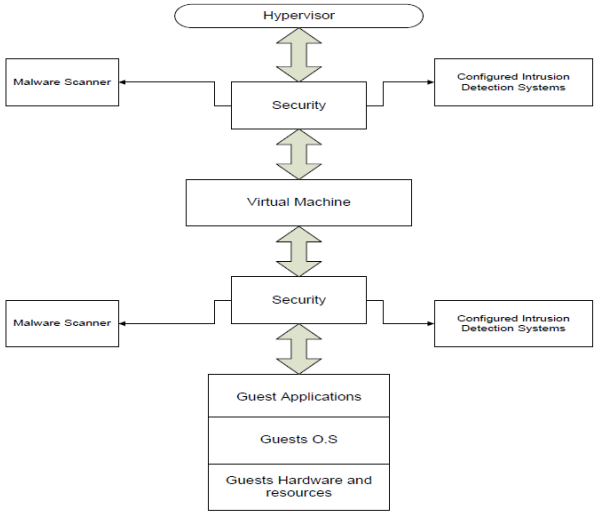

Fig. 5 Security between guests

8. CONCLUSION AND FURTHER WORK

Grid computing remains almost known in most I.T and computing environment hence improving on its security measures so as to maintain its usage. Therefore the first recommendation is the encouragement of I.T managers and businesses about the new ways of improving its security measures through virtualization. Virtualization designed to secure communication supporting heterogeneous platforms, with strong hypervisor security services and configured to detect the traffic between guests could offer a panacea to most the security vulnerabilities in Grid Computing.

There are many areas in Grid computing that could be researched namely the implementation of XSLT programming language, Grid Nomadic Migration and investigating how to combine Service-Oriented Architecture (SOA) and remote code execution programming models.

9. REFERENCES

[1] (n.d.). *Retrieved from www.gridcomputingplanet.com.*

[2] (n.d.). *Retrieved from SETI@home.*

[3] C. Strachey. (1959). *Time sharing in large fast computers. In Proceedings of the International Conference on Information Processing, (pp. 336-341).* UNESCO.

[4] (2011). *Retrieved from energypedia: https://energypedia.info*

[5] A. Berman, V. Bourassa, E. Selberg. (1995). *TRON: Process-Specific File Protection for the UNIX Operating System. In Proc. Usenix Technical Conf. UNIX and Advanced Computing Systems*, (pp. 14–24).

[6] A. Nadalin et al. (2006). *SOAP Message Security 1.1. In Web Services Security. OASIS standard specification.*

[7] (2003). *Advanced Reservation: State of the Art.*

[8] Akshay Luther, Rajkumar Buyya, Ra jiv Ranjan, and Srikumar Venugopal. (2005). *A .NET -Based Enterprise Grid Computing System. 6th International Conference on Internet Computing (ICOMP'05)*. Las Vegas, USA.

[9] Albeaus Bayucan, Robert L. Henderson, Casimir Lesiak, Bhroam Mann, Tom Proett. (1999). *Portable Batch System. MRJ Technology Solutions.*

[10] Aurther, M. R. (2011, September 23). *Programmer*.

[11] (2000). In E. Bötsch, *UNICORE Certification Authority (U-CA), (Version 1.3).*

[12] C. Kesselman and I. Foster. (1998). *The Future of High Performance Distributed Computing.*

[13] Condor High throughput computing. (2011). *Retrieved from www.cs.wisc.edu: http://www.cs.wisc.edu/condor*

[14] Dave Shackleford. (2010, March 16). ProSecurityZone. *Retrieved from www.prosecurityzone.com: http://www.prosecurityzone.com/News/Education__training_and_professional_services/Exhibitions_and_trade_shows/Understanding_virtualization_vulnerabilities_12786.asp#ixzz1bip9sOYX*

[15] David Munoz Sanchez. (2010). *Security solutions in Cloud and Grid Computing.*

[16] Ernst Bötsch. (2001). *UNICORE CA (U-CA) Policy –. In Zertifizierungs-Richtlinien für UNICORE.*

[17] Fathers of the Grid.

[18] Foster, I., Kesselman, C., Tsudik, G., Tuecke, S. (1998). *A security architecture for computational grids*. 5th ACM Conference on Computer and Communications Security, (pp. 83–92).

[19] Fredrik Hedman, KTH. (2007, February). *Interoperability Security Concerns*. San Diego, California, USA.

[20] I. Foster. (2002). *A Three Point Checklist. In what is the Grid?*

[21] I. Foster and C. Kesselman. (1999). *In the Grid: Blueprint for a New Computing Infrastructure*. San Francisco, USA: Morgan Kaufmann Publishers, Inc.

[22] I. Foster and C. Kesselman. (1997). *International Journal of Supercomputer Applications. In Globus: A metacomputing infrastructure toolkit* (pp. 115–129).

[23] Ian Foster, C. Kesselman, S. Tuecke. (2001*). The Anatomy of the Grid: Enabling Scalable Virtual Organizations.*

[24] IanFoster and Carl Kesselman. (2004). *Blueprint for new computing.*

[25] John W Rittongtonhouse , James M Ransom. *Cloud Computing, Management, Implementation and Security. CRC Press.*

[26] (2003). Altair Grid Technologies. In J. P. Jones.

[27] Khavaran Higher-education Institute. (2011). *5th SASTech. 5th Symposium on Advances in Science & Technology. Mashhad*, Iran.

[28] Marty Humphrey, Mary R. Thompson, Keith R. Jackson. (2005). Security for Grids.

[29] R. Ramanathan, F. Bruening. (2004.). *Virtualization - Bringing Flexibility and New Capabilities to Computing Platforms. Intel Corporation.*

[30] R. Uhlig et al. (2005). *In Intel Virtualization Technology Computer*, vol. 38 (pp. 48–56.).

[31] Robert L. Henderson. (1995). *Job scheduling under the Portable Batch. In D. F. Rudolph, Grids as production computing environments System* (pp. 178–186). Springer-Verlag.

[32] S. Anderson et al. (2005). *In Web Services Secure Conversation Language (WS-SecureConversation).* OASIS specification.

[33] Sailer, R., Jaeger, T., Valdez, E., Caceres, R., Perez, R., Berger, S., Griffin, J.L., van Doorn, L. (2005). *Building a MAC-based security architecture for the Xen open-source hypervisor*. 21st Annual Computer Security Applications Conference, *IEEE Computer Society*, (pp. 276–285).

[34] Security & Trust for the Grids. (2005). *Open Grid Forum. Overview of the GSI.*

[35] University of Wisconsin, Madison. (n.d.). *Retrieved from www.cs.wisc.edu/condor*

[36] Virtual LANs. IEEE Computer Society. (n.d.). *Retrieved from http://www.ieee802.org/1/pages/802.1Q.html*.

[37] Wenbo Mao. (2004). *Grid Security with Behaviour Conformity from.*