Grid Security Policy Monitoring System (GridSPMS): Towards Monitoring the Security Dimension in Grids

Abdulghani Suwan and Francois Siewe
School of Computer Science and Informatics
Faculty of Technology
De Montfort University
Leicester, LE1 9BH, UK
p02318242@dmu.ac.uk
FSiewe@dmu.ac.uk

Abstract: Grid computing systems are complex and dynamic environments requiring appropriate automated management mechanisms, which would enable stable and reliable operation of the whole grid ecosystem. Moreover, the recent novel concept of mobile grid computing – a combination of grid systems with mobile devices – also requires new ways of monitoring the emerging security aspects, associated with the ever-increasing role of network connections in mobile grids. Existing grid monitoring systems, albeit suitable for traditional, localised grid systems, seem to be ignoring the security dimension and do not offer appropriate support for enforcing security policies within a distributed grid system enhanced with mobile devices. Accordingly, in this paper we present the Grid Security Policy Monitoring System (GridSPMS) – a novel grid monitoring framework, which extends the traditional support for data monitoring (i.e., tools, protocols, etc.) with mechanisms for collecting run-time data within grids, focussing on the security dimension. By doing so, GridSPMS aims at monitoring the activity within a grid system and detect situations, where security policies have been potentially violated and, therefore, appropriate response actions have to be taken in this respect. Accordingly, GridSPMS has the potential to address security threats existing in the domain of mobile grid computing, and enable grid administrators with support for timely detection of and response to security-related incidents.

Keywords: Grid computing, Security policy, Monitoring System.

1. Introduction

With the emergence and the rapid development of cloud computing in the last decade, grid computing is nowadays mainly used in the context of academic research with little (or no) interest and support from the industry. Essentially, grids have hardly been seen as a commercial product to be offered to a wide range of customers (Foster et al., 2008). As opposed to cloud computing, which is widely known to be a highly commercialised and industry-driven research area, grid computing is becoming to attract less and less attention of the research community.

The situation, however, might change with the recent introduction of *mobile grids.* Mobile grid computing is an emerging computing paradigm, which lies at the intersection of two research areas – namely, grid computing and mobile computing (Litke et al., 2004). It can be seen as a re-incarnation of conventional grid systems, and its main concept is to extend the traditional capabilities of grids – that is, provisioning of a large pool of aggregated computational and storage resources in order to address computationally-intensive tasks (Foster and Kesselman, 1999) – with computational capabilities of mobile devices provisioned over the network. Broadly speaking a mobile grid can be defined as a complex scalable distributed system involving a variety of nodes some of which might be mobile, geographically distributed around the globe and using various communication protocols. From this perspective, mobile grid computing can be seen as an evolution of the concept of grids from traditional, in-premises deployments to a distributed computer architecture, consisting of both server clusters residing in a data centre and multiple mobile devices, such as smartphones, tablets and laptops, connected to the main cluster via a wireless network.

What was regarded as impractical and impossible just several years ago, today is attracting more and more investigation both from industry and academia. Indeed, portable mobile devices are becoming more and powerful in terms of their CPU and memory capabilities, and the bandwidth of 5G mobile networks is approaching 1GB per second. In this light, the potential integration of grids with mobile computing is becoming even more promising. The inherent benefit of mobile grid computing is that it allows users to access grid resources from a mobile device from virtually anywhere at any time, without the requirement to be sitting next to a terminal. Moreover, in mobile grid architectures, mobile devices are expected to act not just as resource

consumers, accessing grid resources, but also act and as resource providers in their own right, contributing to the shared pool of available resources (Katsaros and Polyzos, 2008). This latter feature may enable forming of 'ad-hoc' mobile grid compositions on the spot. In other words, it is possible to create local, short-term mobile grids at various venues and locations with an increased concentration of smart mobile devices, such as, for example, conferences and universities (Litke et al., 2004).

Along with these promising opportunities for increased computational capabilities and ubiquitous network access to computational and storage resources, come emerging challenges so as to how to manage these resulting complex environments and maintain stable quality of service (QoS). The issue of delivering promised QoS to mobile grid users is multi-faceted, and can be seen from several perspectives. In the first instance, QoS depends on the networking capabilities of a mobile grid system. Mobile network bandwidth remains the main factor to guarantee smooth and undisrupted data exchange between mobile devices and grid servers. Another important concern in satisfying QoS requirements is the implementation of the resource management mechanism responsible for resource discovery and selection, job scheduling and replication, data migration and monitoring (Bichhawat and Joshi, 2010). In the presence of a considerably higher number of computational nodes constituting the mobile grid system, it is important to organise how various computational tasks are split, scheduled and executed in an efficient and scalable manner.

The last but not the least, the emergence of mobile grid computing poses new challenges as to how these distributed systems should be properly protected and secured. As opposed to traditional grid architectures, where networking is not regarded as one of the primary issues to take into account, in mobile grids the networking dimension plays a dominant role. The wireless nature of network connections introduces new threats and makes resulting mobile grid systems vulnerable to a wide range of threats such as eavesdropping, data tampering and data tracing (Bichhawat and Joshi, 2010). Accordingly, novel appropriate monitoring and detection mechanisms are required in order to address these security issues and enable mobile grid systems with sufficient self-protective capabilities to maintain required QoS and at the same time provide sufficient scalability – a key feature of mobile grids.

Given these considerations, in this paper we argue that a potential way of supporting security management in mobile grid computing is to monitor grid activities with a goal to detect and report security-related incidents, followed by a generation of an effective security incident response plan. Accordingly, we also present and explain our own *Grid Security Policy Monitoring System* (GridSPMS), which is a framework to monitor security policy compliance in grids, detect potential violations, and report on the detected incidents. We provide a high-level architecture of the framework and discuss its potential benefits and limitations.

Accordingly, the rest of the paper is organised as follows. Section II introduces and briefs the reader on the security issue in grid computing in general and in mobile grids in particular. The section also provides a state-of-the-art overview in the considered research area and identifies existing limitations and gaps. To address the identified gaps, Section III presents our proposed GridSPMS and describes its conceptual architecture in a top-down manner – a conceptual architecture is followed by a more fin-grained description of individual components. Section IV contains some concluding remarks and summarises this paper.

2. Motivation: Insufficient Security in Mobile Grids

Arguably, security (albeit a major research challenge in IT in general) has never been one of the primary concerns in grids. Historically, grids have been associated with relatively small-scale, localised deployments within a single data centre (Bessis et al., 2010), where number of network connections was limited and stable. Also, they have been primarily serving scientific purposes, and therefore the number of users accessing grid resources was typically limited and could be easily controlled, and, therefore, there was no real pressing demand for enforcing various security checks. Moreover, grids have not been seen as a commercial product to be offered to a wide range of customers (Foster et al., 2008), and, consequently, there was relatively little interest and support from the industry. As a result, development of grids, including investigation of security-related issues, has been primarily driven by academic researchers.

For example, Foster et al. (Foster et al., 1998) were among the first to realise and address the challenge of delivering security in grid systems. The authors proposed and implemented an architecture for secure grids. This work established theoretical and practical foundations for designing and implementing grid environments in a secure, robust and reliable manner, and the proposed architecture has become a reference model for other approaches aiming to achieve high levels of security in grids.

However, implementing grid systems in a secure manner following the proposed reference architecture on its own is not enough. An inherent requirement to achieve sufficient levels of security in grids (and in computing

systems in general) is to implement appropriate monitoring mechanisms, which would enable prompt and timely detection of potential violations. Such monitoring mechanisms would collect various security-related data within grids, analyse with respect to a set of security policies, and detect if there any violations calling for immediate responsive action from the human administrator – taken together, we refer to these processes as *security policy monitoring*.

This latter requirement of implementing security policy monitoring mechanisms, however, has been (and still is) beyond the common capabilities of existing grid monitoring systems. As indicated by our recent survey (Suwan et al., 2016), existing grid monitoring frameworks seem to neglect the security dimension. Their capabilities to be extended so as to integrate security-related monitoring metrics and policies are also limited

This is becoming a particularly pressing challenge with the rapid development of mobile technologies and the emergence of mobile grid computing, which can be seen as part of the global paradigm shift in IT. From personal desktop computers we are moving towards portable mobile computing, supported by the powers of clouds. In 2011, smartphone sales exceeded PC sales for the first time in the history (Cocotas, 2012), and as it was recently declared by Apple CEO Tim Cook, the PC is now officially dead (Ghosal, 2015). Admittedly, with this mobile revolution the mobile security issue has been attracting much attention. The market of portable and mobile devices would not be exponentially growing if it was not properly supported by advances in security and data protection. Accordingly, as the mobile technology was advancing and the number of mobile devices was growing, the research and industrial community has been consequently putting more and more efforts into investigating possible approaches to provide efficient and secure solutions for mobile platforms. In the 1990s the mobile technology was in its infancy and security issues were among the minor concerns with little or attention. Then, in the 2000s, the community focused on enabling mobile platforms with reactive self-protective mechanisms, which would fix potential harmful consequences of detected malware. And now we are seeing how more and more proactive mechanisms are attempting to predict and prevent possible attacks and threats before they even happen.

These advances in mobile security, however, have not been yet applied to the domain of mobile grid computing, and in the next section of this paper we are presenting Grid Security Policy Monitoring System (GridSPMS) – our own attempt to create a grid monitoring framework fully supporting the security dimension.

3. GridSPMS: Conceptual Architecture

In order to address the identified gaps in the domain of mobile grid security, we are currently developing our own *Grid Security Policy Monitoring System* (GridSPMS). This system is built on top of an already existing grid monitoring framework Ganglia[1] and extends its functionality with the support for the security dimension. Before presenting the actual design and architecture of GridSPMS, we first outline a (non-exhaustive) list of desired important features, which were taken into consideration and influenced the design of the final system.

These desired features, which concern both functional and non-functional properties, include:

- *Support for monitoring and enforcing security policies* is a fundamental feature of the envisaged monitoring framework. In a broad sense, a security policy is a rule which defines and enforces certain constraints on grid resources. For instance, security policies can be applied to limit access to memory and CPU resources per user, restrict from creating excessive jobs, etc. Typically, security policies may target either individual users, or whole user groups (i.e., user roles).
- *Support for security incident response policies* is another key feature of the proposed grid monitoring framework. Security incident response policies describe reactive actions which need to be taken whenever a security policy is violated – that is, whenever a security incident takes place. In the context of grids, an example of a security incident may be unauthorised access to computational resources, excessive consumption of resources, accessing resources using an insecure network connection, etc. Accordingly, security incident response policies may define reactive actions ranging from simple reporting to the human administrator to more sophisticated actions such as automatically applying appropriate actions to bring the system back to the initial stable state. The former is typically referred to as a *passive* response, and the latter is known as an *active* response.

[1] Ganglia is a scalable distributed monitoring system for high-performance computing systems such as clusters and grids (http://ganglia.sourceforge.net/).

- *Ability to seamlessly integrate with Ganglia* refers to our intention to re-use the positive experience of an already existing and efficient grid monitoring framework without 're-inventing the wheel'. In a sense, we intend to develop a fully compatible extension to the core Ganglia functionality, which will be able to benefit from data collected and supplied by Ganglia to offer additional security-related capabilities.
- *Ease of deployment* is a requirement closely related to and stemming from the previous one. Inheriting Ganglia's functionality, the proposed monitoring system is expected to remain easily deployable and configurable within grid environments.
- *Information abstraction* means that the monitoring system is able to integrate, synthesise and interpret collected information so that only relevant is displayed to the user. Additionally, users are not supposed to interact with the back-end mechanism, which will be transparently abstracted by the front-end interface (i.e., web browser).
- Dynamic information retrieval and near real-time operation are two features, which refer to the system requirement to function in a timely manner. Data, collected by the GridSPMS, is highly dynamic in its very nature, and the system has to be properly equipped so as to collect and process this data with minimum delay. This in turn will facilitate prompt detection of potentially critical situations and timely reporting on security incidents. In such data-intensive environments as (mobile) grids where observed values may become obsolete and outdates within seconds, these requirements are seen as one of the primary challenges.
- *Uniform data representation* is expected to overcome existing heterogeneity in data representation formats. There has to be a common vocabulary of terms – that is, a standard way of representing information in the context of managed grid environments, which may include, for example, various grid components, system users, geo- location, historical values, security incidents, etc.
- *Support for scalability* is an inherent requirement of the grid domain, which is characterised with a highly distributed environment, consisting of a large number of computational nodes and clusters. With the introduction of mobile grids, this challenge is taken to a completely new level, where data has to be collected from multiple remote mobile nodes over the network. The envisaged monitoring framework is expected to support integration of newly-added nodes in an automated manner, so that changes are immediately reflected in the system in a transparent and seamless manner.
- High performance and robustness can be seen as pre-requisites to enable scalability in GridSPMS, which should take into account the demands of a distributed environment to enable failure-free execution. This also implies that grid elements (i.e., individual nodes or whole clusters), which are currently unavailable, crashed or under-performing entities should be safely isolated from the 'healthy' grid to support the stability of the whole grid ecosystem.
- *Ubiquitous remote access over the Internet* is another inherent requirement of mobile grids. With the integration of portable mobile devices into the grids, it is also essential to enable remote access to the presentation layer of the monitoring system via some kind of a 'mission control'. In other words, a user is expected to remotely access and see the grid monitoring dashboard from any remotely-located device at any time. Apart from viewing the current status of the monitored grid system, the user is also supposed to be able to manage and control the system.

3.1 Four-layered conceptual architecture of GridSPMS

We now proceed describing the actual architecture of the proposed monitoring system in a top-down manner. First, we present a high-level conceptual overview of GridSPMS, consisting of four layers, and then go into more details by explaining individual components constituting the system.

Figure 1 schematically depicts the proposed four-layer architecture of the GridSPMS. These conceptual layers are interdependent, so that upper layers build upon and rely on the lower layers. Such organisation also reflects the established structure of modern grid systems, where low-level monitored data is first collected from individual nodes, and then proceeds through a series of abstractions to represent an aggregated and synthesised view on a cluster of nodes, and, finally, on the whole grid. This approach represents the grid as a hierarchical network of clusters and nodes and facilitates scalability. Accordingly, the information flow can be represented by the following triple:

Node → Cluster → Grid

The triple also helps to uniquely identify each individual element within GridSPMS.

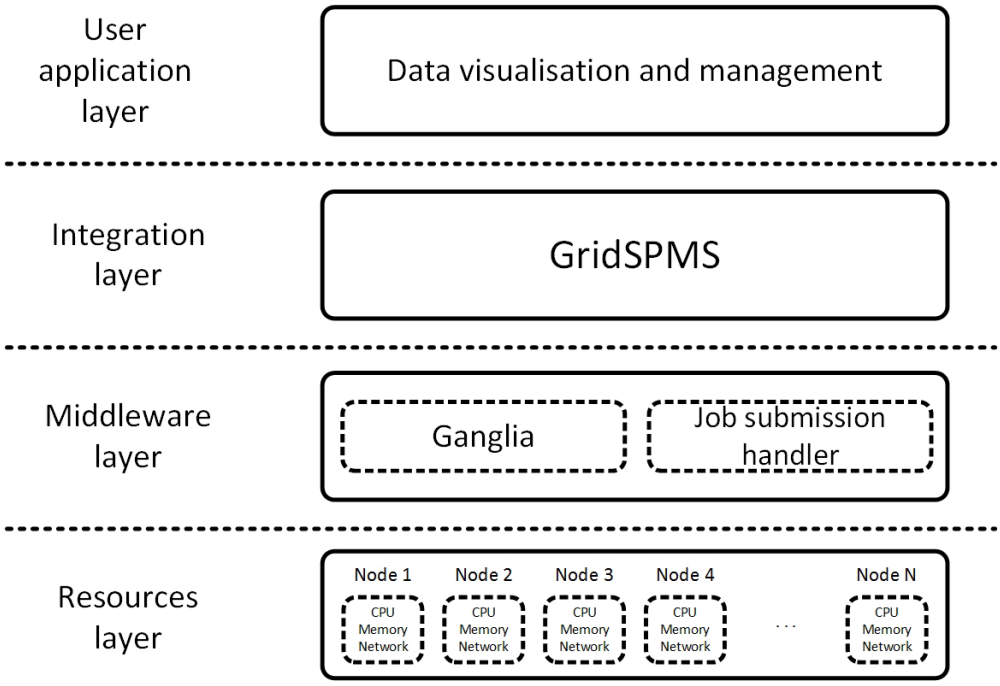

Figure 1: Four-layered architecture of GridSPMS.

- *Resources layer* represents monitored resources (such as individual nodes, network connections, storage devices, etc.), equipped with a Ganglia agent to collect relevant data. Monitored metrics typically include the general availability of a particular resource, as well as CPU and memory utilisation, network statistics, occupied storage space, etc.
- *Middleware layer* represents the Ganglia framework, which is responsible for a wide range of activities, such as resource management and grid monitoring (which is the most important feature in the context of the presented research). Ganglia's monitoring depends on daemon processes called *gmetad* or *gmond*, which serve to collect data about grid nodes. Ganglia monitoring capabilities can be extended with the *Ganglia metric tool* (*gmetric*) which enables monitoring of arbitrary host metrics by expanding the core set of metrics measure by *gmond* by default. Simply put, users can develop their own metrics and deploy them into the monitored grid system to extend the basic monitoring capabilities. Another important component is the *Job Submission Handler*, which is responsible for resource management and job scheduling. This component is capable of checking whether resources are available and users are authorised to access them. Together with the other standard metrics and extended custom metrics, it is possible to extract statistical data per user (i.e., number of allocated nodes, CPU/memory usage, network traffic, user IP address, etc.) from job submission handler and deliver them to Ganglia in an adapted format.
- *Integration layer* is where GridSPMS is located. Our proposed system receives relevant data from Ganglia and implements mechanisms for detecting a security incident and consequent responding to it. GridSPMS consists of four main components – namely, Data Collection, Breach Detection Engine (BDE), Security Policy Database (SPD) and Incident Response Module (IRM) – which will be further described in the next section.
- *Client application layer* represents interfaces through which GridSPMS users can interact with the system. These include an Application Programmable Interface (API), Command Line Interface (CLI), and Web-based Graphical User Interface (GUI). These interfaces are intended to enable users to interact with the system and, if necessary, extend its functionality and integrate in their own software systems (e.g., in science, finances, engineering applications).

3.2 GridSPMS components

Having presented a high-level overview of GridSPMS, we now proceed with a more detailed description of its internal organisation and its individual components.

As it was previously explained, GridSPMS is built on top of Ganglia, which provides the basic tools for resource monitoring in computational grids. From this perspective, it can be seen as an extension to Ganglia, which complements the existing functionality with the support for the security dimension. The diagram in Figure 2 schematically depicts the main components of GridSPMS and their interaction between each other. We distinguish between 12 main components, which we now consider one by one in more details.

It is worth mentioning that the presented component architecture has been inspired by the Common Component Architecture (CCA) (Armstrong et al., 1999) – a reference model which defines how complex computing systems have to be designed and implemented in a modular manner. This architecture has been widely used in the design of cluster and grid systems, where individual components are seen as basic building blocks used to construct a complex system.

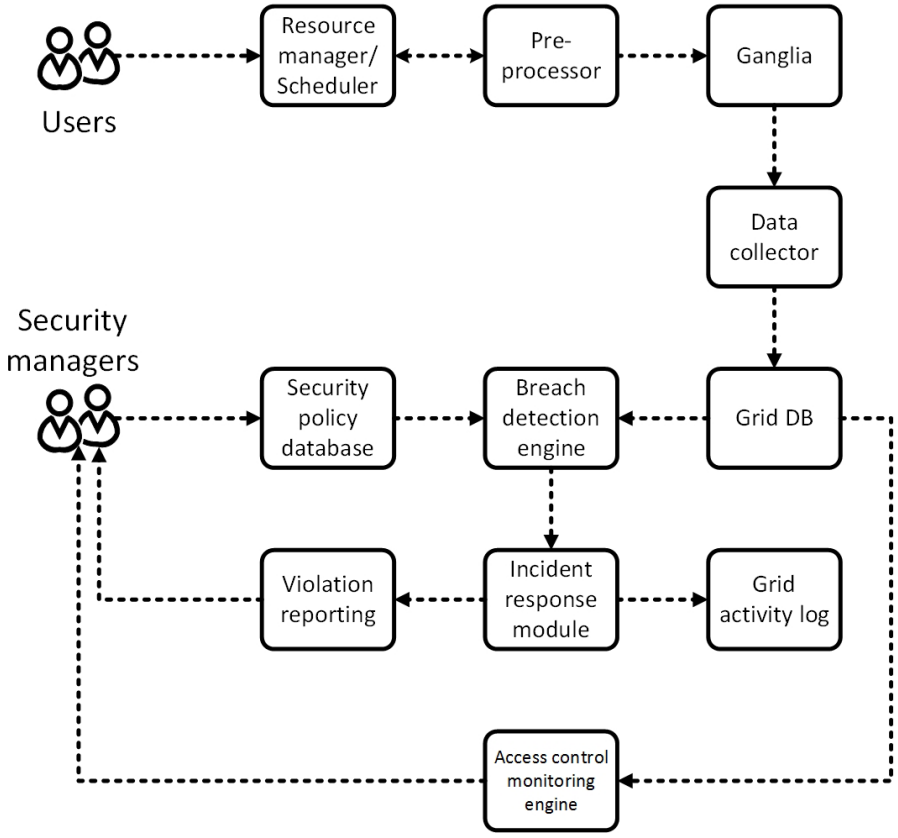

Figure 1: Components constituting GridSMPS at various layers.

- *Resource manager/Scheduler* is part of the core functionality of contemporary grid platforms, which is responsible for i) receiving task requirements from the user, ii) allocating and provisioning appropriate resources to the user, iii) scheduling application jobs and optimising resource utilisation, and iv) monitoring job execution processes. All these activities take place provided the user is authorised to access and use corresponding grid resources.
- *Pre-processor* is responsible for collecting user statistical data like jobs, allocated nodes, CPU/memory utilisation, as well as user IP addresses, which helps to determine user's geographical location. Pre-processor is also responsible for sending the input information to Ganglia via *gmetric* tool.
- *Ganglia* typically includes one or more instances of *gmetad* daemon processes, which are responsible for collecting data about managed grid clusters or nodes. The server running a *gmetad* daemon process constantly sends XML packets over the TCP connection. These packets contain information about the current state of individual nodes and clusters within grids. Depending on the types of *gmond or gmetad* daemon processes, which are responsible for the actual extraction of data, the XML

data includes various fields related to the actual performance of a particular node or cluster and behaviour of the users. Ganglia is also designed to be extensible so that users are also free to develop and integrate their own extension to the core system – a feature which proved to be helpful in the context of our own research. Accordingly, we are also able to extract and send user geo-location data as an XML packet.

- *Data collector* is a component, which periodically polls and receives XML packets sent over the network from Ganglia and parses this information. Depending on the predefined rules it filters, and aggregates the obtained information and persists into the Grid database. At this stage, security concerns are becoming to play an important role, as data collector is now specifically looks at the data related to potential security policy breaches. In this sense, it can be seen as a mediator between the core Ganglia component and GridSPMS.
- *Grid database* is a usual relational database MySQL, which are responsible for storing data received by the data collector from Ganglia. The persisted data is then available to other components of GridSPMS (i.e., BDE and ACME) and perform administrative tasks via the standard SQL queries.
- Breach detection engine (BDE) is a core analysis component of GridSPMS, which queries data from the database and matches it against a set of pre-defined security policies with a goal to detect potential security breaches. Situations where the collected data fails to satisfy the policy constraints are classified as security breaches, and a corresponding security incident has to be reported and/or corresponding response actions have to be taken.
- Security policy database serves to store rules, regulating the security-related activities within the grid system and used by BDE. We assume that security policies are designed and implemented by *policy managers* – human administrators and domain specialists, who also act as supervisors when responses to incidents are reported (i.e., passive responses) and executed (i.e., active responses).
- *Incident response module* serves to report, manage and document all detected grid security incidents. In case of emergency (i.e., a security breach is detected), the incident response module will first notify the human administrator and/or may take certain actions in a more autonomic manner.
- *Violation reporting component* is a simple module, whose only responsibility is to notify the human administrator of the detected security breach via one of the available communication channels (e.g., an e-mail, a sound signal, a pop-up window in the administration panel). A notification message contains information about the violation, including detection time, potential cause of violation, and a user associated with the incident.
- *Access control monitoring engine* (ACME) is a component responsible for collecting data, which concerns accessing the system. It constantly monitors user activity, such as login attempts, user locations and IP addresses, managing user accounts and access rights, etc.
- *Grid activity log* is a component, where all the activities taking place within the managed grid environment are recorded and stored. The storage period may range from few days to several years, depending on the requirements.

4. Conclusion

As we have claimed, grid computing systems are complex and dynamic environments requiring appropriate automated management mechanisms, which would enable stable and reliable operation of the whole grid ecosystem. The research community has addressed this requirement with a number of monitoring frameworks, which serve to collect data at various levels to support decision taking and management activities within grids. However, these existing solutions seem to implement little support for collecting security-related data and enforcing appropriate security policies and constraints in this respect. With the emergence of mobile grid computing, an increasing role of network connections, and a growing number of users remotely accessing computational resources from various locations, grid systems are no longer seen as localised and isolated ecosystems, but are coming to be more open and distributed. In this light, it is becoming more and more important to enable monitoring framework with capabilities to collect security-related data and check whether these observations comply with certain security constraints – that is, to monitor security policies in grids.

Accordingly, in this paper, we presented our work in progress, which aims at creating an efficient solution for enabling monitoring the security dimension in grid systems. The proposed GridSPMS is an extension to Ganglia

and attempts to build on its already existing and efficient capabilities for monitoring various types of data within grids to additionally enable capturing some security-related metrics and match them against a set of security policies.

We have described the proposed architecture in a top-down fashion, starting from a high-level overview of four main levels constituting GridSPMS. Then, we focused on the actual internal organisation and presented its main functional components. More specifically, we explained how the core functionality of Ganglia could be extended so as to incorporate collecting security-related data (e.g., user geo-location) to support compliance checking against a set of security policies. The resulting system is expected to be capable to detect and report on various security-related accidents – an increasingly important feature in the context of emerging mobile grids.

At the moment, the presented paper describes our work in progress, which has been primarily focussing on developing the architectural design of GridSPMS. As a next step we are considering implementing a prototype version of the proposed system, integrating it with Ganglia and validating in an existing grid environment.

References

Armstrong, R., Gannon, D., Geist, A., Keahey, K., Kohn, S., McInnes, L., Parker, S., et al. (1999), "Toward a common component architecture for high-performance scientific computing", *Proceedings of The Eighth International Symposium on High Performance Distributed Computing,* pp. 115–124.

Bessis, N., Asimakopoulou, E., French, T., Norrington, P. and Xhafa, F. (2010), "The big picture, from grids and clouds to crowds: a data collective computational intelligence case proposal for managing disasters", *Proceedings of 2010 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC),* IEEE, pp. 351–356.

Bichhawat, A. and Joshi, R.C. (2010), "A Survey on Issues in Mobile Grid Computing", *Int. J. of Recent Trends in Engineering and Technology*, Vol. 4 No. 2.

Cocotas, A. (2012), "Smartphone Market Forecast: Sales Will Exceed 1.5 Billion Units A Year By 2016", *Business Insider Australia*, 1 March, available at: http://www.businessinsider.com.au/smartphone-market-forecast-sales-will-exceed-15-billion-units-a-year-by-2016-2012-2 (accessed 21 November 2015).

Foster, I. and Kesselman, C. (Eds.). (1999), *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

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

Foster, I., Zhao, Y., Raicu, I. and Lu, S. (2008), "Cloud computing and grid computing 360-degree compared", *Grid Computing Environments Workshop, 2008. GCE'08*, IEEE, pp. 1–10.

Ghosal, A. (2015), "Tim Cook says the PC is dead, but Apple still makes computers", *The Next Web*, November, available at: http://thenextweb.com/insider/2015/11/10/tim-cook-says-the-pc-is-dead-but-apples-still-making-desktops-and-laptops/ (accessed 21 November 2015).

Katsaros, K. and Polyzos, G.C. (2008), "Mobility-Aware Grid Computing", *The Encyclopedia of Information Science and Technology*, Second Edition, IGI Group.

Litke, A., Skoutas, D. and Varvarigou, T. (2004), "Mobile grid computing: Changes and challenges of resource management in a mobile grid environment", *Proceedings of 5th International Conference on Practical Aspects of Knowledge Management (PAKM 2004)*.

Suwan, A., Siewe, F. and Abwnawar, N. (2016), “Towards Monitoring Security Policies in Grid Computing: a Survey”, submitted to *IEEE Technically Sponsored SAI Computing Conference 2016*.