Advanced Vulnerability Assessment Tool for Distributed Systems

Sandor Acs[1,2], Miklos Kozlovszky[12], and Peter Kotcauer[1,2]

[1] University of Obuda, H-1300 Budapest, P.O. BOX 267, Hungary
acs.sandor@biotech.uni-obuda.hu
[2] MTA SZTAKI Computer and Automation Research Institute, H-1518 Budapest, P.O. Box 63, Hungary

Abstract. Large-scale high performance systems have significant amount of processing power. One example of such system is the HP-SEE's HPC and supercomputing infrastructures, which is geologically distributed, and provides 24/7, high performance/high throughput computing services primarily for high-end research communities. Due to the direct impact on research and indirectly on economy such systems can be categorized as critical infrastructure. System features (like non-stop availability, geographically distributed and community based usage) make such infrastructure vulnerable and valuable targets of malicious attacks. In order to decrease the threat, we designed the Advanced Vulnerability Assessment Tool (AVAT) suitable for HPC/supercomputing systems. Our developed solution can submit vulnerability assessment jobs into the HP-SEE infrastructure and run vulnerability assessment on the infrastructure components. It collects assessment information by the decentralized Security Monitor and archives the results received from the components and visualize them via a web interface for the local/regional administrators. In this paper we present our Advanced Vulnerability Assessment Tool, we describe its functionalities and provide its monitoring test results captured in real systems.

Keywords: distributed computing, HPC, security, vulnerability assessment, HP-SEE, supercomputing infrastructure.

1 Introduction

1.1 Large Scientific Computing Resources

Core European e-Infrastructure for large-scale e-Science research consists of distributed computing infrastructure (DCI), distributed storage infrastructure (DSI) and backbone (e.g GANT) network. Large number of initiatives and projects builds up the infrastructure. The SEE-GRID [1] initiative with three consecutive projects (SEE-GRID, SEE-GRID2 and SEE-GRID-SCI) provided the local grid based DCI in the SEE (South-East European) region. The HP-SEE project [2] links together existing and upcoming HPC facilities in the region in a common infrastructure to open up the HPC infrastructures to a wide range

M. Dulea et al. (eds.), *High-Perf. Comp. Infrastr. for South East Europe's Research Communities*, Modeling and Optimization in Science and Technologies 2,
DOI: 10.1007/978-3-319-01520-0_17,

of new user communities, including those of less-resourced countries facilitating cross-border research and collaboration.

1.2 Motivation

There are lots of potential security problems with distributed and shared systems caused by the technology itself and by site/software stack setup or bring on by end-user behaviour.

Our general goal is to provide assistance for system administrators building up sustainable and less vulnerable infrastructure and survive cyber-attacks. For this aim we have created a vulnerability assessment framework for distributed systems in order to decrease threats.

In this paper in Section II we briefly introduce recent vulnerability trends. In Section III we provide information about existing security monitoring solutions. In Section IV we give detailed description about the design and implementation of the developed Advanced Vulnerability Assessment Tool. Later on in Section V we show vulnerability test results of AVAT. At the end of our paper we conclude and explain directions of our future work.

1.3 Vulnerability Trends

The software evolution (size, functionality set, etc.) generally increases complexity in the software stack. It is hard to protect even a single PC node infrastructure against malicious attacks. This problem multiplies significantly if the infrastructure is heavily distributed, contains thousands of cores, and serves hundreds of people. Security issues are constantly explored worldwide and published on the relevant on-line sites as Common Vulnerabilities and Exposures (CVE) [3]. Common Vulnerability Scoring System (CVSS) Initiative [4] was funded by the U.S. Department of Homeland Security in order to provide vendor independent and common scoring system of known vulnerabilities. The scores are integer numbers between 0 and 10.

We analysed the CVSS databases and we can point out that the highest threat was from 2006 to 2008 in the last 10 years. Figure 1 shows the number of published vulnerabilities and the total severity, which was calculated from the number of vulnerabilities multiplied by their CVSS score.

Figure 2 presents that the average of CVSS score is about 6 and their standard deviation is about 2 in every year in the investigated period. One of the biggest problem is, that the successful break-in method (which are definitely very hard to find cyber-attack solutions) against any of the DCI infrastructure elements is very likely reusable and provide vast gain for the intruder.

2 Related Work

The SEE-GRID projects (SEE-GRID-SCI project and its predecessors) are using two tools and three services to monitor their grid infrastructure: Hierarchical

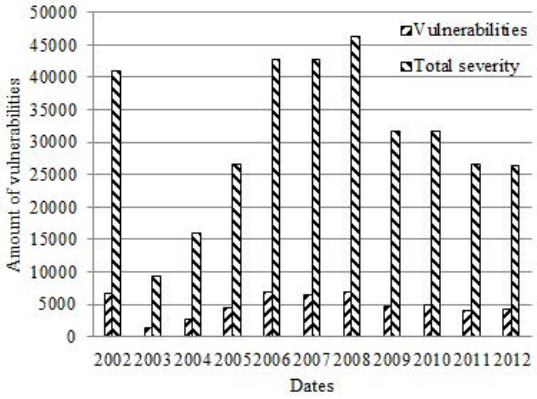

Fig. 1. Trends in number of published vulnerabilities and their total severity

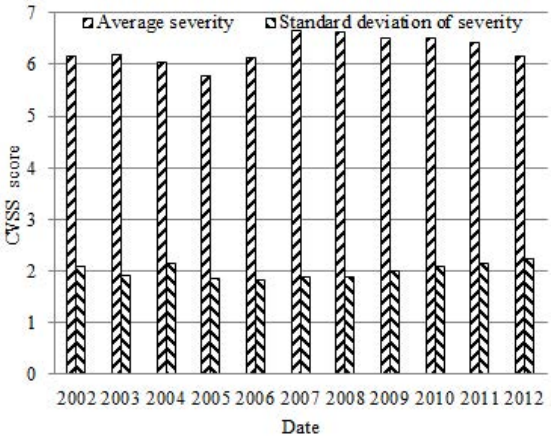

Fig. 2. Average and standard deviation of severity

Grid Site Management (HGSM) [5] that is the central information tool. HGSM is a web based database designed to hold static information about the grid sites. This information bundle includes physical location of the site, emergency contact information, operating system and middleware of the grid site, detailed hardware information of nodes and site downtimes. BBmSAM [5] portal that submits regular SAM (service availability monitoring) tests every 3 hours for interactive tests and every hour for non-interactive tests. The BBmSAM also provides data on site and service availability for the needs of the project. The PAKITI [6] service, which was used for security status monitoring of the infrastructure, collects the list of the installed software packages on the nodes and matches the gathered information with the security database (coming from an external repository). It is using HTTP or HTTPS protocol to communicate and provide a graphical user interface for its users. However the program has major serious

disadvantages in distributed IT environment (e.g grid administrators have to install the client software on every machine and and they should configure the firewall settings as well).

The Grid Site Software Vulnerability Analyzer (GSSVA) [7] tool aimed to ease the problems of PAKITI. Therefore, the GSSVA uses only essential grid protocols in order to work on every site without changes in configuration or installing any software. However, a vulnerability analyzing should have more kind of security audit procedures than only comparing software versions (that PAKITI and GSSVA does). There are some widely used closed and open-source security analyzer frameworks that can provide various types of security information about the investigated infrastructures.

Qualys Guard [8] offers a complete vulnerability management solution. The centralized front-end component is hosted by Qualys, thus the users only need a web browser to manage the target assets, schedule the vulnerability scans, generating customized reports and deal with the remediation tracking. As it is a cloud based service, the attacks are coming from the internet.

The OpenVAS (Open Vulnerability Assessment System) [9] is an open source vulnerability assessment software. It consist about three layers of components. The first layer is the user interface, where users or automatized scripts can access the system. The second layer is the services layer and the third layer is the data layer which means configurations, scanning templates and results. OpenVAS uses Network Vulnerability Test (NVT) for checking the IT resources.

The presented tools are complex and feature-rich, however they cannot be used easily in distributed systems because the HPC centers or grid sites are located on different administration domains. Therefore, the centralized management of these tools could not be satisfied.

3 Design and Implementation

We have followed three basic rules for designing the AVAT. (1) Only open-source software components shall be used to build up the system. (2) The solution should work in the users space of the targeted DCI. We shall not suppose more opportunity to reach the resources (e.g open ports on firewalls, use administrator privileges) than a generic user has, because the investigated infrastructures are located in different administration domains. Therefore, we used only DCI specific protocols and middleware services to deploy services, query/retrieve information or communicate with the DCI resource (e.g job submission). (3) Critical infrastructure assessment should always be carried out with the highest caution. Harmful tests shall not be used during the investigations on the live system to avoid service interruption.

3.1 Architecture

Figure 3 presents the high level schematic system plan with all the AVAT modules. The control module is responsible for connecting the different modules and

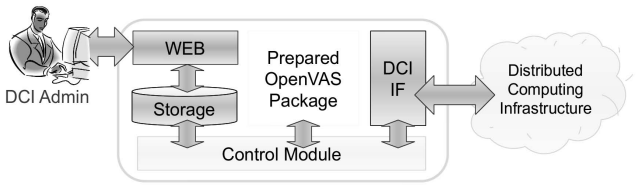

Fig. 3. AVAT modules

scheduling the vulnerability scans. The tool can be used with ARC [10] and gLite [11] based DCI interface module (majority of the HP-SEE infrastructure is using gLite and ARC, as EGIs grid infrastructure does it in a similar way). These modules contain the middleware specific commands to copy and run the vulnerability scanner and to gather the results of the investigations. The prepared OpenVAS package contains a modified and precompiled OpenVAS vulnerability scanner. The AVAT stores the scan results and makes them available for the administrators of the resources.

3.2 Introducing gLite Based Resources

The following sub-section introduces a brief overview about some components of the gLite middleware.

On the User interface (UI), a user can gain access (after the successful authentication) to use the DCI resources. The UI provides command line interface to the end user to utilize all the basic Grid operations. Computing Element (CE) is some set of computing resources localized at a site (i.e. a cluster, a computing farm). Worker Nodes (WN) are execute the tasks. A Storage Element (SE) provides uniform access to data storage resources. The Information Service (IS) provides information about the grid resources and their status. Berkeley Database Information Index (BDII) is used to store and publish monitoring and accounting data from the grid sites. Virtual Organization Membership Service (VOMS) is tightly coupled system for managing authorization data within multi-institutional collaborations.

So far we know how a gLite based middleware builds up, we can describes how the AVAT checks the targeted grid resources. We need to note here, that AVAT is a generic solution and capable to assess both gLite and ARC based DCIs. Investigation of ARC based resources is very similar, only the component names and the used commands are different. The AVAT uses the following steps in order to check gLite base resources: The control module sets up the environment (e.g name of the investigated VO and the entry point of the information system). Then, the control module collects the available resources from the information system and it sends test jobs into the resources individually via the DCI module.

Figure 4 shows the object interactions when AVAT investigates a gLite based infrastructure. The DCI module uses a gLite UI to submit the test jobs to the WMS. The WMS forward the job to the corresponding CE. The CE schedules

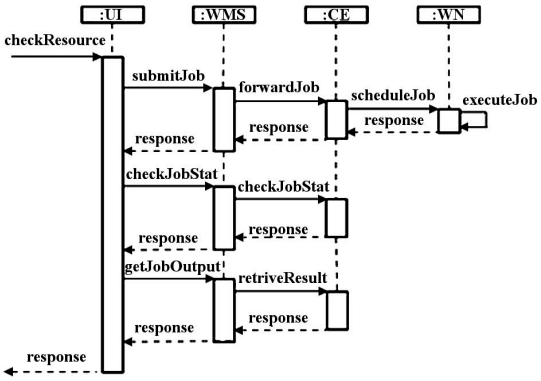

Fig. 4. Investigating a gLite resource by AVAT

the task to one of its WNs. The WN execute the test job and the CE sends back the results to the WMS. The AVAT checks the status of the job periodically and if it is ready than the results will be gathered by the UI. The test job (that runs on WNs) contains the following sub- tasks: (1) Download the precompiled OpenVAS server, client and libraries. (2) Set up the environment (e.g set up the path of the binaries and the libraries). (3) Update the collection of the NVTs. (4) Start the OpenVAS server. (5) Start the OpenVAS client, connect to the server and scan the local machine.

4 Investigating the Results of the Test Scans

To prove AVATs capabilities and test its usability parameters, both gLite and ARC based DCIs have been investigated during autumn 2012.

4.1 SEE-GRID VO

The AVAT queries the available resources from the info-system of the SEE-GRID gLite based grid VO and sends test jobs into the corresponding sites. These test jobs characteristically run from 20 to 30 minutes. There is a parameter in the system that determines the delay of collecting the results. This parameter is one hour by default, however it can be freely adjusted to the underlying DCI.

Figure 5 summarizes the results of the investigation. Most of the sites have vulnerability issues. There are two sites having more than 10 security holes and the average is 3.5/site. Only 4 from the 16 investigated sites were up-to-date or secured with hardened kernel.

4.2 HP-SEE Supercomputing Infrastructure

The HP-SEE supercomputing infrastructure does not have any centralized authorization entry point (like the VOMS in the gLite based grids). Therefore, we

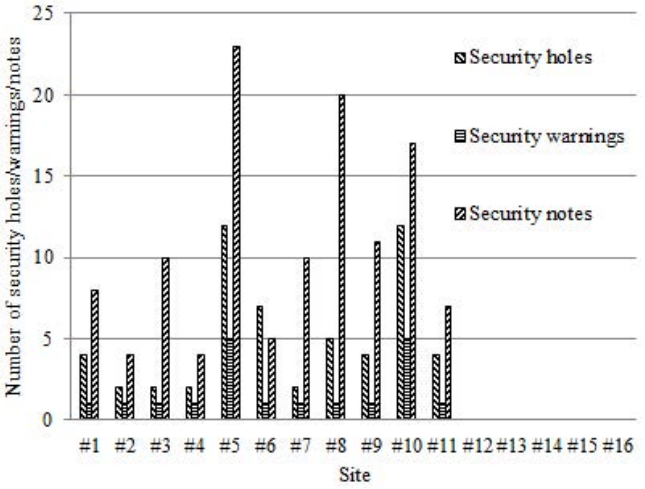

Fig. 5. Vulnerability status of the SEE-GRID VO

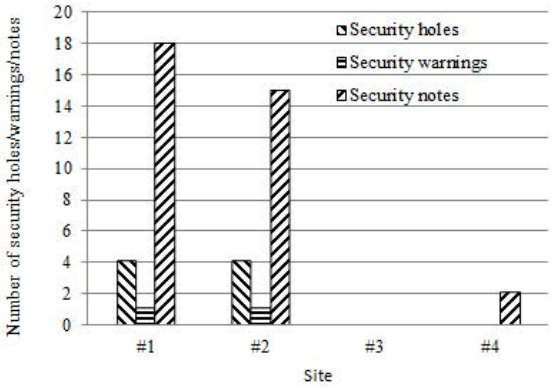

Fig. 6. Vulnerability status of the investigated HPC centers

registered into four sites of HP-SEE. The AVAT test jobs had been sent to the HP centers separately and the results were collected about 5 minutes later.

Figure 6 summarizes the results of the investigation. The integrated OpenVAS service reported four security holes and numerous (15 and 18) security warnings for site 1 and site 2. There were not detected any vulnerability on the third and fourth sites.

5 Conclusion and Future Work

In this paper, we discussed the major vulnerability sources in distributed computing infrastructures and we presented some of the recently used monitoring

solutions and vulnerability frameworks in DCIs (highlighted the SEE-GRID-SCI and HP-SEE DCIs). We pointed out that the centralized vulnerability assessment solutions for distributed systems (PAKITI and GSSVA) only do software version checks. However, the feature-rich frameworks (e.g OpenVAS) could not work easily in different administrations domains. Then, we introduced our proposed software solution; the Advanced Vulnerability Assessment Tool (AVAT) that combines the benefits of the GSSVA and OpenVAS frameworks. Moreover, we discussed the results of using AVAT on two different type of DCI: grid SEE-GRID-SCI and supercomputing infrastructure HP-SEE, based on two different middleware (gLites and ARC).

In the future, we plan to extend the AVAT with more DCI interface modules: we would like to add EC2 interface in order to support cloud vulnerability assessment too. The proposed software can easily (re)used for other DCIs and authors are jointly working with various DCI communities to open up the vulnerability service for other projects and infrastructure providers.

Acknowledgment. The research leading to these results has received funding from the European Social Fund and the Hungarian TMOP-4.2.1.B-11/2/KMR-2011-0001 Kritikus infrastruktura vedelmi kutatasok project and as well as from European Union Seventh Framework Programme (FP7/2008-2013) under grant agreement no RI-261499 (HP-SEE) High-Performance Computing Infrastructure for South East Europes Research Communities project.

References

1. The SEE-GRID-SCI website (2012), `http://www.see-grid-sci.eu/`
2. Kozlovszky, M., Windisch, G., Balasko, A.: Short fragment sequence alignment on the HP-SEE infrastructure. In: MIPRO, 2012 Proceedings of the 35th International Convention, May 21-25, pp. 442–445 (2012)
3. Martin, R.A.: Managing Vulnerabilities in Networked Systems. IEEE Computer Society Computer Magazine, 32–38 (2001), `http://cve.mitre.org/`
4. Mell, P., Scarfone, K., Romanosky, S.: A complete guide to the common vulnerability scoring system, version 2.0. Forum of Incident Response and Security Teams (June 2007)
5. Balaz, A., Prnjat, O., Vudragovic, D., Slavniv, V., Liabotis, I., Atanassov, E., Jakimovski, B., Savic, M.: Development of Grid e-Infrastructure in South-Eastern Europe. J. Grid. Comput. (9), 135–154 (2011)
6. The Pakiti website (2012), `http://pakiti.sourceforge.net/`
7. Acs, S., Kozlovszky, M., Balaton, Z.: Automation of security analysis for service grid systems. In: PARENG 2009, The First International Conference on Parallel, Distributed and Grid Computing for Engineering, Pcs, Hungary (2009)
8. The Qualys website (2012), `http://www.qualys.com/`
9. The OpenVAS website (2012), `http://www.openvas.org`
10. The ARC website (2012), `http://www.nordugrid.org/arc/`
11. The gLite website (2012), `http://glite.web.cern.ch/glite/`