ISSN 0146-4116, Automatic Control and Computer Sciences, 2016, Vol. 50, No. 8, pp. 669–672. © Allerton Press, Inc., 2016.
Original Russian Text © A.S. Konoplev, M.O. Kalinin, 2016, published in Problemy Informatsionnoi Bezopasnosti, Komp'yuternye Sistemy.

Tasks of Providing Information Security in Distributed Computing Networks

A. S. Konoplev* and M. O. Kalinin**

Peter the Great St. Petersburg Polytechnic University, St. Petersburg, Russia

**e-mail: artem.konoplev@ibks.ftk.spbstu.ru*

***e-mail: maxim.kalinin@ibks.ftk.spbstu.ru*

Received June 17, 2016

Abstract—The issue of providing information security for data and computing resources in grid networks is reviewed. Specific features of architecture of distributed computing networks based on grid platforms are analyzed. Security threats specific for grid systems are typified. The available measures ensuring security for grid systems are considered, and their drawbacks are indicated. The set of applied issues associated with ensuring grid protection from unauthorized access is defined.

Keywords: distributed computing networks, grid, information security, security threats, security policy

DOI: 10.3103/S0146411616080101

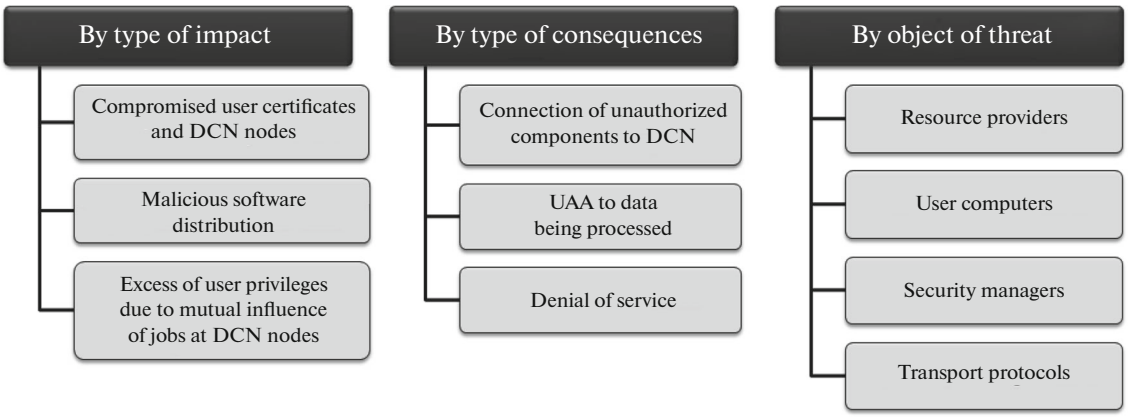
Nowadays, distributed computing networks (DCN) based on computing grid platforms are widely applied for solving high performance and resource-intensive computational tasks in science and commerce. Since data being processed in grid are highly valuable, special attention is paid to information security aspects in such systems.

The grid system is a heterogeneous framework, which includes personal computers, workstations, servers, mainframes, and other computing systems combined into one virtual supercomputer. The fundamental principle of shared access to grid resources is implemented with virtual organizations (VO), the set of grid system users solving the same applied task. Due to functional peculiarities of grid systems, they are characterized by decentralization, heterogeneity, and high dynamically changeable states, which complicate the issue of ensuring their security, including protection of computing and information resources.

Analysis of architecture and protection mechanisms of grid systems made it possible to typify security threats specific for DCN based on the following criteria (figure):

—type of consequences;

—objective of threat;



Security threat classification for DCN.

—type of impact on DCN.

DoS attacks are performed as a result of network attacks on basic DCN services, disconnection of authorized components from the DCN, DCN overload, which blocks the work of users and services.

Malicious software distribution attacks are topical for DCN because it provides a legal channel for performing distributed calculations, i.e., they are a perfect environment for malicious software distribution.

The threat of unauthorized access (UAA) to user's information and computing resources may come from both authorized DCN users and components and outside intruders. UAA types specific for any DCN are as follows:

—connection of an unauthorized user or component to the system (the component is unauthorized when its certificate is not issued by a trusted certification authority);

—attempt to access to the data of DCN users by processes and users of the local host environment;

—attempt to excess the privileges by a DCN user, application or service.

Attacks on web services include traditional attacks, such as XSS, session hijacking, password theft, and social engineering methods.

Depending on what nodes are threatened, the following consequences of successful threat implementation by an intruder may be defined:

—user certificate hijacking making it possible to bypass the authentication procedure when accessing DCN resources;

—UAA to user data bypassing security policy (SP) requirements;

—Denial of access to system resources for authorized participants of computing process (for example, for local users of resource providers);

—Denial of service, including failure of nodes responsible for coordination of DCN services and components, distribution of user requests for the use of DCN resources, authorization servers, and other security managers;

—unauthorized use of system resources (for example, by processes of local user accounts of resource providers).

To protect DCN nodes from potentially malicious software transmitted into the DCN with user jobs (request for accessing computing resources implies execution of a certain user application at the DCN node (nodes)), several approaches are presently used:

—the use of Proof-Carrying-Code;

—sandbox-type isolated software frameworks;

—virtualization (complete emulation, paravirtualization, hardware virtualization).

Proof-Carrying-Code is a software-based mechanism, which allows the host system to verify application properties using the formal proof integrated into the executable code [1]. The host system may quickly check the validity of proof and compare the check results with its own SP requirements to define whether the given application is secure.

However, the use of this technology in the DCN has two major constraints. The first constraint is that due to high heterogeneity of DCN software vendors (DCN users) would have to adapt to the SP requirements of each resource provider individually, which would dramatically reduce the scalability potential of such systems and increase downtime of the equipment. Another constraint is due to the fact that the majority of advanced DCN implementations are constructed based on freely distributed software supplemented with a number of third-party libraries and modules, whose initial text is unavailable to a common DCN user. Therefore, it is impossible in this case to perform modifications of this software required by the use of Proof-Carrying-Code technology.

The sandbox framework operates using the principle of limiting the activity of potentially malicious applications in a way that they are unable to damage the user's system. Activity limitation is reached by executing unauthorized software in the restricted environment, where the application has no permissions to access sensitive system files, registry keys, and other important information.

Advantages of systems built using the sandbox are as follows:

—low system resources consumption;

—moderate hardware requirements;

—small number of requests to the computer user.

Disadvantages of sandbox-based systems:

—the user has to possess knowledge of principles of OS operation;

—impossibility of countering active computer infection.

An evolution of sandbox software framework is presented by virtualization, the technique of representation of a set of computing resources or their logical combination, which in a certain way overcomes the original capabilities. Typically, virtualized resources include computational facilities and data storage. In other words, virtualization is an isolation of computing processes and resources from each other.

Symmetric multiprocessor computer architectures with more than one CPU may be observed as as a sample of virtualized system. OS are typically configured in a way that several CPUs are represented as a single processing unit. That is why software applications may be written for one logical (virtual) computing unit, which is significantly easier, than working with a large number of various processor configurations.

The use of virtualization for DCN is especially peculiar, when access to grid is provided via web dashboard (it similar to cloud computing systems).

Since computing platform for running user jobs in DCN is represented by connected personal computers, rather than isolated servers (as in cloud computing systems), isolation of the data of DCN users from the impacts of the host environment becomes an important issue for ensuring data security from UAA attacks. To achieve this, a trusted platform is used in DCN nodes [2], which makes it possible to protect against the processes from privileged local users at DCN nodes.

The data of DCN users at the moment of job execution at DCN nodes are stored at the dedicated protected storage in an encrypted mode. The access key to the storage is only known to the user who initiated the request for job execution. Distribution of cryptographic keys is implemented via trusted software DCN components.

User access to computing resources of the DCN actually implies the necessity of executing applications of some users in software framework (in operating system) of other users. Thus, the task of preserving availability of resources for all participants of the computing process becomes topical.

The issue of process starvation at DCN nodes may occur, if user jobs occupy processor or RAM resources of the system to an extent that local users of such system turn out to be unable to normally interact with it. This problem often arises due to errors in applications, which cause deadlocks and lockups in the system.

To avoid process starvation in DCN, two approaches are used:

—resource reservation;

—priority reduction.

The approach based on resource reservation migrated to DCN from cloud computing systems, where each virtual machine (computing unit of a cloud) is assigned with a strictly limited amount of the computer's computing capabilities, which cannot be exceeded. Similarly, maximum threshold values of system resources used, which cannot be exceeded by user jobs, are set in the DCN.

Another approach to solving this problem implies assigning lower priority to utility processes and processes executed on behalf of the outside DCN user compared to processes executed on behalf of local users at DCN nodes.

To ensure security of DCN from network threats, such systems are integrated with various information security tools (IST): network firewalls (instance.g., Adaptive Firewall for the Grid [3]), intrusion detection systems (IDS) (e.g., Snort for Grid [4]), and alerting systems. In this case, hardware and software components, i.e., security managers, which have IDS, network firewalls, and antiviral tools installed, are integrated into the DCN. Security managers are connected by dedicated communication channels, through which alerts are broadcasted in case of intrusion detection. Upon receiving the alert, each host duplicates it for all resource providers undergoing its control. As a result, all DCN nodes isolate the affected node, which becomes a source of a threat, thereby excluding the possibility of attack expansion.

Authorization mechanisms used in the DCN are divided into two types. The first type of authorization allows resource providers to set the account, under which any computations are performed. However, this mechanism does not involve VO [5], which forms the active basis for theDCN, as it includes just one subject in such systems, and this makes it impossible for the provider to restrict access for a certain user or extend the access rights of another. Thus, many DCNs use authorization, where each user who initiates processing of a certain job, is assigned with a certain local account.

The discussed approaches to ensuring operational security of the DCN are expensive and hard to implement, and hence rarely used in contemporary DCN. In addition, there are no protection tools making it possible to efficiently counteract attempted excess of privileges by DCN users and components, as well as classical attacks on the client typical for information systems with web services. Thus, the analysis

allows us to conclude that at present DCN security functions are only partially implemented and do not provide adequate protection for the DCN from UAA.

User job distribution model in DCN [6] is defined by the tuple $\sum = (\nu_0, J, RP, RT, SP, R, \Psi)$, where ν_0 is the initial DCN state from the set of system states $V = \{n\}$; J is the set of active jobs, each of which is mapped to the DCN user, who initiated its execution; RP is the set of resource providers; RT is the set of types of jobs processed at resource providers; SP are security policy rules; R is current access relationships at resource providers; $\Psi: J \times RP \times RT \times SP \times R \to V$ is the transition function of the DCN from one state to another.

An important advantage of the model based on mathematical technique of Petri nets is simplicity of construction of hierarchical structures in the process of modeling parallel processes in systems. In the DCN this process is represented by user job distribution between system nodes. Here, parameters setting the definition domain of function Ψ are variable. Formalization of a transition function, as well as presence or absence of the indicated parameters and range of the function, allow us to define the set of applied tasks for ensuring DCN data protection from UAA:

1. Control of user access to data being processed. Sets J, RT, SP, R are known; set RP is to be defined. The presence of the indicated parameters allows us to define sets of DCN nodes for execution of user jobs considering SP requirements. Thus, the necessary condition of secure user job distribution in DCN is fulfilled, under which each system transition to a new state does not conflict with SP requirements.

2. SP verification. Sets J, RT, RP, SP are known; set R is to be defined. The presence of the indicated parameters allows us to identify access relationships leading to deviations from SP requirements.

The search of nodes suitable for user job execution is performed by a dedicated DCN service operating at resource providers. Therefore, to solve the task of secure user job distribution in DCN, the transition function is to be integrated into the suggested service. Thus, suitable nodes will be selected considering not only their availability and DCN resource type possessed by the node, but also SP requirements, which either grant or deny the use of resources at the given DCN node.

Overall, solving the stated issue, implementing the indicated software module and its integration into resource providers will make it possible to automate the security analysis procedure and make it impartial, thereby ensuring high reliability and security of DCN.

ACKNOWLEDGMENTS

The work is supported by the Ministry of Education and Science of the Russian Federation within the framework of the state assignment to higher-education institutions subordinate to the Ministry of Education and Science in the field of scientific research (State assignment no. 2.1778.2014/K as of July 15, 2014; Application code 1778).

REFERENCES

1. Necula, G., Proof-carrying code, *Conference Record of POPL'97: The 24th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages,* 1997, 106–119.
2. Lohr, H., Ramasamy, H.V., Sadeghi, A., Schulz, S., Schunter, M., and Stuble, C., *Enhancing Grid Security Using Trusted Virtualization,* Springer, 2007.
3. Yao, D., Adaptive firewalls for grid computing, in *Informatics and Mathematical Modelling,* Technical University of Denmark, 2005.
4. Kumar, M., Hanumanthappa, M., and Kumar, T.V.S., Intrusion detection system for grid computing using SNORT, *IEEE 2012 International Conference on Computing, Communication and Applications,* 2012, pp. 1–6.
5. Sciaba, A., Burke, S., Campana, S., Lanciotti, E., Litmaath, M., Lorenzo, P.M., Miccio, V., Nater, C., and Santinelli, R., *Glite 3.2 User Guide,* CERN, 2011.
6. Konoplev, A.S., Kalinin, M.O., Zegzhda, D.P., and Moskvin, D.A., Access control model for Grid calculations, *Proceedings of the 2014 International Conference on Communications and Computers (CC'14),* St. Petersburg, 2014, pp. 54–57.

Translated by A. Amitin