Therefore, the control system front-end has been designed for cinemas of the Republic of Belarus. This system will allow to make the process of poster management easier and faster. The graphical user interface is designed. This interface is clear to any common user without any learning of working with a system. There is a very detailed help page. It contains step descriptions for all user cases in the system. In case of wrong actions a user will be notified with an error message. In case of successful action completion a user will receive a message that the operation is successfully completed.

REFERENCES

1. Макфарланд, Д. Большая книга CSS3 / Д. Макфарландю – 3-е изд. – СПб. : Питер, 2014 – 608 с.
2. Бенедетти, Р. Изучаем работу с JQuery / Р. Бенедетти, Р. Крэнли. – СПб. : Питер, 2012, – 512 с.
3. Спецификация Google – Material design [Электронный ресурс] / Google. Material design. – Режим доступа: http://www.google.com/design/spec/. – Дата доступа: 10.01.2016.
4. Плюсы и минусы CMS [Электронный ресурс] / Mywebblog.ru Блог Вебмастера – Режим доступа: http://mywebblog.ru/sozdanie/plyusy-i-minusy-cms.html. – Дата доступа: 10.01.2016.

UDC 004.415.25

THE SYSTEM PERFORMANCE ANALYSIS TO COMBAT SOFTWARE ABNORMAL ACTIVITY

ALIAKSEI RAMASHKA, KANSTANTSIN RAKHANAU
Polotsk State University, Belarus

This article defines the functional structure of the system to combat the abnormal activity of software, proving the performance analysis.

Recently, with the growth of information technology in economy and industry, and the appearance of information, presenting a certain value, whether it is a certain production technology or a customer's data, the problem of information protection has become acute.

Thus, information should be protected from unauthorized access and leakage. In terms of information technology, the computer equipment, primarily computer workstation, company employees pose a particular threat. [3]. One of the ways to gain unauthorized access to information is the introduction of the software with not declared possibilities that generate anomalous activity software.

The Functional Structure of the System

The automated system should be a software system to detect abnormal activity of software, as well as eliminate the possibility of changing information and unauthorized access to it.

Abnormal activity detection system must meet the following requirements:

– To analyze the running of the test process (application);
– The ability to analyze the signature of the executable file of the test process;
– The ability to analyze network connections opened by the process;
– The ability to analyze network traffic generated by the application;
– Implementation of the decision-making by the operator or automatically;
– To meet the requirements of fault tolerance (the failure of the system components, malicious attacks on the system resources);
– To save high performance during peak loads;
– Load balancing between multiple modules threat detection;
– Low load power and hardware resources of the computer;
– Support for cloud computing.

Based on the above-cited requirements, the system comprises the following components:

– The subsystem of information on network activity of the application;
– The subsystem intercept network packets;
– Subsystem integrity monitoring system;
– Subsystem threat detection;
– Subsystem automatic response;
– Subsystem load balancing;
– Remote Control Subsystem;
– Administration subsystem;
– The authorization subsystem.

The subsystem of information on network activity is designed for permanent monitoring of all the processes on the computer, as well as to retrieve data about network connections, initiated by these processes.

The subsystem is designed to intercept network packets to collect data on all the network packets generated or received by the processes on the computer.

The subsystem integrity monitoring system is designed to monitor the state of the system. In case of accidental or unauthorized shutdown of one of the components of the system, this subsystem quickly restores the modified file.

The subsystem threat detection is the core of the whole system. It is responsible for the analysis of the events collected by the subsystem of network activity and the application subsystem intercept network packets.

The subsystem of automatic response is responsible for the measures, taken on the basis of threat detection subsystem analysis.

The subsystem load is responsible for allocating the tasks of analysis and decision-making evenly among all the computers on the network in a situation where the server does not have enough hardware resources or when it fails. When the load is much greater than that one which the basic analyzer can sustain, this subsystem automatically distributes the load for slow client machines. The balancing runs in the following way: if the server CPU usage is greater than 80%, the transfer of control over the management services on the client machines is commanded. The Management Service calculates the percentage of CPU usage and the amount of free memory and writes this value down into the database. The Empirical formula has been obtained to calculate the number of users that are currently possible to analyze:

$$N = RT * \frac{PU_e}{PU_c} * \frac{RA_e}{RA_c} \quad (1)$$

PU_e, RA_e - experimentally obtained values of the CPU usage percentage and the amount of free memory, and PUc, RAc - the current values of the same parameters, RT - number of threads in the CPU.

Remote control subsystem is responsible for the automatic execution of commands or applications to block network traffic from the subsystem automatic response.

Administration subsystem is designed to enable the functioning of the system in semi-automatic mode, which makes the systems administrator or security officer block applications or network traffic depending on the event that occurred.

Subsystems threat detection and decision is made on the remote host. It is implemented on the concept of cloud computing, an example of which is presented in [4]. It thereby greatly reduces the load and the consumption of hardware resources, as the result of the analysis.

The presented structure describes the basic feature set, and it can be expanded to over-dependence on specialization.

Performance Test

To check the analysis accuracy and the response rate, test applications have been created, the core functionality of which is to send a request to a remote server. The ten test runs did not reveal a single case when the test application failed to transmit data.

The overall software performance has been checked on the systems with 8GB of RAM and a quad-core processor. The operating system was reset, to minimize the impact of the other applications, running in the operating system, on the experiment. To emulate the processing stages, multiple clients run threads of execution were performed. They processed the data from 100 clients. The results of the experiment are shown in Figs. 1, 2, 3.

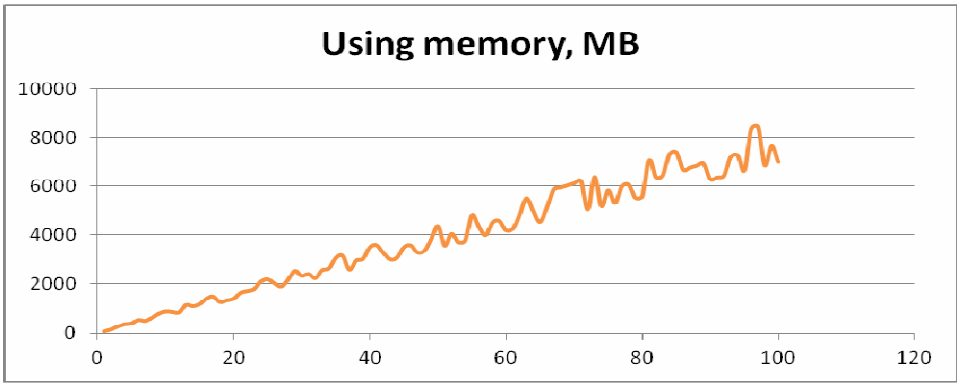

Fig. 1. Dependence of Memory Usage on the Number of Clients Processed

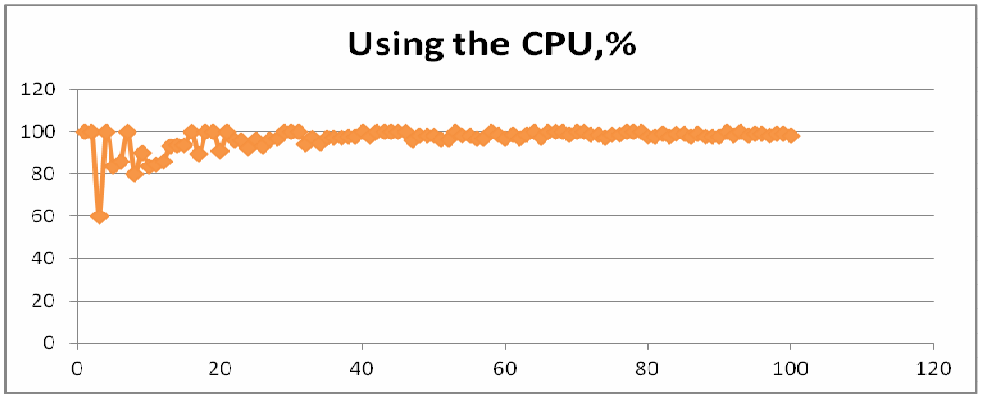

Fig. 2. Dependence of CPU Usage on the Number of Clients Processed

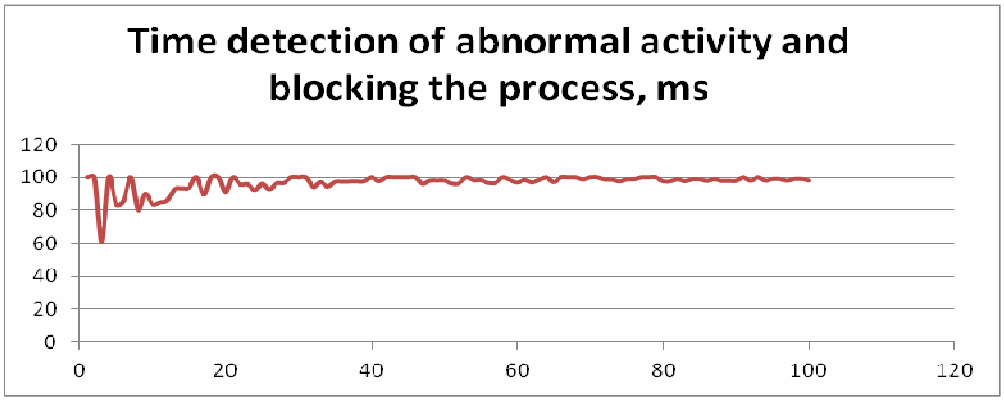

Fig. 3. Dependence of the Software Operation Time on the Number of Clients Processed

Analysis of the data:

– the amount of RAM used is directly proportional to the number of clients served. The differences in Fig. 1 illustrate the smaller sample of data for analysis, which, respectively, requires less memory for storage;

– when there is a large number of clients at the same time, CPU utilization approaches 100%. For this reason, to process large amounts of data, it is advisable to use a dedicated server with lots of RAM and a powerful multi-core processor. If this is not possible, we recommend using the distributed processing;

– the time spent on data processing and decision making is directly proportional to the workload of the processor and does not exceed 100 ms.

The functional structure of the system to combat abnormal activity of software that has high reliability and performance is shown through the use of distributed computing and cloud technologies. The main feature of this specific-structure is modular, due to which, at the lowest cost, it is possible to integrate additional modules analysis and response. High performance is confirmed by carrying out load testing.

REFERENCES

1. Группа компаний InfoWatch [Электронный ресурс] / InfoWatch, 2015. – Режим доступа: https://www.infowatch.ru/sites/default/files/files/products/appercut/Appercut_Company_Brochure_Rus.pdf. – Дата доступа: 16.09.2015.
2. Seagate Technology LLC [Электронный ресурс] / Архитектуры облачных систем обработки и хранения данных. – 2015. – Режим доступа: http://www.seagate.com/ru/ru/tech-insights/cloud-compute-and-cloud-storage-architecture-master-ti/. – Дата доступа: 16.09.2015.
3. Министерство внутренних дел Республики Беларусь [Электронный ресурс] / Первое российское исследование скрытых угроз. – 2011. – Режим доступа: http://mvd.gov.by/main.aspx?guid=55533.– Дата доступа: 16.09.2015.
4. Орлов, С. Облачные сервисы: безопасность и надежность // С. Орлов // Журнал сетевых решений. – 2012. – № 12. – 10 с.