Technique of Data Visualization: Example of Network Topology Display for Security Monitoring

Maxim Kolomeec, Andrey Chechulin, Anton Pronoza, and Igor Kotenko*
Laboratory of Computer Security Problems
St. Petersburg Institute for Informatics and Automation (SPIIRAS)
39, 14 Liniya, St. Petersburg, Russia
{kolomeec, chechulin, pronoza, ivkote}@comsec.spb.ru

Abstract

The paper presents the results of research devoted to the development of an unified flexible visualization system for security monitoring of computer networks used in the SIEM systems. The developed models and technique of visualization are used for selection of methods of data collection, normalization, preprocessing and representation. The individual components of the proposed visualization system are described using set-theoretic models. To analyze the operability of the developed models and methodologies a software prototype of the visualization system is developed and experiments are conducted.

Keywords: visualization techniques, formal models, visualization of topology of a computer network, security monitoring, computer network, SIEM, cyber security.

1 Introduction

Contemporary information systems are characterized by a large amount of processed data, so visualization tools became an important instrument for solving problems of data analysis. Methods of visual data analysis helps to efficiently explore big data and extract new knowledge from arrays of heterogeneous or non-formalized data. The basic idea of visual analytics is to combine the peculiarities of human visual perception of the information and capabilities of electronic data processing, which results in the creation of highly interactive software allowing the user to dive into data to better understand the results provided by the security information and event management (SIEM) system (or security monitoring system) in information and telecommunication systems and to make efficient decisions.

However, the visual analysis of data requires to consider many different issues, dealing, primarily, with the specification of problem solved with their help, the determination of source data and choice of efficient schemes of graphic information encoding, which defines how the data is displayed using graphical attributes such as color, size, location in space. There are many different ways to visualize data, and to successfully determine the most appropriate model of the graphical representation we need to take into consideration specific application area for which you are developing the visual analytics system, as well as the human visual perception of the information.

The visualization process is the display of the user information in the form of an image on the screen. However, the display of information is only the final stage of the visualization process, preceded by the information processing and conversion to a form suitable for display. Thus, it seems necessary from a theoretical point of view to consider the data structures used in visualization of computing infrastructure, as well as the method of their organization to uniformly get different representations of a computer network.

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 7:1 (Mar. 2016), pp. 58-78
*Corresponding author: Tel: +7(812) 328-71-81, Web: http://www.comsec.spb.ru/

The aim of the presented work is development of generalized technique and formal mathematical models of representation of heterogeneous data and demonstration of how these technique and models can be used to visualize computer networks for security monitoring using tree and graph structures.

As input data for this paper we considered the information entering the security monitoring system that is used to fill attributes of models of information objects. Due to the heterogeneity of this information, there is a need for its formalization and unification in order to display the security status of a computer network.

The novelty of the technique and models of data visualization proposed in the paper is to use the proposed flexible unified approach to the construction and use of security monitoring visualization systems. This approach is based on the one hand on formal mathematical models, allowing to combine individual components of the system, and on the other hand it takes into account the characteristics of the cognitive apparatus of the operator, thereby increasing the efficiency of visual analysis of the displayed information. Thus, the proposed models and technique allow to use a single model for data visualization, which, depending on the displayed data, the operator's requirements and the features of the cognitive apparatus, is capable of displaying the input data.

The paper describes the architecture of visualization system as well as developed mathematical models and technique of visualization. The paper also gives an example of using the developed technique for visualizing the topology of a computer network. This example shows how the technique can be applied to a particular task at various stages of building of the visualization model – from data collection to their display. Elements of the developed software prototype of the visualization system are provided.

The paper contains the following sections. Section 2 presents the analysis of the existing literature in the domain of information visualization concerning safety. In section 3 the general architecture of the visualization system and formal mathematical models describing its elements are presented. Section 4 describes the technique of selection of approaches and algorithms to the construction of the components of the visualization system. Section 5 describes an example of using the methodology for forming components of the visualization system. Section 6 gives the general analysis of the obtained results. Section 7 summarizes the results of the work and analyzes the possible directions for further research.

2 Analysis of relevant works

Visualization of information related to the computer network security is a relatively new area of visual analytics. However the constantly increasing amounts of data, speed of their processing and variety of data types in information infrastructures contributed to the development of new models, algorithms, and visualization tools.

In general, the process of visual analysis can be described by a sequence of actions "Preliminary analysis $\rightarrow$ General representation $\rightarrow$ Scaling, filtering $\rightarrow$ Further analysis $\rightarrow$ Details on demand", which emphasizes the necessity of combined application of methods of graphical representation of data and their automated processing.

To improve the efficiency of the graphical representation of data we use different methods of user interaction with them [1]: filtering; isolation and linking; reconfiguration of data models allowing to explore data from different viewpoints; dynamic coding, enabling different ways to represent the properties of an object, for example, using the graphic attributes; zooming, which allows to control the detailization of the image; interactive distortion of the image, allowing to obtain detailed images of selected portions of the source image.

This approach is in good agreement with the model of the design of graphic systems "data $\rightarrow$ display $\rightarrow$ view $\leftrightarrow$ control" [2], according to which the data and model of visualization, as well as models of visualization and graphical representations should be separated, to enable support for multiple models of

visualization and graphical representations for the same data set.

However the papers, considering the visualization systems, are as a rule devoted to the results of specific developments – that are special cases, which do not consider the process of building visualization systems in general. For example, paper [3] discusses the system of visualization of network traffic based on ports and hosts, allowing to identify network attacks and scanning. A similar system is presented in [4], where the analysis of the network traffic based on the matrices is done that allows to visualize up to 10,000 hosts. Existing works typically offer specific architectures of visualization systems and so do not consider any common generic methodology for constructing the visualization system or architecture that is applicable to other visualization systems.

The classic representation of a model of a computer network is a graph in which nodes are understood to be the hosts of the network, and arcs are used to denote connections between hosts. Papers [5, 6] describe the approach in which in order to represent the nodes of the graph it is suggested to show the metrics of security of the respective hosts. This approach is extended in [1], so the user can see both current and previous values of the security metrics.

There are many papers devoted to the individual components of the visualization systems and their architectures, for example, for data collection, data analysis and presentation. In the area of data collection on the status of the network the classic approach is the use of active and passive scanners. One of the most popular active scanners for the analysis of networks is Nmap [7], and preferred passive scanner is Wireshark [8]. For example, paper [9] describes the software tool which implements the visualization of computer networks on the basis of analysis of transferred traffic, without consideration of other sources of information. In [10] a technique of visualization of networks is proposed that is based both on active information gathering using service protocols (ICMP, RIP, etc.) and on passive one, using extraction of information from a database of routers and other devices. This approach allows to visualize the network topology by means of a standard network operating system, however, it does not provide information about the state of its security.

We should specially note many works devoted to various ways and techniques of displaying data. In previous work of the authors [11] there were collected and described the main models and graphical ways of displaying data, as well as techniques of their estimation and the efficient application depending on the usage scenario.

3 The visualization system

The process of visualization, from data collection and to their display, can be represented in the form of typical model of the structure of the visualization process – the visualization pipeline (Figure 1) [11].

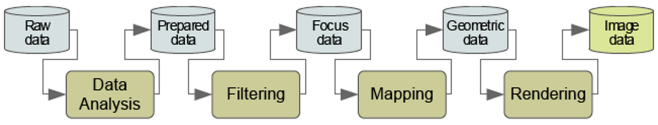


Figure 1: Visualization pipeline

Despite the different representations of this model, as well as multiple modifications, the visualization pipeline gives an overview of the sequence of data processing for visualization, and, as a consequence, of the components that are necessary for the architecture. In accordance with the processes represented in this model, you can define a minimal set of components of the overall architecture of visualization system: data collection component, data analysis component, markup component, component of data display. It should be noted that the collecting component may not collect the data by itself but

instead import them from different systems. Component for analysis may perform normalization, filtering, aggregation, correlation or use external systems. Component for markup translates data into graphic patterns. The display component allows to display data on the screen of the user's computer. Thus, when constructing the overall architecture of the visualization system we should consider the components and processes of the visualization pipeline model.

In Figure 2 the proposed scheme of functioning of the subsystem of visualization is presented. It is based on general approach to the analysis of multidimensional data proposed in [12].

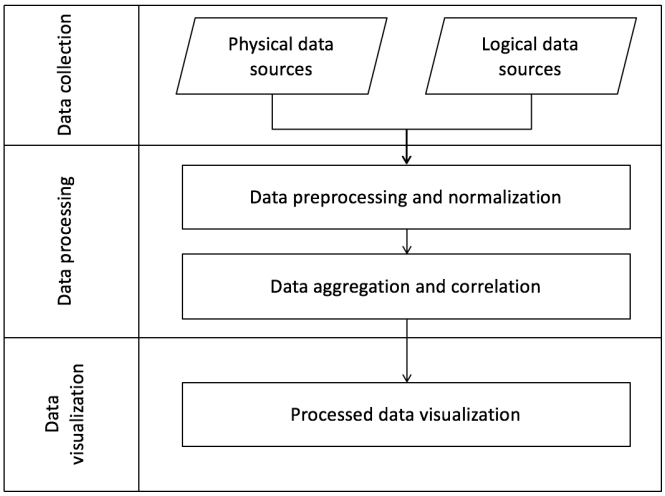


Figure 2: The general scheme of the visualization of multidimensional data

This scheme corresponds with the four steps that are required for data visualization: "raw data collection $\rightarrow$ raw data preprocessing $\rightarrow$ new features forming $\leftrightarrow$ data visualization and management". On the first step, data is collected from different sources (for computer network these sources are active and passive tools for data collection (Nmap, Nessus, Wireshark, etc) and operator's knowledge). On the second step, the data is normalized, i.e. algorithm converts data and it's characteristics to a unify format. On the third step, the formation of new structures and the facts on the basis of available data is performed (for example, on the basis of different types of connections the general matrix of relationships formed for an analyzed computer network). On the fourth step the best visualization model is selected and the very visualization is performed.

3.1 General architecture of the visualization system

In order to build a visualization system we need to develop a flexible architecture that will include all elements of the build process for visualization: from data collection to their display. To do this, one must: (1) define the principles of source data collection, where the flexibility of the architecture should give the ability to attach new data sources to the visualization system; (2) determine the sequence of the processes of data preprocessing, as the operations that are necessary to produce data for visualization are often not commutative; (3) identify possible data loss at visualization to allow visualization of additional information upon request.

Thus, for visualization systems the modular architecture is better suited where each module is independent and can be supplemented in such a way that it will not affect the operability of the other modules. In Figure 3 visualization system is presented which consists of five main modules:

- data acquisition module;

- data normalization module;
- aggregation and correlation module;
- data storage module;
- visualization module.

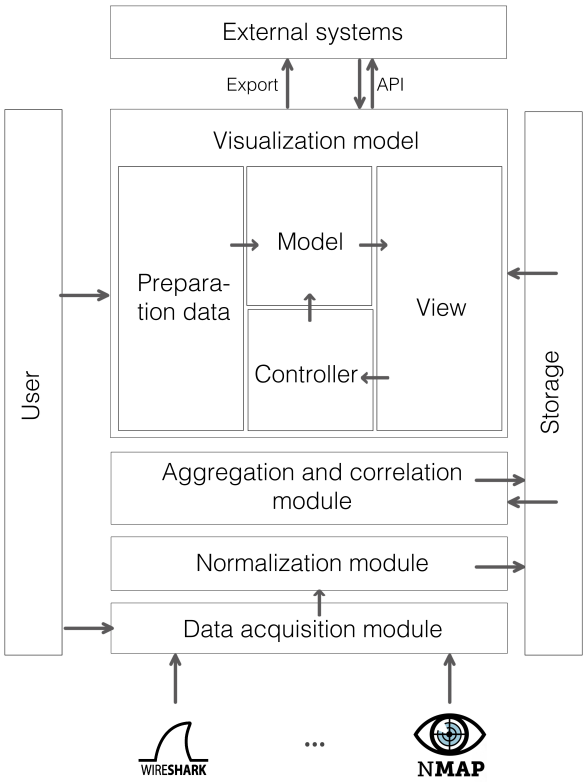


Figure 3: The architecture of the visualization system

Data acquisition module includes interfaces for various systems that may be sources of data. Such systems can be both active and passive computer networks data collection systems, such as Nmap, Wireshark, etc. and other software components and systems. Because each source provides data in its unique form, the data acquisition module includes parsers for each source, which allow to convert data from various sources into the internal structures of the system. It is also supposed that the user can add to the system its own custom parsers, if he wants to collect data from specific systems.

Once the data were received and converted into the representation understandable by the system, it is necessary to normalize the data, that is to convert the set of data structures that are specific for each of the sources to the unified structure and types of values. For the implementation of this step data normalization module is responsible. It should be noted that in the case of connection to the module data collection of specific source it is necessary not only to load a new parser to the collection module, but also to add the appropriate component in the normalization module.

After data transformation to the uniform structure their analysis may be performed. For this step the module of aggregation and correlation of data is responsible. As in various sources some data may be repeated, it is necessary to initially produce aggregation. After that, to identify patterns and make

conclusions, the correlation should be performed, which is the algorithmic core of the analysis. It should be noted that this module in addition to aggregation and correlation component may include other components necessary for the analysis. After the analysis the data are archived for further work with them from visualization module.

Visualization module consists of a component for data preparation, that converts the data from storage to graphics, and classic pattern "model-view-controller". The user or an information system using a controller are capable of controlling presentation, which can be obtained in the form of a web page or be exported to external systems through API functions.

3.2 Mathematical models of the visualization system components

Let us give the formal description of the overall architecture of the visualization system and denote the restrictions imposed on the elements of the modules that were introduced above. First of all, we need to determine the model of the computer network N consisting of many hosts H and connections C between them as

$$N = (H, C) \tag{1}$$

Further, such network N components as the network itself, the host $h \in H$ or the connection between hosts $c \in C$ will be called information objects that make up the set O. The host model h can include many different attributes, the values of which affect the state of its security. Let us build the model of the host, which includes the most important from our point of view characteristics

$$h = (ID, P, V, t^h, I_{cr}, Sc) \tag{2}$$

where ID is identification information, P – installed hardware and software, V – set of vulnerabilities, $t \in T$ – type of network object (workstation, server, router, firewall, etc.), I_{cr} – the importance of information being processed, Sc – the set of characteristics that can be collected on the host as a result of the external scan. The model of communication between hosts must include directly hosts h_i and h_j, the physical data transmission channels $T' \subset T$ that exist between hosts, as well as hardware and software $U'_t \subset U$ installed on the host and is able to conduct information exchange. Thus, the communication model c will take the form

$$c = (h_i, h_j, T', U'_t) \tag{3}$$

Models of data transmission channels T, as well as models running on top of each channel hardware and software U_t include, respectively, such required attributes, as the characteristics of the source and receiver, the transmission protocol, type (port) of transmission mode:

$$T = \{(p_i, p_j, protocol, type)\}, p_i, p_j \in P \tag{4}$$

$$U_t = \{(p_i, p_j, protocol, port)\}, p_i, p_j \in P, U_t \subset U' \tag{5}$$

Given in (1) (2) (5) models describe the main objects of computing infrastructure, information about which is needed to be collected, processed and displayed by the visualization system. Now let us consider algorithm of network visualization system in accordance with the basic modules of this system. As noted above, the data acquisition module is responsible for the initial reception of information from a variety of sources. Let be the set of all available sources of information about the multitude of information objects O, then each element $l \in L$ is represented in the form $l = (p, c_t)$ where $p \in P$ is hardware or software tool that generates security message, c_t is technique of communication of the information source with the data acquisition module of visualization subsystem.

The elements of the set L can be divided into two classes. The first class contains physical information sources L_p. On the one hand, L_p refers to the network N itself and processes happening inside it, from the other – external for this network information spaces, such as the open network Internet or the security administrator. The second class includes logical sources of information L_l, which are components of the assessment system of security of the information object. Thus,

$$L = L_p \cup L_l \tag{6}$$

Data collection and its passing to the corresponding visualization systems module can be done in active or passive way:

$$c_t = \{a \vee p\} \tag{7}$$

Active way $c_t = a$ is that the information source independently initiates communication with the data acquisition unit, for example, notifying the domain controller about user authentication in the network. Passive way $c_t = p$ is to poll by module of relevant sources of information, such as servers of updates of antivirus programs. Note that techniques to scan the network using security scanners or collecting network traffic belong to the passive techniques. The normalization module transforms the message S^i that comes from a physical source of information $l^i \in L_p$ and has, as a rule, a unique internal format, to the formal data structure:

$$a_{s^i} = parse_{l^i}(s^i) \tag{8}$$

In general, the model of normalized security event coming from a physical source would be

$$a = \{(type, source, severity, timestamp, msg)\}, a \in A \tag{9}$$

where *type* is the message type, *source* is the source that generated the message, *severity* is the importance of the message, *timestamp* is timestamp. The message body *msg* is a structure which fields contain characteristics specific to the object or its component. A logical source of information provides a calculation of security of information objects and is a function of the form

$$f(O') \to R, O' \subseteq O \tag{10}$$

Let us build the model of aggregation and correlation of data coming from information sources, based on the following classification of relationships T^c between hosts:

1. the physical accessibility at which between two hosts there is a physical communication channel;
2. the relationship of the accessibility at which one can establish a data transmission channel using a network protocol;
3. the relationship through the use of virtual communication channels, at which there exist logical communication channels between the hosts, that have specific characteristics from the point of view of security characteristics, for example, with traffic encryption;
4. the relationship of functional dependence, in which to communicate with the main host it is necessary to have connection with the dependent host.
5. the trust relationship, when the communication channel existing between the hosts is not controlled by security policies;

6. the relation of vulnerability in which the vulnerability exploitation on one host leads to compromise of another host.

Let us introduce a multidimensional matrix of connections as

$$M^c = \|m_{ijk}\|, i, j = 1..\vee H\vee, k = 1..\vee T^c\vee \tag{11}$$

its elements are obtained as a result of mapping

$$F^c : (N,N,N) \to H \times H \times T^c$$
$$m_{ijk} = F^c(i,j,k) = \begin{cases} (h_i, h_j, t_k, msg) & \text{if } h_i, h_j \text{ in a relationship } t_k \\ (h_i, h_j, 0) & \text{, else} \end{cases} \tag{12}$$

Thus, the elements of the matrix M^c are the sets (h_i, h_j, t_k, msg), that represent ordered pair of hosts and the type of communication (lack of communication) between them. The value of attribute *msg* may contain additional information on specific type of communication. Let us introduce the procedure of forming the slice of matrix M^c using the fixation of index $k = \tilde{k}$:

$$\pi(\tilde{k}) = \|m_{i,j,\tilde{k}}\|, i, j = 1..\vee H\vee \tag{13}$$

Each slice $\pi(\tilde{k})$ represents the adjacency matrix of a graph, whose nodes are hosts and the presence of the edge determines the presence of relevant link between hosts. The interpretation of each slice are shown in table 1.

$\pi(k)$	Interpretation
$\pi(1)$	Network topology
$\pi(2)$	Availability graph
$\pi(3)$	Graph of protected channels
$\pi(4)$	Graph of dependencies
$\pi(5)$	Trust graph
$\pi(6)$	Graph of attacks

Table 1: Results of function $\pi(k)$ and their interpretation

Note that the matrix (11) represents the information that can be stored in computer memory using a multidimensional array using the structure of the data, presented in the form of heterogeneous objects "key"-"value" as its elements. The specified multi-dimensional array as well as formalized security messages, can be without additional transformations placed in a document-oriented storage.

Visualization module may include different patterns of presentation of the matrix created, however, the present study uses the model of graph structures as the most informative and common.

In general, the matrix slice $\pi(\tilde{k})$ is displayed as an undirected graph $G = (V,E)$, where V are the vertices of the graph representing hosts, E are edges specifying links between hosts.

Let us define the mathematical model for visualization of the specified graph as follows:

$$\chi_k(\pi(k), V_{scale}, G_{type}) = G(G'_k), k = 1..|T^c|, \tag{14}$$
$$G'_k = (V_{scale}, E'), V_{scale} \subseteq V, E' \subseteq E$$

where $\pi(k)$ is the corresponding slice of the matrix M^c, G'_k is the subgraph of G_i containing the vertices from the set V_{scale}, G is procedure of visualization of the graph. Sets V_{scale} and G_{type} are determined by the user and provide geometric scaling and the layout of the visual representation accordingly. The procedure of visualization of the graph G provides the output graph on the screen in the manner specified in G_{type}. Let us consider two approaches to rendering a single host – graph node G. The first approach is to associate each host with its icon containing information, for example, about the type t^h of the node. In this case, the model of visualization is presented as a function of the host

$$D(t^h) = image \tag{15}$$

where t^h is type of host h, *image* is the icon corresponding to the type of the host. The second approach is to associate the glyph to each host, i.e. the image that shows the security metrics calculated for a given host. Glyph base B^t_h to host h at the moment of time t is the set

$$B^h_t = \{f_i | f_i(O^h_i) = x_i\}, i = 1..n, x_i \in R \tag{16}$$

where $O^h_i \subseteq O$ is the set of groups of information objects that contain information about the host h on which it makes sense to display the appropriate security metrics $f_i \in F$. In particular, such a set may consist of one single host h. The mathematical model of the glyph is like

$$G^h_k = \{B^h_{t_1}, .., B^h_{t_k}\}, k \geq 1 \tag{17}$$

4 The technique of data visualization models application

The quality of the visualization system functioning depends on how efficiently the user perceives information during visual analysis of the images provided by the system, and how these images are informative. The informativity, in this case, can be represented as a detail-completeness, which is expressed in the fullness of simultaneously displayed data and metrics, and the efficiency of perception can be represented in the form of speed and ease of data processing. It is obvious that graphical models, which form the basis of visualization systems, have different ratio of efficiency to informativity. Often the increase of informativity negatively affects the efficiency and vice versa. Examples of informative but inefficient tables and uninformative, but efficient semaphore are depicted on the left and right parts of Figure 4.

Input Packet Length	Input Rate	Input Media Overhead (Ethernet)	Total Input Port bytes	MAC removed bytes (including CRC)	Other PP/MAC/FR AMER removed bytes	PP Packet add bytes (to fabric and peer PP)	Switch Port Packet Size	Switch Port effective Input Rate
64	1.00E+10	20	84	24	0	12	72	8.57E+09
128	1.00E+10	20	148	24	0	12	136	9.19E+09
256	1.00E+10	20	276	24	0	12	264	9.57E+09
1500	1.00E+10	20	1520	24	0	12	1508	9.92E+09

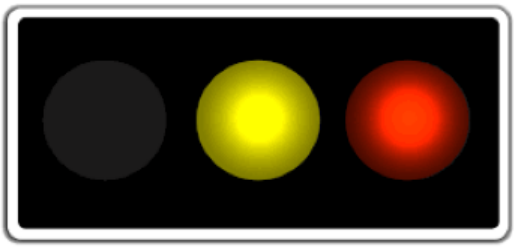


Figure 4: Examples of table and semaphore

To improve the efficiency of visualization systems, the visualization models are developed and implemented which can be described as conceptual graphical models. However, in order to increase the efficiency of visualization systems, the introduction and development of new graphical models should be produced according to the technique, which can provide the necessary ratio of efficiency to informativity in accordance with the scenario of application of visualization system.

Paper [11] describes the basic graphical model used in visualization systems, as well as what the parameters of these models affect the ratio of efficiency to informativity. The informativity of visualization depends on the detalization of data received in the process of collecting, aggregation and correlation, and of graphical models displaying the data. In most systems for visualizing the following graphical models are used, which are selected based on the analyzed data and requirements for the informativity from the side of scenario of the visualization system:

- graphics;
- graphs (including maps of trees);
- geographic maps;
- matrices;
- histograms;
- trilinear coordinates;
- parallel coordinates.

When using classical models for visual data presentation some visual techniques are used, such as the location of an object in space, the encoding of information by using its shape, size and color [11].

It should be noted that the above mentioned models of information objects, although focused on the use of visualization in graph structures, also include information that should be presented in other graphical models. For example, during presentation of key hosts security in a distributed computer network with the help of geographic maps we can obtain information about the location of the host from its identity *ID* and its security state may be represented using base glyph B_t^h of that host.

The efficiency of perception depends on how the graphical model corresponds to the concepts of visualization, such as:

- the pattern of visual search;
- consideration of human cognitive apparatus;
- the consistency of the data and their presentation;
- the control of information noise;
- the presence of direct manipulations in the model;
- concept of graphic design being used in the model.

Thus, the formed image should be clear and informative and should not go beyond the human cognitive apparatus [13].

The required ratio between the efficiency and informativity can also be achieved using of additional tools in graphical models, such as ”fisheye” [14], ”multiple view”, ”semantic zooming” [15], ”small differences” [16] and others. It is also important to provide the user with an opportunity to get comprehensive information about selected object using visual search tools [17].

Thus we can conclude that the ratio between the efficiency and informativity depends on the technique of implementing of the graphical model that is used in the visualization system, and the approach

to the collection and analysis of data that will be visualized. Therefore the developed technique of application of the models should include the process of data collection and analysis, as well as the process of implementation of graphical models in the system.

The technique of applying visualization data to improve the efficiency of the visualization system is shown in Figure 5. Technique consists of six steps that show the process of collection, preparation of data for visualization as well as the process of implementation of the graphical model.

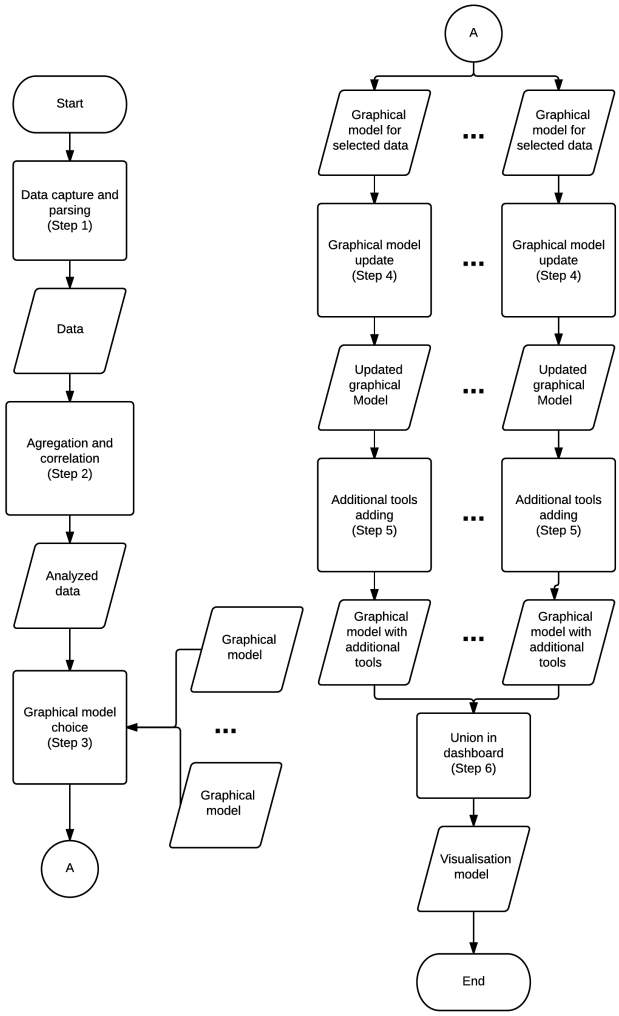


Figure 5: Technique of implementation of data visualization models for improvement of efficiency of visualization system

Step 1. The selection of data sources L. At this stage we must determine which system will be responsible for data collection with regard to the scenario of application of the visualization system. As each data collection system provides data in a unique format S^i, it is necessary to develop parsers (8) to the data acquisition module and to define the interfaces of interaction of data acquisition systems with the visualization system (7), as well as to develop algorithms of normalization to convert the data to the internal structure (9).

Step 2. The choice of algorithms for aggregation and correlation. This stage is the most significant at the stage of formation of data for visualization. Based on the selected algorithms and sequence of their

application data analysis will be performed, as well as data heir integration and event generation (11). The informativity of the visualization depends from this stage, and therefore, on the basis of final data of this phase there will be done the selection of the graphical model for visualization.

Step 3. The formation of the list of graphical models. At this stage it is necessary to select those graphical models that allow to visualize data resulting from the previous step. When choosing models we must consider the degree of informativity that the chosen model can provide. Criteria for selection can be: the detailization of the data, possibility for simultaneous display of all data as a whole, the ratio between the amount of data metrics to the number of metrics supported by graphics model, etc. Accordingly, when choosing the model, we also consider the visualization system usage scenario.

Step 4. Change of graphic models to ensure the appropriate ratio of informativity to efficiency. For each previously selected graphical model it is necessary to check how the graphical model corresponds to the concepts of visualization and apply the constraints or addition of new elements to the graphical model. It is necessary to consider that, as a rule, increasing the efficiency of perception has a negative impact on informativity. Changes need to be applied in such a way that the highest possible level of efficiency of perception from the side of visualization system scenario would be achieved. At the same time the changes to be applied should not have significant negative effect on the informativity, as well as to neutralize the advantages of the selected in the third step graphical model.

Step 5. Adding additional tools. For further adjustment of the ratio of efficiency to informativity, we can consider adding additional tools to the graphical model. Adding of new tools must be performed with the constraints described in step 4.

Step 6. Combining of all displays to the information panel. The final stage is the combining of graphical models, tools for work with graphical models, tools for working with modules of data acquisition and analysis into an unified visualization system.

5 Example of the application of the models and technique

Let us consider the application of the proposed models and technique on the example of visualization of the network topology being performed for security monitoring. For this purpose we have developed a software prototype of the visualization system that implements the following rendering models: (1) directional and non-directional graph; (2) directional and non-directional graph using the glyphs to display metrics; (3) the matrix; (4) map trees; (5) the Voronoi diagram [11].

The prototype is a web server that is deployed using the Apache Tomcat servlet container. Part of the server, which is responsible for preparing data for visualization, is implemented using Java EE, Spring framework, services JAX-WS and servlets. The visualization view (part of the pattern ”model – view – controller”, Figure 3) is implemented in HTML pages using JavaScript D3js visualization libraries. Thus, the access to the visualization model is done from computer or mobile phone through the browser.

5.1 Description of source data

It is assumed that the user needs to visualize a computer network consisting of 10 hosts located in three segments. Visualization is necessary for analyzing network topology and security status monitoring of network objects. In order to display computer attacks we use an approach based on attack trees [18]. For visualizing the hosts' status the following metrics are used: (1) the criticality of the information located on the host; (2) existence of vulnerabilities with different levels of complexity of exploitation; (3) possible damage caused by a successful attack on the host; (4) the probability of attack on this host in the context of past actions of intruder [19]. It is assumed that the network size can be increased 2-3 times, because some of the workstations can be turned off and the network users can connect their mobile

devices and notebooks to the network via Wi-Fi.

Let us consider three examples of applications of the technique with consideration of changes in source data requirements to visualization:

1. The number of hosts in the computer network equals to 10. It is required to visualize the network topology displaying the network objects' types.
2. The number of hosts in the computer network equals to 10. It is required to visualize the network topology displaying four types of metrics for each host.
3. The number of hosts in the computer network equals to 100. It is required to visualize the network topology displaying four types of metrics for each host.

5.2 Example 1. Visualization of a small computer network topology

In accordance with the technique at the first stage it is necessary to ensure the collection of data. To visualize the topology and network parameters, the most important are the physical sources of information L_p. For example, active and passive scanners, such as Nmap and Wireshark, can be used for that. Depending on the scenario of application of the visualization system, systems whose data do not contain information on the state of the network, also can serve as sources, however, such data can be obtained in the process of correlation. The list of such systems is rather wide, they can be antiviruses, embedded devices of the Internet of Things, systems of collecting and processing event logs, etc. From the entire set of messages being collected from sources we need to select only those that contain information about the hosts and the physical channel between them:

$$A_p = \{a \vee a \in I(h_i, h_j, T'), a \in A\} \tag{18}$$

The set of messages A_p allows us to fill the attributes of communication model (3), and it is possible to assume that $U' = \emptyset$.

It should be noted that it may not be enough to use the software in order to collect reliable information about the state of the network. For example, the passive analysis tools detect only hosts sending or receiving network traffic, while active tools can only detect hosts that are responding requests. It may also relate to the importance $I_c r$ of processed on the host information that it is impossible to determine by automatic techniques. Therefore, as an additional data source for the visualization system, a system supporting the data entry by the operator may be used, thereby providing not an automatic, but automized data collection. Usually this combination, despite the need to involve a person to the collection process, provides greater reliability of the data that will be visualized. Collection systems can be connected using the API of the visualization system. If the collection system provides data in a specific format, an additional parser (8) must be downloaded into the data acquisition module of the visualization system, and the normalization module should be updated with a component allowing to convert the information received into the internal data structure of the system (9).

The second step is to provide the analysis of information gathered from the sources. For this purpose it is necessary to carry out the correlation and aggregation of data, since different sources may provide the same information with some differences. For example, a heterogeneous data about network activity, network packets characteristics, information received from the system operator, etc. may be used to form a multidimensional matrix of connections (11) with $k = 1$ (physical accessibility ratio). The result of the slice of the matrix $\pi(1)$ will be the adjacency matrix that specifies an undirected graph, whose nodes are hosts and edges are physical communication channels.

The next step is to choose the graphical models that allow us to visualize the analyzed data. In most existing systems graphs, matrices, and maps of trees are used for computer network data visualization.

The above listed models have different advantages depending on the usage scenario. Matrices are good for displaying computer networks with a complex topology, where each host has a set of connections. Maps of trees are efficient in displaying metrics of purely hierarchical networks. Graphs are efficient for visualizing small computer networks or networks, which hosts do not contain multiple connections. However, as a rule, when the visualization system is being created, there is no certainty about the composition of the data, and the scenario of system usage may be defined only in general. In such cases it is advisable to follow the paradigm of multiple view, when the same data is represented with different graphical models, each of which has its advantages in certain scenarios. These three graphical models meet the selection criteria (the ability to display network topology and metrics of the hosts), however on subsequent steps we will consider only the graphical model of graph (14) (Figure 6), as the most simple, but at the same time the most common graphical model. The subsequent steps may also be applied to maps of trees and the matrices since there is no reason for which they do not meet the specified requirements, however, in this example, work with them in step 4 and step 5 is skipped.

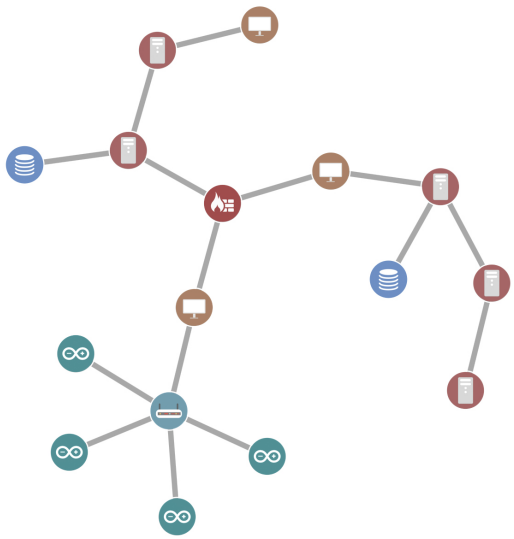


Figure 6: Example of visualization based on graphic model of graph

5.3 Example 2. Visualization of a small network topology with the security metrics

Let us consider the graphical model of the graph from the point of view of the new requirement and taking into account the models of the visualization that are listed in section 4. The usage scenario involves the construction of visualization for network topology analysis and values of multiple metrics for each host and links between the hosts. The network topology is already presented in a graph by vertices and edges, so from this point of view it is not necessary to change anything. In order to display metrics of connections between hosts it is necessary to change the display of the graph's edges. Edges can display at least 4 metrics using color of edge, transparency of edge, thickness of edge and direction of edge. It should be noted that in the case of bidirectional communication there is an overlapping of the edges, and one cannot display metrics as the length of edges, edges thickness and color of edges (Figure 7). However, simultaneous display of all four metrics communication possible when displaying edges as shown in Figure 8. In this case the edges are arranged in a sequence, with the respective location of the arrowhead indicating the direction.

Figure 7: Overlap of each other by edges with different directions, color, transparency and thickness

Figure 8: Sequential display without overlapping of each other by edges with different directions, color, transparency and thickness

It should be noted that in this case the efficiency of information perception may not be increased, since the graphical model is overloaded and from the point of view of cognitive perception the user will be spending disproportionately much time analyzing the links. Suppose that the criterion of informativity suggests the presence of only one metric – direction of communication, and consistent addition of edges is not an advantage. Thus, for the purpose of increasing the efficiency of perception, it is advisable to display edges as in Figure 8.

From the point of view of the graph visualization model (14), the selection of edges (routes), represented by changing the color scheme, would be

$$\eta^e(H') = color(\{e_{ij} | e_{i,j} = (v_i, v_j), v_i, v_j \in H', H' \subseteq H\}) \tag{19}$$

In the case of hosts that are displayed on a graph as vertices, there are several ways to display metrics: vertex size, vertex color, vertex opacity. We can also use the form of vertex (round, triangle, etc.), however, many forms of tops will have a negative impact on cognitive perception, and it would be appropriate to use if goal is increase of informativity. Transparency should also not be used because it will have to be restricted with the lower threshold value, for example 30 percent, as in the case of greater transparency of the vertices will be faded. Thus, it remains possible to operate only with the color of vertices and its size. However, instead of the standard vertices display (15) as in Figure 6 we can use glyphs (17), for additional display metrics (Figure 9). In this form the glyph is a circle and a ring, which are divided into 4 sections. Each section corresponds to a metric which value corresponds to the color. Sections of the circle represent the metric in the current moment of time, and the sections of the ring – in the previous moment, thus allowing to carry out analysis in time. The numerical value of each partition is stored in the glyph base (16) for each host. The usage of glyphs allows us to increase the number of

displayed metrics, while not having negative impact on the efficiency of perception. Thus, as a result of the fourth step, the graphical model of graph may look like in Figure 9.

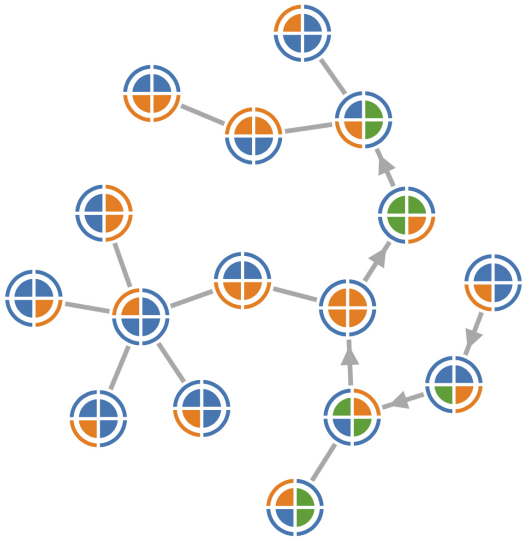


Figure 9: Example of a graph using glyphs

5.4 Example 3. Visualization of an average computer network topology considering the security metrics

In order to implement a visualization for the changed source data is necessary to determine what additional tools for work with the visualization system can minimize the disadvantages of graphical models. In the case of the graph the main disadvantage is its overload in the case of a large number of vertices or a large number of edges. The criterion of overload may be set as a relative indicator of the number of the overlapping elements. For example, such elements may be edges if the graph is not planar. It should be noted that if the graph's overload is significant, the methodology will result in recommendation of using the visualization model in the form of a matrix. However, in this case, when the usage scenario assumes exactly the visualization of the topology, and the number of hosts still allows you to display the computer network on the screen, the technique proposes using the tool "Fish eye" as a zooming tool (Figure 10). In other aspects the constructed graphical model corresponds to the required level of informativity and efficiency and the use of additional tools is not required.

5.5 Conclusions on the discussed visualization examples

At the final stage it is necessary to combine all the constructed graphical models and their controls into the unified dashboard so that the concept of multiple look be fulfilled. In the case of a change of the source data the proposed model technique of visualization may change. Thus, to avoid cognitive dissonance, when one model is replaced by another, the technique adds a new model to the dashboard and alerts the user that there is a more efficient technique of display for a given set of data.

Thus, for described at the beginning of the section usage scenario and its modifications, the technique can offer 3 models of visualization (table 2). In the first case, when network is represented by a small number of hosts, and the usage scenario involves the use of visualization model for network topology

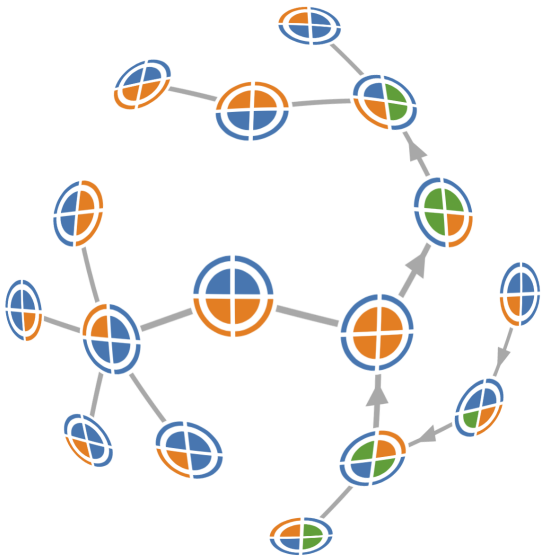


Figure 10: Example of a graph using the lens ”Fish eye”

analysis it is advisable to use a graph. In the second case, when the requirement to display the metrics is added, it is advisable to use a directed graph with glyphs. In the third case, when the number of hosts increases significantly, the model is complemented with additional visualization tool “Fish eye”.

Example	**Number of hosts**	**Features to be visualizated**	**Visualization model proposed by the technique**
1	10 hosts	Topology	Graph
2	10 hosts	Topology, security metrics	Graph with glyphs
3	100 hosts	Topology, security metrics	Graph with glyphs and usage of tool ”Fish eye”

Table 2: Visualization models proposed by the technique

Thus, the application of visualization models using the described technique allows to achieve a required ratio of informativity to efficiency and high performance of visualization systems in comparison with conventional application of models of visualization. Thus, the proposed visualization technique depends entirely on the data displayed and the scenario defined by the user.

6 Discussion

Today, with the increase in the volume and heterogenity of information the problem of its visualization has become urgent. This encourages researchers to create new conceptual visualization models and to improve already existing, which has a positive effect on their amount. With the increase of computer networks and the emergence of new technologies the number of information sources is also increasing, and due to necessity to detect new and sophisticated computer attacks new metrics and, as a result, new data collection systems appear.

As the number of visualization models is becoming greater, as well as the number of data sources, the generalization of the visualization system architecture and the visualization technique to improve the efficiency of visualization systems allows to navigate the vast choice of models and architectural

components in case of creation of own visualization system. Due to proposed in the technique selection and addition of the models, which is based on the compromise of data informativity and the efficiency of the user perception, the greater efficiency is achieved in the usage of visualization models and as a consequence the efficiency of the visualization system itself increases.

The advantage of the proposed technique and the developed visualization system is displayed in their adaptability to different usage scenarios that were shown in the example. This results in more efficient interaction between the user and visualization system. It should be noted that if the technique determines that a different model fits better for visualization for a given dataset, in order to avoid cognitive dissonance, the decision to change the type of display is up to the user. Another advantage of the developed visualization system is its flexibility: possibility of varying the sources of information, correlation and mapping rules.

It should be noted that the ability to connect different data sources to the visualization system means that the user will need to develop parsers, components for normalization and correlation rules. Also, the isolation of the system means that it does not consider the architecture of systems that communicate with it, for example how it will connect to the data source of scanner of SIEM system, which is a disadvantage. However, the ability to connect different data sources and isolation of the system provides a more wide range of its applications. This system, as well as the technique, is not tied to any particular usage scenario, and depends solely on the reliability of the data provided by the sources.

It is possible to consider separately the implementation of visualization tools in existing SIEM systems. For example, in the OSSIM SIEM system [20] the visualization of security settings is represented by several universal models, such as histograms, pie charts, graphs and tables. The analysis of other systems and scientific papers have shown that other SIEM systems also, in most cases, use simple and universal models of visualization. However, for analyzing large volumes of data, as well as the data that include many metrics, it is advisable to use more complex visualization models, using which we can identify patterns and, as a consequence, to produce more efficient visual analysis. In this regard, there is a trend of implementing more complex visualization models.

On the other hand, when using complex models of visualization users are confronted with the fact that the models are overloaded with graphic elements and are difficult to read. The problem of choosing visualization models and adaptation of them to each specific usage scenario of visualization system arises. The solution to this problem may be obtained from the proposed technique based on combination of complex visualization models and taking into account the architecture of the visualization system and the processes that precede the display of data.

7 Conclusion

The proposed architecture and technique are designed to create a comprehensive vision of the process of constructing a system for data visualization, including processes of data collection and analysis. To describe these processes the mathematical models of visualization of computer networks using graph and tree structures are suggested.

In the paper the notion of information object as a mandatory component of the computer infrastructure is introduced, and mathematical models are presented that contain attributes needed to analyze the security of the computer network. On the basis of the classification of types of relationships between information objects multidimensional matrix of information relations between objects is constructed and it is shown how the slices of the specified matrix can represent computer infrastructure from different viewpoints and focus the operator's attention on specific problem areas of the network.

The newly introduced notion of ratio of informativity and efficiency takes into account the informativity of the data represented by visualization and efficiency of these data perception by the user. On

the basis of this ratio the technique of applying data visualization to improve the efficiency of visualization systems is done, which allows to build a visualization system according to the earlier presented architecture, as well as to more efficiently use the system with graphical models for data visualization.

The paper provides the example that illustrates usage of models and technique for building a visualization system of a computer network, the example of usage of graph in accordance with the required ratio of efficiency and informativity is presented.

In further research it is planned to expand the list of used visualization models, to consider in more detail the general architecture of generalized visualization system and possibilities of its practical application to the problems of security monitoring.

Acknowledgment

This research is being supported by the Ministry of Education and Science of The Russian Federation (contract 14.604.21.0137, unique contract identifier RFMEFI60414X0137) in St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS).

References

[1] E. Novikova and I. Kotenko, "Visualization of Security Metrics for Cyber Situation Awareness," in *Proc. of the 9th International Conference on Availability, Reliability and Security (ARES'14), Fribourg, Switzerland*. IEEE, September 2014, pp. 506–513.

[2] E. Chi, "A taxonomy of visualization techniques using the fata state reference model," in *Proc. of the IEEE Symposium on Information Visualization (INFOVIZ'00), Washington, DC, USA*. IEEE, October 2000, pp. 69–75.

[3] K. Lakkaraju, W. Yurcik, and A. Lee, "NVisionIP: netflow visualizations of system state for security situational awareness," in *Proc. of the 11th ACM Conference on Computer and Communications Security (CCS'04), Washington, DC, USA*. ACM, October 2004, pp. 65–72.

[4] R. Ball, G. Fink, and C. North, "Home-centric visualization of network traffic for security administration," in *Proc. of the 11th ACM Conference on Computer and Communications Security (CCS'04), Washington, DC, USA*. ACM, October 2004, pp. 55–64.

[5] R. Erbacher, "Visualization Design for Immediate High-Level Situational Assessment," in *Proc. of the 9th International Symposium on Visualization for Cyber Security (VizSec'12), Seattle, WA, USA*. ACM, October 2012, pp. 17–24.

[6] W. Matuszak, L. DiPippo, and Y. Lindsay, "Sun CyberSAVe – Situational Awareness Visualization for Cyber Security of Smart Grid Systems," in *Proc. of the 10th Workshop on Visualization for Cyber Security (VizSec'13), Atlanta, GA, USA*. ACM, October 2013, pp. 25–32.

[7] F. Gordon, *Nmap Network Scanning*. Insecure, 2009.

[8] A. Orebaugh, G. Ramirez, and J. Beale, *Wireshark and Ethereal*. Syngress, 2007.

[9] P. Gomer and J. Johnzon, "Computer Network Analysis by Visualization," Master's thesis, Department of Computer Science and Engineering, Division of Networks and Systems, Chalmers univercity of technology, Goteborg, Sweden, 2010.

[10] G. Battista and M. Rimondini, *Computer Networks*. CRC Press, 2013.

[11] K. Kolomeec, A. Chechulin, and I. Kotenko, "Methodological Primitives for Phased Construction of Data Visualization Models," *Journal of Internet Services and Information Security*, vol. 5, no. 4, pp. 60–84, November 2015.

[12] A. Bondarev and V. Galaktionov, "Analysis of multi-dimensional data in the problems of multi-parameter optimization with the use of imaging techniques," *Scientific visualization*, vol. 4, no. 2, pp. 1–13, 2012.

[13] B. Goldstein, *Cognitive Psychology*. Thomson Wadsworth, 2005.

[14] M. Sarkar and M. Brown, "Graphical fisheye views," *Communications of the ACM*, vol. 37, no. 12, pp. 73–83, 1991.

[15] G. Watson, "Lecture 15 - Visualisation of Abstract Information," Edin-burgh Virtual Environment Centre, 2004.

[16] L. Wroblewski, "Small Multiples Within a User Interface," http://www.uxmatters.com/mt/archives/2005/12/small-multiples-within-a-user-interface.php, December 2005, [Online; Accessed on March 30, 2016].

[17] D. Ferebee and D. Dasgupta, "Security Visualization Survey," in *Proc. of the 12th Colloquium for Information Systems Security Education (CISSE'08), Dallas, Texas, USA*. CISSE, June 2008, pp. 119–126.

[18] A. Chechulin and I. Kotenko, "Attack Tree-Based Approach for RealTime Security Event Processing," *Automatic Control and Computer Sciences*, vol. 49, no. 8, pp. 701–704, 2015.

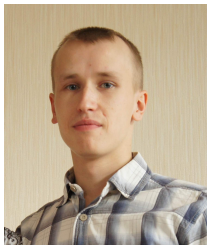
[19] I. Kotenko, E. Doynikova, and A. Chechulin, "Security metrics based on attack graphs for the Olympic Games scenario," in *Proc. of the 22th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP'14), Torino, Italy*. IEEE, February 2014, pp. 561–568.

[20] "SIEM OSSIM," https://www.alienvault.com/products/ossim, accessed February 2016.

Author Biography

Maxim Kolomeec is currently pursuing Specialist degree in Information Security at the Saint-Petersburg Electrotechnical University "LETI", Russia. He is working as a developer at the Laboratory of Computer Security Problems of St.Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS). His research interests include distributed system security, and security visualization.

Andrey Chechulin received his B.S. and M.S. in Computer science and computer facilities from Saint-Petersburg State Polytechnical University and PhD from St.Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS) in 2013. In 2015 he was awarded the medal of the Russian Academy of Science in area of computer science, computer engineering and automation. At the moment he holds a position of senior researcher at the Laboratory of Computer Security Problems of SPIIRAS. He is the author of more than 50 refereed publications and has a high experience in the research on computer network security and participated as an investigator in several projects on developing new security technologies. His primary research interests include computer network security, intrusion detection, analysis of the network traffic and vulnerabilities.



Anton Pronoza is currently a PhD student at the Laboratory of Computer Security Problems of St.Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS). His research interests include security visualization and big data analysis.

Igor Kotenko graduated with honors from St.Petersburg Academy of Space Engineering and St. Petersburg Signal Academy. He obtained the Ph.D. degree in 1990 and the National degree of Doctor of Engineering Science in 1999. He is Professor of computer science and Head of the Laboratory of Computer Security Problems of St. Petersburg Institute for Informatics and Automation. He is the author of more than 250 refereed publications, including 12 textbooks and monographs. Igor Kotenko has a high experience in the research on computer network security and participated in several projects on developing new security technologies. For example, he was a project leader in the research projects from the US Air Force research department, via its EOARD (European Office of Aerospace Research and Development) branch, EU FP7 and FP6 Projects, HP, Intel, F-Secure, etc. The research results of Igor Kotenko were tested and implemented in more than fifty Russian research and development projects.