Visualization Model for Monitoring of Computer Networks Security Based on the Analogue of Voronoi Diagrams

Maxim Kolomeets, Andrey Chechulin, Igor Kotenko

Visualization Model for Monitoring of Computer Networks Security based on the Analogue of Voronoi Diagrams

Maxim Kolomeets, Andrey Chechulin, and Igor Kotenko

Laboratory of Computer Security Problems
St. Petersburg Institute for Informatics and Automation (SPIIRAS)
39, 14 Liniya, St. Petersburg, Russia
{kolomeec, chechulin, ivkote}@comsec.spb.ru

Abstract. In this paper we propose an approach to the development of the computer network visualization system for security monitoring, which uses a conceptually new model of graphic visualization that is similar to the Voronoi diagrams. The proposed graphical model uses the size, color and opacity of the cell to display host parameters. The paper describes a technique for new graphical model construction and gives examples of its application along with traditional graph based and other models.

Keywords: visual analytics, visualization of security data, graphical models, computer networks, Voronoi diagrams.

1 Introduction

Computer networks are rapidly growing today. Meanwhile, the more devices are in the network, the harder it is to ensure its security. This problem is met by operators of security systems of corporate level (e.g., security information and event management systems, SIEM systems), when the analyzed computer network is measured not only by hundreds of employees' workplaces and high order technical equipment, but also by smart doors, servers, various sensors of climate, security, etc.

To cope with the control of growing networks we need to apply systems for monitoring network security, which give us possibility to visualize the computer network and parameters of its state in a simple and efficient manner. But, as a rule, in such systems the network is presented with the application of rather traditional graphical models, for example, in the form of graphs or tables, that are difficult to understand in the case of large networks and display of a variety of parameters. In order to cope with this problem it is necessary to improve the efficiency of visualization means by complex use of various graphical models such as graphs, matrices, treemaps, parallel coordinates, etc. in the framework of the multiple view concept [1]. At the same time it is necessary to increase the efficiency of visualization of particular graphic models.

In the case of visualization of computer networks and their security, different techniques are developed that allow clustering of segments of the network (e.g., based on clustering of graph elements) or encapsulation of the state parameters [2]. Yet another solution to the problem is to develop conceptually new graphical models, which

are able to present information in a form that is new for the user and that allows to increase the efficiency of the user's work.

The novelty of this paper is to use a conceptually new graphic visualization model similar to the Voronoi diagrams, which allows to increase the effectiveness of visual analysis for the computer network security, for example, as one of functions of the SIEM system. It is expected that this model will be used in the developed visualization system in the framework of the supported multiple view concept.

The main contribution of the paper lies in the fact that it offers a new technique of visualization of network security, as well as reveals the theoretical and practical side of how this technique can be used in SIEM systems. The organization of the paper is as follows. Section 2 analyzes existing graphical models that can be used to visualize parameters of computer network security with description of their advantages and disadvantages. In section 3 we describe the developed conceptually new graphical model. Section 4 discusses the developed system for visualization of computer network security and provides examples of application of the proposed graphical model in the framework of this system. In section 5 and 6 the proposed graphical model is evaluated, as well as its comparison with other graphical models is performed. Section 7 discusses conclusions and future research directions.

2 Review of computer network visualization techniques

Within the developed computer network visualization system, graphs, matrices, and treemaps were mainly used to realize the multiple view concept. The listed graphical model (Fig. 1) have different advantages and disadvantages [3], which can also be expressed in terms of informativity and ease of perception and use.

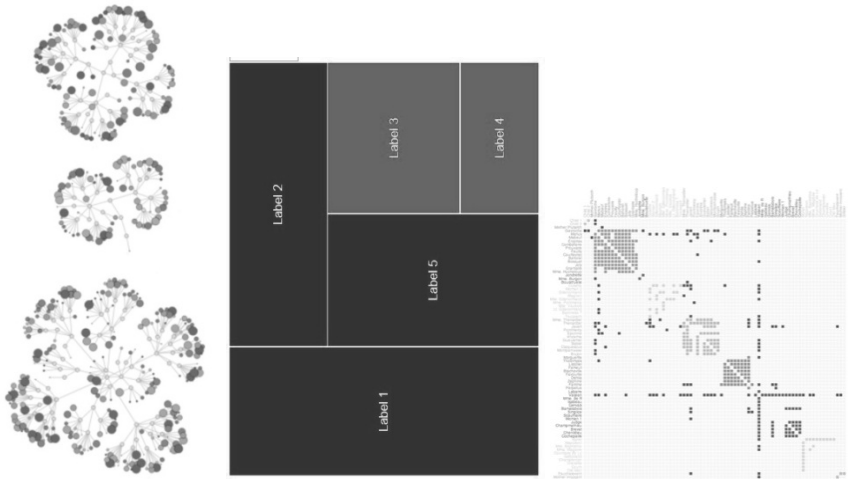
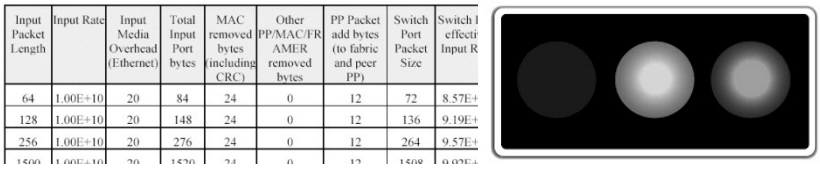


Fig. 1. Examples of graphical models (left to right): graphs, treemaps and matrices

Informativity can be represented as the detailization level, which is expressed in the compliteness of simultaneously displayed data. Ease of perception and use can be represented in the form of speed and simplicity of user interactions with displayed

data. It is obvious that different graphical models have different ratio of informativity and ease of perception and use. Thus, often increase of the informativity due to the resulting congestion of the graphical model negatively affects the ease of perception and use, and vice versa. Some examples are an informative but difficult to read table, and an uninformative, but easy to understand semaphore shown in Fig. 2.



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 I effecti Input R
64	1.00E+10	20	84	24	0	12	72	8.57E+
128	1.00E+10	20	148	24	0	12	136	9.19E+
256	1.00E+10	20	276	24	0	12	264	9.57E+
1500	1.00E+10	20	1520	24	0	12	1508	9.82E+

Fig. 2. Examples of a table and a semaphore

Let us consider from this point of view the graphic models [4-5], which are most often used to visualize the security of computer networks. Matrices [6] (Fig. 1, right) are efficient in displaying the relations of elements of a small computer network that has complex topology and where each host has many connections. Security parameters, determined based on network traffic, can be set using the color and transparency of cells, located at intersections of rows and columns. However, the size of the cells depends on the dimension of the matrix, which is set by the number of network hosts. With the increase in the number of rows and columns of the matrix, the size of a particular cell goes to one pixel, the perception of the color tone and the more transparency of which is difficult. Inefficient use of space of matrices should be noted, when most of the cells remains blank, which has negative effect on the users' perception of parameters, especially when large computer networks are visualized. It is also worth considering that the matrices can visualize the links parameters, but not the parameters of the computer network hosts.

On the other hand, matrices efficiently display clusters of network, and they can be used to construct attack graphs [7] (Fig. 3, left) and to analyze entire segments of the network, rather than individual hosts.

Matrices can also be expanded by displaying multidimensional data in 3D space. For example, the 3D analogue of a matrix is the dispersion chart [8], which is represented as a cube which axes are local IP addresses, global IP addresses and port numbers, and the color of the dot shows successful and unsuccessful attempts to establish TCP connections (Fig. 3, right). Thus, the presence of long lines or planes, consisting of the points on the dispersion chart, can inform about network scanning.

For efficient visualization of host parameters (for example, the data of the vulnerability scanners [6] (Fig. 4, left), assessment of criticality of assets, etc.) of a computer network, one can also use treemaps [9] (Fig. 1, in the center). Treemaps are efficient to display parameters, as they can handle the color, depth and size. It should be noted that in the presence of cells of large area, one can also use transparency as an additional parameter. However, treemaps are suitable to visualize purely hierarchical networks and are unable to visualize the interaction parameters. However, along with matrices, treemaps can be expanded, for example, for constructing attack graphs [10], when the steps of the intruder are displayed by directed edges (Fig. 4, right).

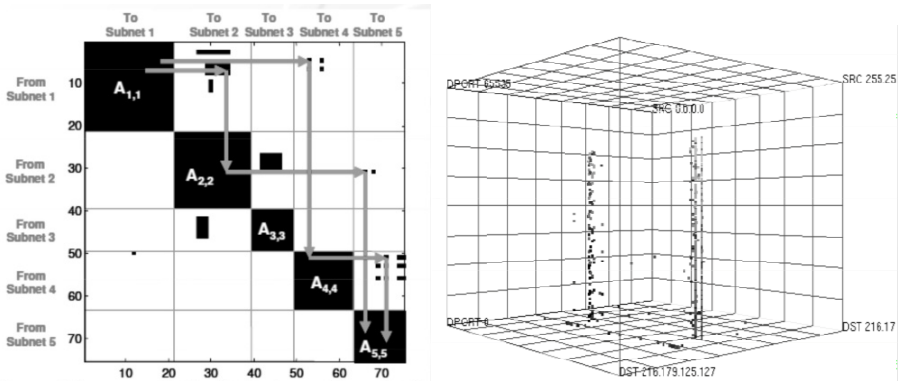


Fig. 3. Usage of a matrix to visualize attack graphs (left) and as the dispersion chart (right)

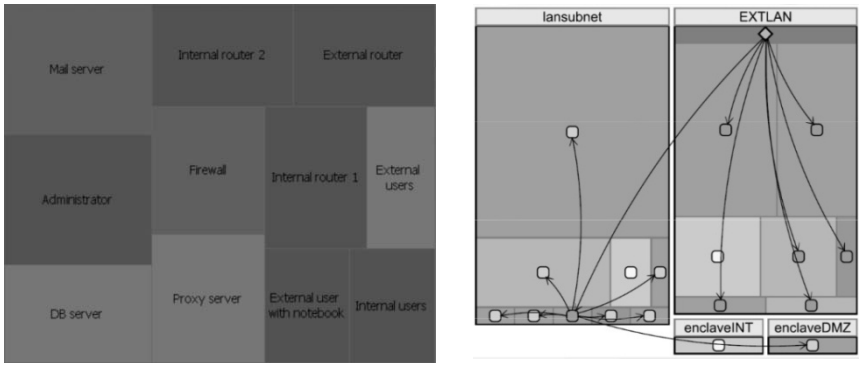


Fig. 4. Visualization of security vulnerabilities (left) and attack graph (right) using treemaps

Graphs [11], as the most common way to visualize computer networks, are efficient for topology visualization and can display parameters of hosts using the vertices of the graph, as well as interaction parameters using edges. However, like matrices, graphs inefficiently use space, leaving large empty areas. However, graphs are often used to display the network topology, when host type is used as vertices (Fig. 5).

For visualization of parameters vertices of graphs can be replaced by glyphs [12]. Glyphs can be represented in a pie chart in which the number of equal sized segments depends on the number of displayed parameters (Fig. 6). The parameters themselves can be expressed in the form of segments' color and their transparency. The glyphs can be augmented by the ring with the same number of segments that display the previous parameter value. Due to this, it is possible to produce a historical analysis.

The disadvantage of glyphs is that when they are used the graphical model becomes overloaded, thereby ease of perception deteriorates. It is worth saying that the graphs have many variations of how they are built and used. For example, graphs can be used for visual analysis of granting access to resources [13] with the role access system that supports hierarchies or groups of users, to visualize patterns of network traffic [14], for visualizing logs of financial transactions [12], for visualization of computer attacks [15,16], etc.

Fig. 5. A graph displaying host type

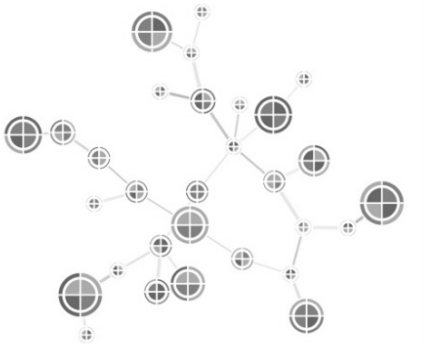


Fig. 6. A graph, augmented by glyphs

At the analysis of existing graphic models the features were outlined that allow the user to efficiently analyze information. First, the user perceives better spatial mapping (plane and spacial figures), while the color and shape of figures are optional parameters. It is easy to demonstrate: the most important security parameters in treemaps (Fig. 4) are displayed with size of the planes; graphs (Fig. 5 and Fig. 6) operate with the sizes of the vertices; and at rendering in the form of a matrix (Fig. 3) the operator most often searches for and analyzes structures in the form of planes and lengthwise lines, consisting of individual cells. Second, it can be concluded that to visualize the links parameters it is better to use matrices, but they are unable to visualize hosts. For visualization of hosts parameters it is better to use treemaps, but they are unable to visualize links. In the case when it is necessary to visualize both hosts and links one needs to use graphs. Thus, the analysis of the advantages and drawbacks of existing graphical models showed that the simultaneous display of parameters of hosts and links is only possible with the use of graphs. However, if the vertices in the graphs are displayed in the form of planes, the operator will be able to analyze the information faster. Thus, analysis of relevant works showed that the creation of graphical models that will allow to effectively display both parameters (as in the treemaps) and the topology (as in graphs) is a perspective approach.

3 The proposed graphical model

For monitoring of computer network security a graphical model is proposed [17], which visually resembles Voronoi diagram [18], however, it is not the same from a mathematical point of view (Fig. 7). The main idea is to integrate capabilities of graphs to visualize the topology of the computer network and treemaps to visualize parameters. The solution appeared from the representation of network hosts in the form of cells, and links between hosts in the form of links between these cells. At the same time in the graphical model there are separators (dark grey in Fig. 7) which divide the cells that are next to each other, but have no links. For ease of understanding, we can present an analogy in the form of a maze: cells-polygons that represent hosts can be interpreted as the maze rooms; the links of the cells that represent the relationship between hosts can be interpreted as doors between the maze rooms; the separators that represent the lack of links can be interpreted as the maze walls.

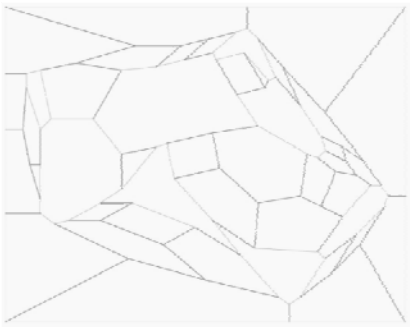


Fig. 7. The proposed graphical model

The algorithm for constructing the proposed graphical model is more complex than the algorithms for constructing graphs, matrices or treemaps, and consists of four steps: (1) building of the convex hull of a given planar graph; (2) implementation of the restricted Delaunay triangulation [18]; (3) formation of cells, based on triangulation; (4) selection of separators.

Let us consider the algorithm for constructing the proposed graphical model in more detail, on the example of the implemented software tool that provides a visual interface to display the security parameters of computer networks.

The proposed graphical model is implemented on the basis of the graph adjacency matrix. The first step builds a planar graph, which is supplemented by the convex hull. The convex hull is required for obtaining the convex figure, which is used to perform the next step. Graph which will be used in this example, and the result of the first step are shown in Fig. 8 and Fig. 9 respectively.

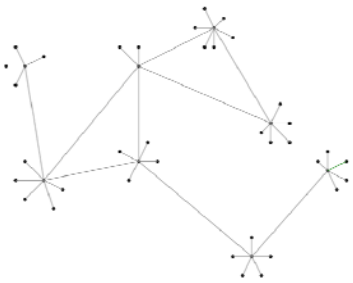


Fig. 8. Planar graph

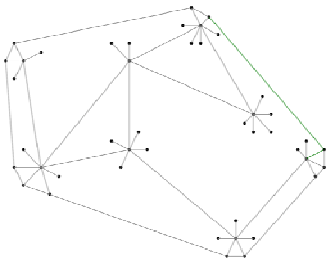


Fig. 9. The construction of the convex hull

In the next step for the resulting figure we should produce a restricted Delaunay triangulation [18]. It is worth noting the importance of implementing exactly limited triangulation, as it allows to triangulate the figures taking into account already existing relations and to avoid crossing of edges. The result of this step is shown in Fig. 10. The next step is to form the cells that will serve as the basis of the graphical model. For this we need to associate a subset of the triangles, obtained as the result of triangulation, with the corresponding vertex of the graph.

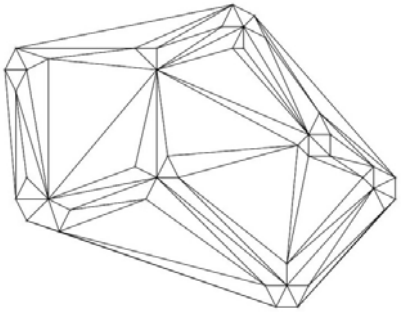


Fig. 10. The graph and his triangulation

For each vertex of the host (in Fig. 11 it is selected as a light grey circle) we find a subset of triangles (in Fig. 11 they are highlighted in gray) that includes this vertex. Next for triangles of this subset the weight centers are determined (in Fig. 12 they are shown as light gray points), union of which gives the desired polygon (in Fig. 12 it is highlighted in light gray edges).

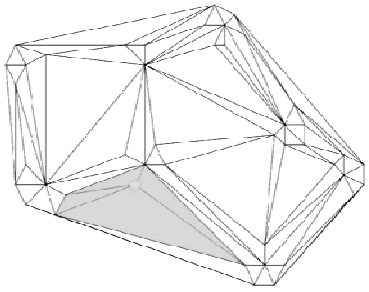


Fig. 11. The determination of triangles required to build a cell

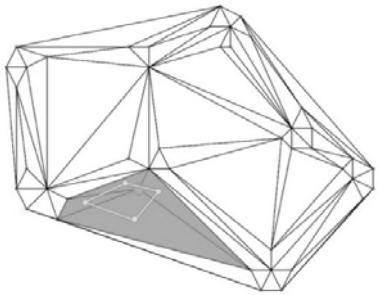


Fig. 12. The construction of the cell

The resulting polygon corresponds to the host, based on which we defined the subset of triangles of the triangulation. The result of the step is shown in Fig. 13, where the edges of the figure resulting from the triangulation are black, and the edges of the desired cells are red (in Fig. 13 it is light grey).

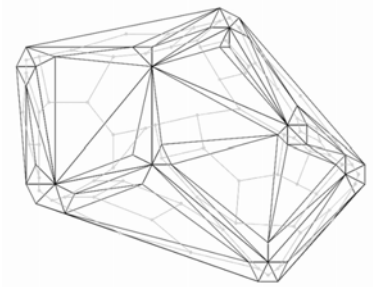


Fig. 13. The result of building the cells of the graphical model

The next step is to outline the separators. Since each cell corresponds to a specific host, the edges of the cells-hosts, with which there is a link, can be designated a certain color, for example gray. All other edges will be separators and will have an appropriate color (e.g. red in Fig. 14). To obtain a figure of a certain shape (e.g. rectangle as in Fig. 14) we can also add points to the cells of the convex hull, or move these points to the required positions. In Fig. 14 the resulting shape is depicted with addition of new points, and Fig. 15 shows a figure with relocation of the common points of the polygons-hosts of the convex hull.

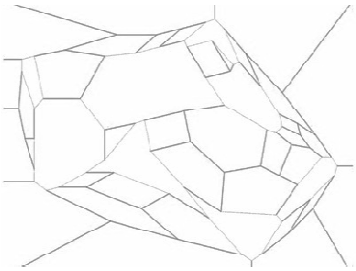


Fig. 14. Adding new points to the polygons

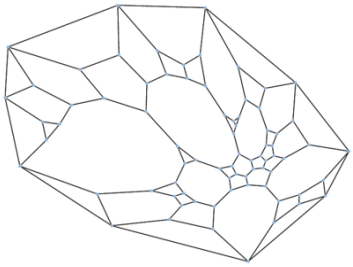


Fig. 15. Relocation of points of the polygons

Therefore, the graphical model allows to display hosts in the form of planes, and the links between the hosts - in the form of contact planes. The formal description of the algorithm to build the presented graphical model can be represented as the following pseudo code:

```
// step 1 - making the convex hull of planar graph
graph = getPlanarGraph(adjacencyMatrix)
graph = addConvexHullToGraph(graph)
//step 2 - making the constrained Delaunay triangulation
listOfTriangles = makeConstrainedDelaunayTriangulation(graph)
listOfPolygons
//step 3 - making cells based on triangulation
for each dot from graph {
       listOfTrianglesThatContainsDot = getTriangles(dot,
listOfTriangles);
       polygon
       for each triangle from listOfTrianglesThatContainsDot{
              triangleCenter = getCenterOfTriangle(triangle)
              addPontToPolygon(polygon, triangleCenter)
       }}
//step 4 - making separators
for each polygon from listOfPolygons{
       for each line from polygon{
              connectedPolygon = getConnectedPolygon(line, polygon)
              connectionIsExist = isConnectionExist(polygon,
connectedPolygon, graph)
              if connectionIsExist == true{
                     color = grey
              }else{
                     color = red }
              setLineColor(line, color)
       }}
// the end of algorithm is drawing of cells
draw(listOfPolygons)
```

It should be noted that from the mathematical point of view [18], the proposed model is not a Voronoi diagram, despite the visual similarity, since the Voronoi diagram has the different mathematical meaning, and existing algorithms for its construction do not allow to visualize the computer network topology when constructing the chart based on vertices of the graph.

4 Examples of application of the proposed graphical model

The proposed graphical model can be used to visualize the security parameters of a computer network or to analyze the behavior of the attacker. It may be required when informing the operator of SIEM system about the threats of a security breach or by visual analysis of computer network security. Let us consider examples of application of the proposed graphical model in the framework of implementation of the visualization system in more details.

4.1 Description of the visualisation system

To analyze the security of a computer network the visualization system is developed, which supports the display of the computer network using both the classical methods of visualization, such as graphs, graphs augmented with glyphs, treemaps and matrices and the graphical model, proposed in this paper. The example of dashboard of the developed visualization system is depicted in Fig. 16.

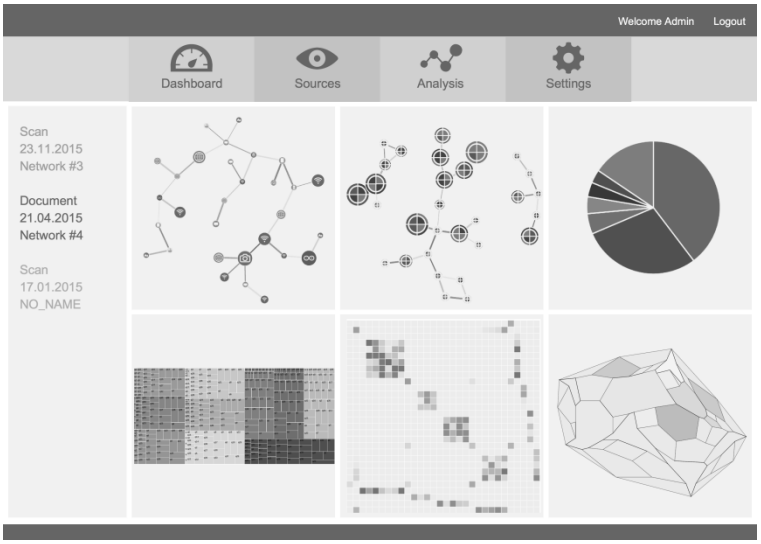


Fig. 16. The dashboard of the developed visualization system

The system includes the ability to manage data sources, aggregation and correlation of data collected from sources and tools for visual analytics of computer network security. Tabs to navigate to the relevant controls are at the top of the dashboard in Fig. 16.

In the left part of the dashboard in Fig. 16 there is enumeration of representations of computer networks formed on the basis of data from different sources.

The central part of the dashboard is the implementation of the multiple view concept, when the data is displayed in different views: as graph; as the graph augmented with glyphs; charts and diagrams; treemaps; matrices and as graphical model, formed as the analogue of Voronoi diagrams, presented in this paper.

In various usage scenarios, the user selects the graphical model, which is capable to visualize data the user needs at the moment most efficiently. In the following subsections the examples of scenarios of using the developed graphical model in comparison with the graphs will be presented.

4.2 Description of the source data

Data of a computer network (Fig. 17), which consists of 9 segments, will be used as source data for visualization.

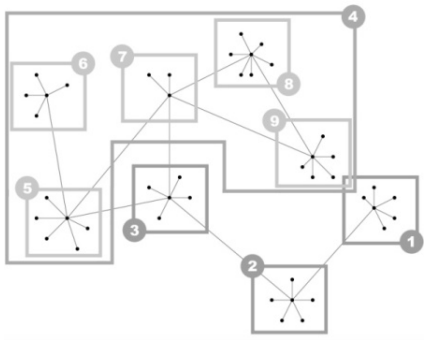


Fig. 17. Segments of a computer network

The segment, which consists of external users who have remote connection to the computer network via the Internet, is represented in block 1. Block 2 displays the web server for remote connectivity, as well as hosts needed to operate the web server. Block 3 represents the security system located between the web server and the demilitarized zone (DMZ). The demilitarized zone is shown by block 4 and includes segments of the internal network of the company. Block 5 and block 9 are the computers of the internal network users. Block 6 displays the devices connected to the network through Wi-Fi connection. Block 7 is the server of storing and processing data. Block 8 corresponds to the virtualization server, together with the virtual machines placed on it.

4.3 Example 1. Visualization of the state of computer network security

To visualize the security of a computer network for each host the indicators of protection from attacks and possible damage in case of compromise of this host are calculated. In the proposed graphical model the security level against attacks can be represented in the form of a polygon color - hosts that have passed the threshold are in red color (in Fig. 18 – dark grey), and possible damage is shown as the size of the polygon (area of polygon). We shall also consider the example of visualization based on

the graph, where the security is represented by vertex color, and the possible damage - as the radius of the vertex. Let us consider two variants.

In the first case (Fig. 18) the vulnerable hosts are red (dark grey in Fig. 18) are strongly scattered. Almost every network segment has vulnerable hosts. Despite the fact that damage at their compromise a small, scatteredness gives greater variability of actions for the attacker due to the presence of many potential hosts from which attack may occur.

Fig. 19 shows the corresponding visualization based on the graph, where the value of the possible damage in case of compromise is outlined by radius of vertex, and the presence of vulnerability is marked in red color (in Fig. 19 light gray, as they are almost invisible, they are indicated by arrows). It is obvious that the proposed graphical model allows us to more quickly identify vulnerable hosts and produce visual analysis of the damage done when they are compromised.

In another case (Fig. 20), it is clear that virtualization platform is under vulnerability, and therefore all machines, located on it, as well as some computers of internal users are also under vulnerability. Corresponding visualization based on the graph is shown in Fig. 21. As in the previous case, visual analysis of potential damage and identifying the vulnerable segment are more efficient when using the proposed model.

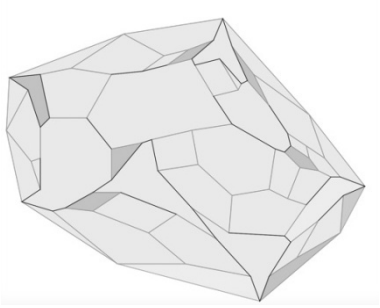


Fig. 18. Scattered unprotected hosts are presented on the basis of the proposed model

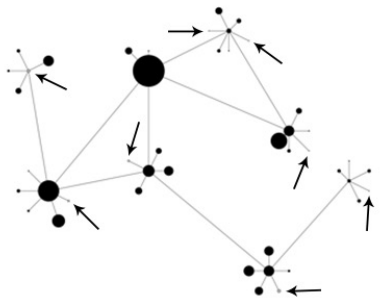


Fig. 19. Scattered unprotected hosts are presented on the basis of the graph

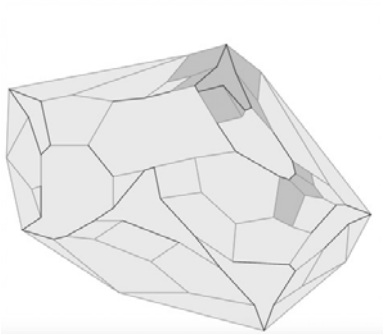


Fig. 20. Two segments of the network with unprotected hosts, presented on the basis of the proposed model

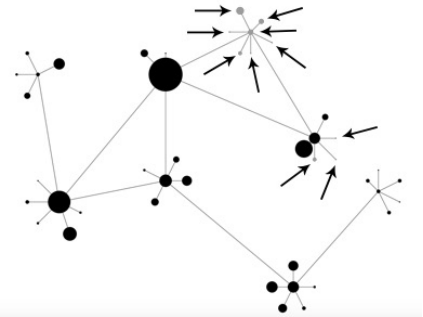


Fig. 21. Two segments of the network with unprotected hosts, presented on the basis of the graph

4.4 Example 2. Visualization of the attack route

Visualization of the route of the attack (Fig. 22) may be noted as another example of using the proposed graphical model.

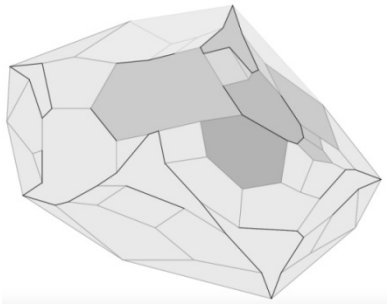


Fig. 22. Visualization of the attack route

The host from which an attacker carries out an attack is one of the computers of internal users. This host can be denoted in blue (dark gray in Fig. 22). All hosts, to which the actions of the attacker were recorded, can be also displayed in blue, however with different degree of transparency, which will depend on the intensity of the actions of the attacker. Thus it is possible to analyze which segment of the network is affected by the attack and if the attacker compromised the most important hosts or not. One can also estimate how close the attacker came to certain hosts for subsequent selection of protection strategies. In Fig. 22 it is seen that in such a scenario only some elements of the network will be affected (virtualization platform and servers for storing and processing data), however the potential damage from compromise is unreasonably great.

5 Evaluation of the proposed graphical model

In order to evaluate the proposed graphical model two groups of indicators were identified – performance indicators and functionality indicators.

Functionality indicators define the set of scenarios that the proposed graphical model is able to visualize, such as network connectivity, network size, abilities to display topology, possibility to display the parameters of hosts and links between hosts. Performance indicators define the efficiency of user's perception of information and ease of working with data, for example: the efficiency of indicators perception, efficiency of finding a way to attack, efficiency of analysis of network segments.

The evaluation of the proposed graphical model was carried out in comparison with the visualization of a computer network as a graph.

The evaluation was performed using a survey of experts. All experts note that in a scenario, when the computer network is a non-planar graph, application of the proposed graphical model is impossible, which is the main drawback of the proposed graphical model. In some scenarios, when non-planar graphs are rarely used or not used at all, the proposed graphical model is by an order more efficient. It is also noted

that the proposed graphical model is not efficient in visualization of small computer networks, and to visualize them it is more efficient to use matrices and graphs.

Thus, the graphical model is most appropriate to be used when rendering a medium to large scale computer networks. High computational complexity of the algorithm is also a disadvantage, however this can be shortened by using a client-server architecture, when only coordinates of the cells will be transferred to the thin client.

As a whole experts agree in opinion that the proposed graphical model has more options for visualizing metrics and network segments in comparison with the classical methods of visualization that are based on graphs, matrices and treemaps.

Thus, the proposed graphical model is an alternative to visualization of computer networks in the form of graphs, treemaps and matrices. The use of human spatial perception (the location of the cell-computers relative to each other) and the absence of necessity of edges provides a number of advantages at the cognitive level of perception of visualization. On the other hand, a disadvantage of the presented graphical model is that it can help to visualize exclusively planar graphs, which reduces the usage scope. However, in some scenarios, when graphs are not planar, are rarely used or not used at all, the proposed graphical model is by order more efficient. It is also worth noting that the use of the proposed graphical model is applicable for visualization of any object that can be represented as a planar graph.

It is supposed that the proposed graphical model can improve the efficiency of visual analytics using the graphical models to visualize the computer networks within the multiple view concept in SIEM systems.

6 Discussion

Based on the evaluation of the proposed graphical model a comparative table (tab. 1) was built with three other graphical models - graphs, treemaps and matrices. It should be noted that graphical models are considered in the minimum version, i.e. without additions and excluding the presentation of information in the form of text and signatures. The table shows the display capabilities and the number of simultaneously displayed parameters of security of network hosts, network connections, possibility of extension of graphical models for visualization in 3D space, possibility of visualization of topological parameters and possibility of display of networks of different topological types. The cells contain evaluations of the efficiency of parameters display perception in one way or another. Four estimations are outlined: (1) does not support - graphical model does not support this method of visualization; (2) supports - graphical model supports this method of visualization, but with restrictions; (3) good - graphical model supports this method of visualization; (4) fine - graphical model supports this method of visualization, the efficiency of the perception is high and the user easily analyzes the information.

Graphs are not restricted in display of size and color of vertices, size and color of connections, clustering, and can also display any topological types. Graphs have a limit on the transparency of the elements, as with the transparency of less than 30% the item will be hard to read. Graphs also can display hosts and connections by a limited set of geometric shapes to be visualized in 3D, provided that clustering of the vertices is done. The only thing that graphs do not support is display of nesting.

Table 1. Comparison of the possibilities of graphs, treemaps, matrices, and the proposed model

		Graphs	Treemaps	Matrices	The proposed model
Hosts parameters display	size	good	fine	does not support	fine
	color	good	fine	does not support	fine
	transparency	supports	good	does not support	good
Display of parameters of links	size	good	does not support	does not support	good
	color	good	does not support	good	good
	transparency	supports	does not support	supports	good
	shape	supports	does not support	supports	supports
Possibilities of extension for display in 3D		supports	does not support	supports	supports
Display of topology parameters	hosts clasterization	good	good	fine	good
	incapsulation of hosts (nestness)	does not support	fine	does not support	fine
Display of different topological types	hierarchical	good	fine	good	fine
	planar	good	does not support	good	fine
	non-planar	good	does not support	fine	does not support

Treemaps do well with display of hosts using the size and color of planes. It is the use of planes that allows the user to efficiently analyze information. With it, treemaps have no restrictions in the display of hosts with transparency, because even at minimal transparency the outlines of the plane are kept, and the user cannot miss it, as it would be in the case of graphs. However, treemaps cannot display the parameters of the links and cannot be displayed in 3D. But at the same time they do not have restrictions on clustering of hosts and perfectly display nesting, as it is the basis of the very concept of the treemaps. However, because of the nesting treemaps allow to display exclusively hierarchical networks.

Matrices cannot display hosts, but allow to visualize them when using the cells' colors and their transparency. Matrices cannot operate with cells' sizes. Transparency of the matrix cells have the same restriction as graphs (at least 30% transparency), but they have no restrictions in colors. Cells themselves can also be represented as a limited set of geometric shapes and can be displayed in 3D. Matrices cannot visualize nesting of the hosts, however, with the help of links they can efficiently display the network's clusters that will be represented in the form of planes (Fig. 1, right part). Matrices can display without limitations any topological types, however they are most efficient in finding clusters in complex non-planar networks.

The proposed graphical model uses the size, color and opacity of the cell for displaying of parameters of the host, as in the trees of maps, so their estimations coincide. To display the links' parameters it uses size, color and transparency of edges of cells. Edges are visualized as lines like in graphs, therefore their estimations are also identical, except for the transparency – in the proposed model, even at zero transparency the contour of lines is preserved. The proposed graphical model can be repre-

sented in the form of a 3D polyhedron, where the edges will correspond to the cells. The graphical model also has the capability of clustering, as shown in Fig. 18, and through the use of planes, inside which one can place similar planes, it supports nesting. The proposed graphical model also supports the display of hierarchical and planar networks, but does not support non-planar network.

Thus, we can see that the proposed model in some cases (in the hierarchical and planar network topologies) can provide an alternative to graphs, treemaps or matrices, or be used as a supplement to graphical models within the multiple view concept, based on which the dashboard of security systems are built.

The development of the presented graphical model is been continued. At the moment the development of the algorithm of the polymorphism of cells is performed, in order to be able to change the size and shape of the cells without violating the topology. The development of the algorithm of display of proposed graphical model in 3D is carried out, this can be achieved by imposing points of the graphical model on the sphere, and then to draw cross-sections of the sphere at the points that correspond to the cells.

Another direction for future research is to analyze the possibility and efficiency of using the proposed graphical model for visualization of processes associated with information security, but which are not associated with computer networks. Any process that is currently visualized using planar graphs (some examples are displayed in section 2), can be visualized with the proposed graphical model. Thus, despite the fact that in the implemented system the proposed model is used to find vulnerabilities, risk assessment and other parameters of a computer network, the scope and possibility of application to ensure information security are much wider.

7 Conclusion

In this paper the analysis of existing visualization methods was performed and the new graphical model based on the analogue of Voronoi diagrams was presented. We demonstrated that the proposed graphical model of visualization of computer networks allows in some cases to display data more efficiently, compared to already existing graphical models. The developed visualization system was presented used to analyze the security of computer networks, and examples of visual analysis of the computer network state were provided. The estimation of the proposed model for visualizing the security parameters of computer network was done, and the comparison of its efficiency with graphs, matrices and treemaps was performed. Directions of future research based on the use of the proposed graphical model were presented, in particular the development of algorithms for polymorphism and nesting of cells, display in 3D and using the proposed model for visual analytics of processes and objects, which were previously represented as graphs.

Acknowledgements.

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 SPIIRAS.

References

1. Wang, M., Woodruff, A., Kuchinsky, A.: Guidelines for Using Multiple Views in Information Visualization. J. Advanced Visual Interfaces, pp. 110-119 (2000)
2. Shi, L.: Scalable network traffic visualization using compressed graphs. In: Shi, L., Liao, Q., Sun, X., Chen, Y., Lin, C. Proc. of IEEE International Conference on Big Data (BigData 2013), Santa Clara, CA (2013)
3. Tufte, E.: Visual Explanations. Graphics Press, Cheshire, Connecticut (1997)
4. Klyshinskij, J., Rysakov, S., Shihov, A.: Review of the methods of multidimensional data visualization. J. New information technologies in automated systems, pp. 519–530 (2014)
5. Marty, R.: Applied Security Visualization. Addison Wesley Professional (2009)
6. Kwan-Liu, M.: Cyber Security Through Visualization. In: Asia Pacific Symposium on Information Visualisation, Tokyo, Japan (2006)
7. Noel, S., Jajodia, S.: Understanding Complex Network Attack Graphs through Clustered Adjacency Matrices. In: 21st Annual Computer Security Applications Conference (ACSAC'05), IEEE Computer Society (2005)
8. Lau, S.: The spinning cube of potential doom. In: Communications of the ACM, vol. 47(6), pp. 24-26 (2004)
9. Harrison, L., Spahn, R., Iannacone, M., Downing, E., Goodall, J.: Nessus Vulnerability Visualization for the Web. In: VizSec '12, Seattle, WA, USA (2012)
10. Williams, L., Lippmann, R., Ingols, K.: GARNET: A Graphical Attack Graph and Reachability Network Evaluation Tool. In: 5th International Workshop on Visualisation for Computer Security (VizSec'08), LNCS, Vol.5210, Springer-Verlag (2008)
11. McGuffin, M.: Simple Algorithms for Network Visualization: A Tutorial. J. Tsinghua science and technology, Vol. 17, No. 4 (2012)
12. Novikova, E., Kotenko, I.: Analytical Visualization Techniques for Security Information and Event Management. In: 21th Euromicro International Conference on Parallel, Distributed and network-based Processing (PDP 2013), Belfast (2013)
13. Montemayor, J., Freeman, A., Gersh, J., Llanso, T., Patrone, D.: Information Visualisation for Rule-based Resource Access Control. In: International Symposium on Usable Privacy and Security (SOUPS) (2006)
14. Glatz, E.: Visualizing Big Network Traffic Data using Frequent Pattern Mining and Hypergraphs. In: Glatz, E., Mavromatidis, S., Ager, B., Dimitropoulos X. Proc. of the First IMC Workshop on Internet Visualization (WIV 2012), Boston, MA, USA (2012)
15. Mansmann, F.: Visual Support for Analyzing Network Traffic and Intrusion Detection Events using TreeMap and Graph Representations. In: Mansmann, F., Fischer, F., Keim, D.A., North, S.C. Proc. of the Symposium on Computer Human Interaction for the Management of Information Technology (CHiMiT'09), No. 3, pp. 19-28 (2009)
16. Kotenko, I., Chechulin, A.: Common Framework for Attack Modeling and Security Evaluation in SIEM Systems. In: 2012 IEEE International Conference on Green Computing and Communications, Conference on Internet of Things, and Conference on Cyber, Physical and Social Computing, Besançon, France (2012)
17. Kolomeec, M., Chechulin, A., Kotenko, I.: Methodological Primitives for Phased Construction of Data Visualization Models. J. Internet Services and Information Security (JISIS), Vol.5, No.4, November, pp. 60-84 (2015)
18. Aurenhammer, F., Klein, R., Lee, D.: Voronoi Diagrams and Delaunay Triangulations. World Scientific Publishing Co. (2013)