A visual analytics approach for the cyber forensics based on different views of the network traffic

Igor Kotenko[1,2]*, Maxim Kolomeets[1,2], Andrey Chechulin[1,2], and Yannick Chevalier[2]
[1]*Laboratory of Computer Security Problems, St. Petersburg Institute for Informatics and Automation (SPIIRAS), 39, 14-th Liniya, Saint-Petersburg, 199178, Russia*
{kotenko,kolomeec,chechulin}@comsec.spb.ru
[2]*Saint-Petersburg ITMO University, 49, Kronverksky Prospect, St. Petersburg, Russia*
yannick.chevalier@gmail.com

Abstract

Network forensics is based on the analysis of network traffic. Traffic analysis is a routine procedure, but it allows one to not only identify the cause of the security breach, but also step by step to recreate the whole picture of what happened. To analyze the traffic, investigators usually use Wireshark, a software that has the graphical interface and has greater capabilities for sorting and filtering packets. But even with it, packet analysis takes a lot of time. In this paper, we propose an approach for cyber forensics based on different views on the network traffic. Using this approach, it is possible to significantly improve the efficiency of forensic scientists, including the rapid localization of anomalies and, importantly, the creation of easily understandable graphical proofs and histories of computer attacks. The example of the investigation of the attack SSL-strip is a way to classify different views (slices) of traffic and a scheme for using for these slices different models of visualization. Also provides an assessment and recommendations for the application of visual analytics methods.

Keywords: network forensics, visual analytics, data visualization, traffic analysis, cyber-attack investigation.

1 Introduction

Models and methods of network forensics, unlike models and methods of access control, cryptography and other methods of protecting information, are not sufficiently developed. Companies primarily want to prevent an attack, to assess the risks and identify their weaknesses. Network forensics differs in that it starts mainly after the attack, when the company already considers the losses.

In forensics, as a branch of information security, there are as many areas as there are types of computer attacks. Its arsenal includes examination of hard drive, RAM dumps for malware actions, analysis of DLP-data for finding insiders and, of course, analysis of network traffic. The criminalists use any means available to them to identify the cause of the security breach and draw up a solid evidence base.

In most cases, the criminalist is confronted with the analysis of network traffic. Traffic analysis is a routine job, but its analysis allows not only to identify the cause of the security breach, but also to reconstruct a step-by-step whole picture of what happened: to identify the potential attacker (or at least the start point), recreate the sequence of actions, and the opportunities for the affected assets to identify potential intentions and develop recommendations to close the exploited vulnerabilities.

For analysis of traffic criminalists usually use Wireshark [1] – software with a graphical interface that allows one to overview network packets and has extensive sorting and filtering capabilities. Traffic analysis is a time-consuming task - the criminalist has to browse the packages hour after hour in the hope

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), 9:2 (June 2018), pp. 57-73
*Corresponding author: Tel: +7(812) 328-71-81, Web: http://www.comsec.spb.ru/

of identifying and linking the anomalies. In order to expand the capabilities of forensic scientists and simplify their work, it is proposed to equip them with visual analytics tools.

Visual analytics has proven itself in the simulation of computer attacks and monitoring of network security [2, 3, 4]. It is difficult to imagine a modern security system that does not even have the basic tools for visualizing the state. These opportunities can be transferred to network forensic science. Moreover, network traffic is a well-structured set. It is easy to parse and process. The main difficulty is that there is a lot of data and the investigator has to create corresponding representations of data-slices that can contain the necessary information.

In this paper, we propose an approach to the use of methods and means of visual analytics for network forensics, a method for classifying different representations (data-slices) of traffic, and a scheme for using various visualization models for these slices. Using this approach, it is possible to significantly improve the efficiency of forensic scientists, including the rapid localization of anomalies and, importantly, the creation of easily understandable graphical proofs and histories of computer attacks. The paper suggests models and methods of visual analytics, applicable for network criminalistics.

In existing works [5, 6, 7], visual analytics is used to solve forensic problems by applying one or more visualization models. To solve not specific, but common tasks, an approach is needed that will allow one to determine what kind of visualization model should be used in this or that situation. The novelty of this scientific work is the usage of data sets classification for traffic slices visualization. Classification allows one to determine the type and structure of data and limit the number of suitable imaging models. The contribution of this work is a visualization technique that allows one to visualize anomalies in traffic sections from data generation and to their drawing in a graphical form on the screen of the investigator. The technique also allows one to determine which different visualization models for one data set can be used, depending on the situation. It is likely that using one visualization model, the anomalies will not be as well distinguished as using the other.

This paper is an extended version of the paper [8]. The main extensions are a more comprehensive analysis of visual analytics techniques, detailed representations of visual analytics technique suggested and experiments.

The work consists of four parts. The first section is Introduction. The second section deals with relevant imaging models. In addition, since it is difficult to visualize all the necessary details using a single model, a classification of the various representations of the traffic data of the slices are given. Classification of slices is necessary to determine the structure of the set and to select a suitable model of visualization. In the third section, a specific example of a computer crime investigation based on the SSL-strip attack demonstrates the proposed method for classifying different traffic slices and a scheme for using different models of visualization for these slices. The fourth section discusses the effectiveness of visual analytics and its possible role in the criminalist's arsenal.

2 Visual investigation technique

The visualization technique can be used both for visual analysis directly in the investigation of incidents, and for forming a visual representation of situation for the report. At first, the investigator must create slices of data from the traffic which may contain information about anomalies. For each slice, the investigator should define its type in order to understand which visualization model can render it. The next step is to classify the slices in order to determine what visualization models are suitable for a particular set and to limit the number of models. After choosing a model, investigator needs to bring these slices to a format suitable for visualization in accordance with the type of slice and visualize them. Thus, the visualization technique consists of two stages: creating and classifying the analyzed data slice, selecting an appropriate model or several visualization models to bring the slice to the format required for

visualization and application of the visualization model to the data.

2.1 Data-slices classification

Visualization models are used as templates for displaying different data structures [9]. Knowing the models allows one to choose the appropriate visualization technique, depending on the situation. Models can be divided into two large groups.

The first group is numerical data models. Each value in these data is represented by a measure either quantitative (for example, the number of events) or categorical (for example, the type of event). Such data are the most common and are usually presented in the form of tables. They are easy to manage, easy to store, and intuitive. To work with a small amount of data, you can use MS Excel, Numbers or Libre Office, which also have simple visualization models built in, such as pie charts, bar charts and line charts. It should be noted that as the data grows, the complexity of their management grows, which can cause unintended errors [10]. Therefore, to avoid errors, large sets of numerical data are best stored in relational databases.

The second group is not numerical data models. Unlike numerical data, not numerical have one important attribute they contain links. These links can be expressed as a relationship, nesting or dependences. The presence of relationships makes it difficult to store not numerical data in tables. Therefore, they prefer to write to object-oriented structures, JSON files or non-relational databases (for example, Mongo DB). It is not so often necessary to work with not numerical data, but they represent the greatest problem in the visual analysis for the analyst-criminalist. Let us consider them in more detail.

Not numerical data can always be represented as a graph. If the numerical data can be represented practically using any numerical data visualization model, the choice of the model of visualization not numerical data depends on the type of the topology of the graph:

- tree – hierarchical data, for example, attack tree;
- planar – data that can be drawn on a plane without intersecting edges, for example, a graph of rooms;
- semi-structured – an unstructured data graph, in the subgraph of which you can select a specific structure (Hairball, Tree, etc.), for example, individual network segments will almost certainly have a tree topology;
- unstructured – data in which it is difficult to single out any structure, for example, the social graph;
- combined – data with different types of links, each type can be represented separately by a different type of topology, for example, the physical infrastructure of the network and the transfer of traffic between hosts.

It's important to pay attention to the order of classification [11]. A tree is always a planar graph. The planar graph is always semi-structured, and so on from the list. The reverse way of the list is different. For example, only some of the planar graphs are trees, and so on. This indicates an important feature: visualization models for the relevant structure can always visualize structures that are higher in the list. At the same time, they cannot visualize the structures that are below.

2.2 Choosing of data visualization model

Having determined the structure of the data, it is easy enough to choose the appropriate visualization model. Models of visualization can display a slice of only a specific format. For example, it is impossible

to represent the topology of a computer network using a linear chart. In order to determine the models that are suitable for a particular section, we have compiled a correspondence table (Table 1). This table includes the basic models of visualization. This section shows some of the most different models of visualization: Bar Chart (Figure 3), Parallel Coordinates (Figure 4), Trilinear Coordinates (Figure 5), Scatter Plot (Figure 6), Circle Packing (Figure 7), TreeMaps (Figure 8), Voronoi Maps (Figure 9) and Chord Diagram (Figure 10). Let's consider their features.

Bar Chart [3] allows one to compare the values between each other. As a rule, one axe represents categories, and the second - quantitative data. Bar Chart visualize numerical data. Figure 3 shows the top port denies for the view of "bad port" activity [4]. It's important to note, that bar chart is refers to "accurate models" [12]. In accurate models is easy to compare values in comparison with "not accurate" models. Try to compare values "A" and "B" in "not accurate" PieChart in Figure 1 and the same values in accurate BarChart in Figure 2. At the same time, if your data is not accurate, for example some probabilities, usage of not accurate models helps to hide small difference, which may effect on decision making, although they should not.

Type	**Visualization Model**
Numerical	Pie Chart, Line Chart, Bar Chart, Scatter Plot, Parallel Coordinates, Triangle Coordinates
Tree	Circle Packing, TreeMaps, Voronoi TreeMaps, Radial Trees
Planar	Voronoi Maps, Graph of the Rooms
Semi-structured	Chord Diagram
Not-structured	Graph, Matrix
Combined	Different combinations of models [11]

Table 1: Data-slices classification

Parallel coordinates [13] are suitable for multidimensional data. Each vertical line is the axis of one of the metrics. It does not have to be quantitative. The axis can be made qualitative for some relationships. For example, the criticality levels "High", "Medium" and "Low" can be arranged one after the other so that it is intuitively clear that the criticality is in the upper part of the "High" axis, in the medium "Medium", and in the lower "Low". Having arranged the axes, a broken line is drawn for each row of the numerical set, with a fracture when crossing with the axis. In Figure 4 shows a network scan [14].

Triangular coordinates [15] allows one to visualize the three metrics expressed as a percentage. For example, Fig. 5 shows the accuracy test of color detection sensor – dots color is the actual color of the object that was detected by the sensor, when the position of each dot represents the RGB ratio that was generated by sensor. Typically, for such data, three-dimensional cubes are used, in which each dimension is an axis. In the case of relative data, only half of the cube will be used, since the values can individually be up to 100%, but not their sum. Also, the 3D cube is more difficult to control on the screen, it always has to be rotated. Triangular coordinates provide a good two-dimensional alternative.

The dispersion graph [3] allows us to judge the distribution of objects. For example, in Fig. 6 displays 21 days of firewall traffic [4]. A typical linear graph (top of the Figure 6) displays a large number of extremes. It is difficult to single out trends from it. The dispersion graph (bottom of the Figure 6) allows you to see trends and retains the ability to visualize extremes.

Circle Packing [16] is a simple visualization of trees. Using the size of the circles, the metric of the object is displayed. And all the descendants associated with the object are already inside this ball. Figure 7 shows the circle packing used in the CyberPetri system on the CDX 2016 [17].

Trees maps [18], unlike the Circle Packing, do not have unused space. They, like the circle packing,

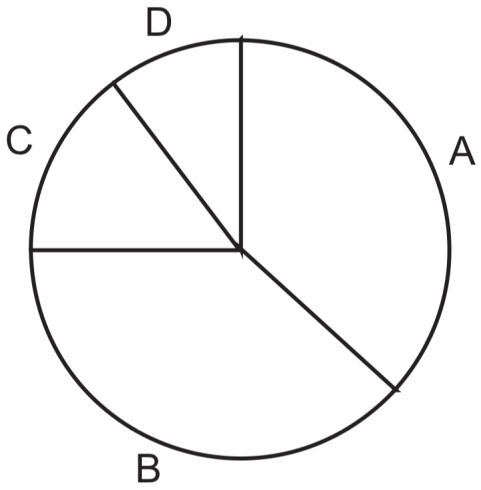

Figure 1: “not accurate” PieChart [12]

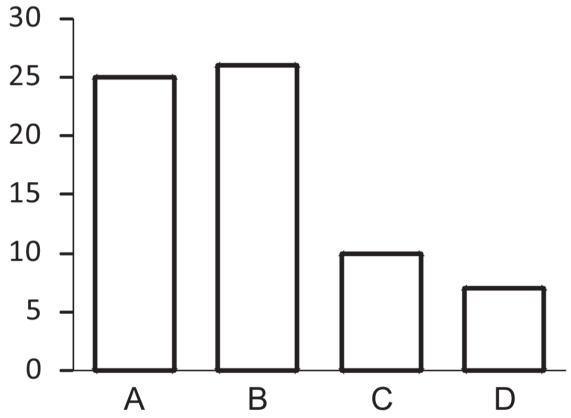

Figure 2: “accurate” BarChart [12]

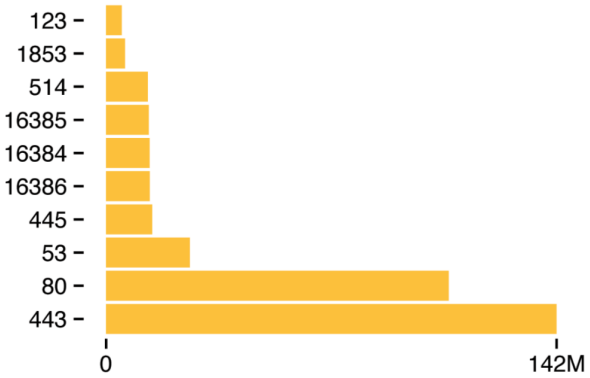

Figure 3: BarChart [3]

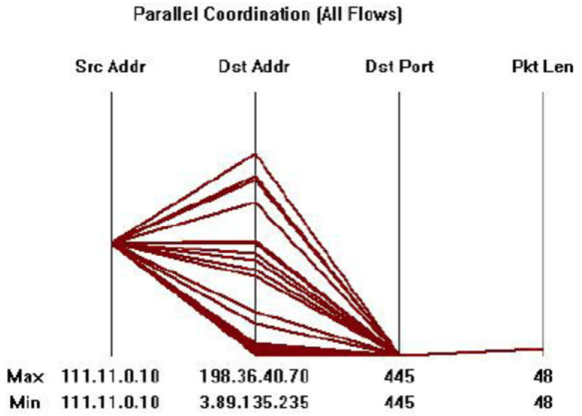

Figure 4: Parallel Coordinates [14]

visualize the connections by nesting, and using the size of rectangles the values of different metrics [19]. In Figure 8 [20] depicts the security report in the form TreeMaps: user-defined criticality vs. security level.

Voronoi’s maps [21] are suitable for visualization of planar data. Each cell is represented by an object, and the face between cells is a link between objects. Using the size and color of the cells, you can display metrics. In Figure 9 depicts the graph of the network and the Voronoi map constructed on its basis.

The Chord diagram [3] is a striking example of semi-structured data. It is suitable for visualizing objects in which one type of connection is expressed by a hierarchy, and the second is unstructured. In Figure 10, the physical hierarchy of the computer network is represented by an outer ring, and the logical links between individual hosts, in the form of curves inside the ring.

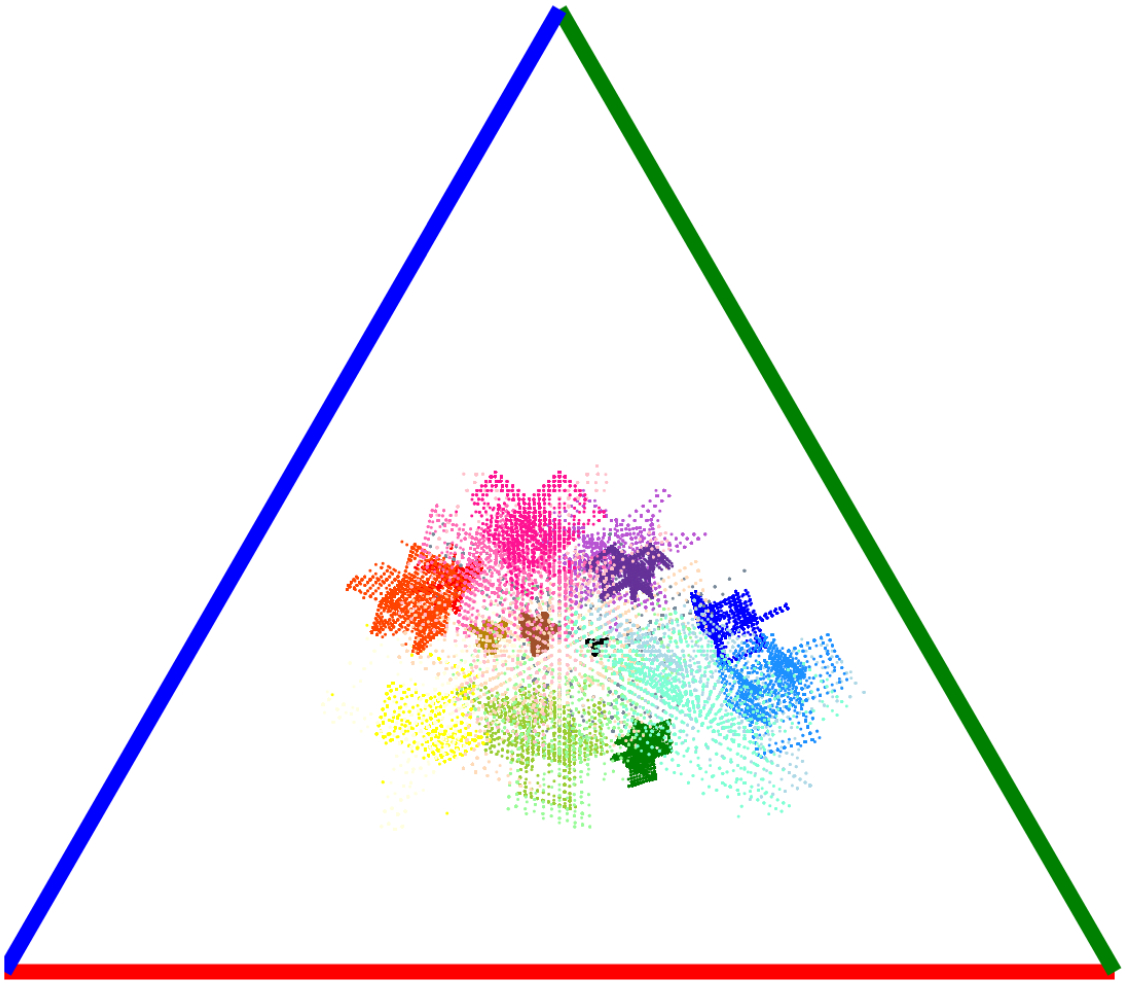

Figure 5: Trilinear model represents measurements of color sensor

3 Illustration of the Proposed Approach Application

As an illustration of the application of the proposed approach to the use of methods and means of visual analytics for network forensics, we will use the example of a security breach that was developed in the framework of the project "Educating the Next Generation Experts in Cyber Security: the new EU-recognized Master's program". Research Project of the European Community program TEMPUS No. 544455-TEMPUS-1-2013-1-SE-TEMPUS-JPCR. Let's imagine a way to classify different views (slices) of traffic and a scheme for using various models of visualization for these slices on the example of the investigation of the SSL-strip attack. This section consists of three parts: the legend of the attack, the description of the investigation using Wireshark and the description of the investigation using visual analytics.

3.1 Security Breach Story

There was a room with 4 persons Patrick, Lisa, John and Harry. It is becoming known that John received several spam messages from Lisa. Lisa said that she did not send these messages. You decide to take

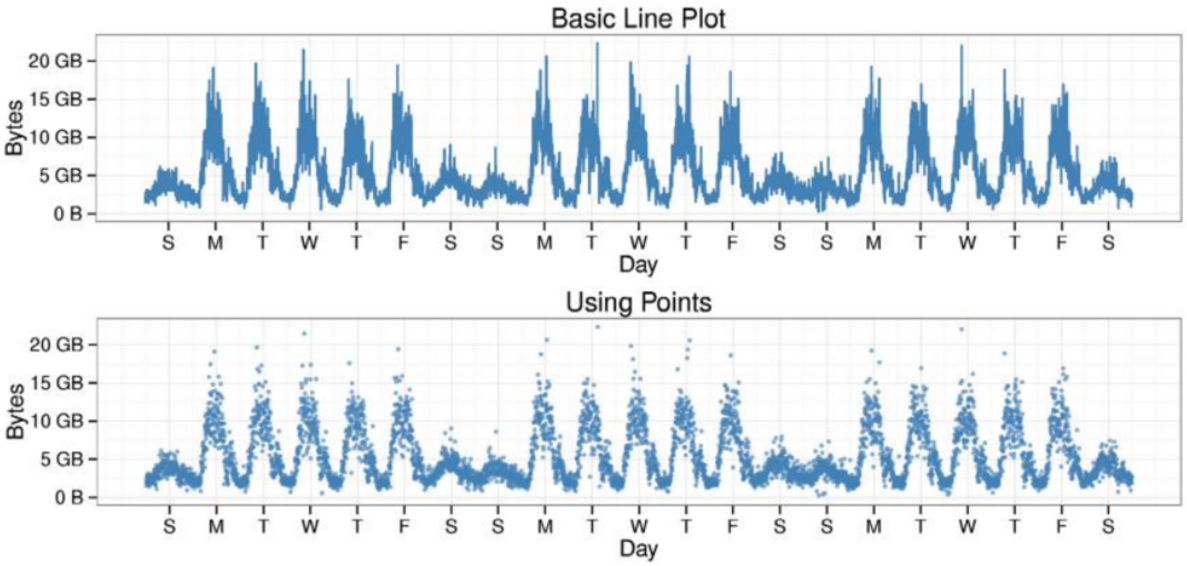

Figure 6: 21 days of traffic on line chart and scatter plot [3]

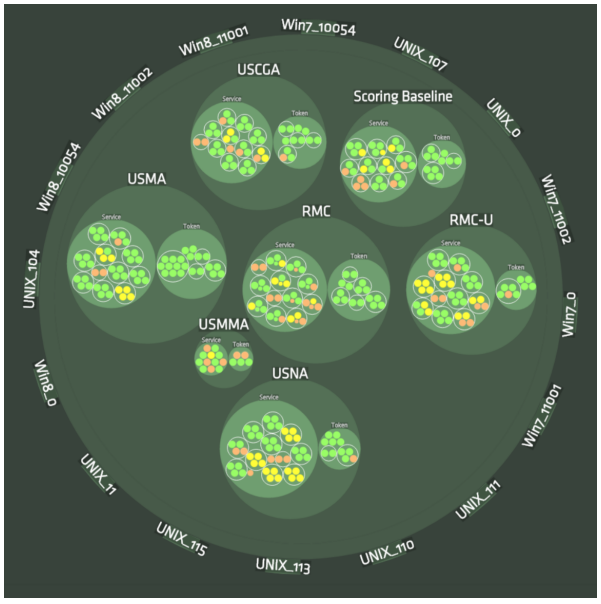

Figure 7: Circle Packing in CyberPetri [16]

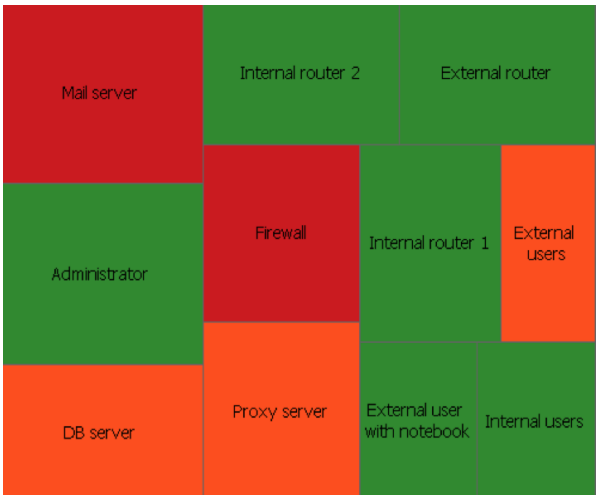

Figure 8: Security report based on TreeMaps [20]

up the issue in detail and examine network traffic. The system administrator provided data on the computer network (shown in Table 2) and recorded computer network traffic in the PCAP format. You can download PCAP-file and relative use-case description files via the link [22].

3.2 Basic Solving

This use-case is suggested to be solved by analyzing the PCAP-file using Wireshark. In total there are 200584 records in the file, and on average the problem can be solved in a few hours. In addition, the criminalist must draw up an account of the incident.

Thus, the criminalist faces two tasks: he must identify the suspects and must provide evidence against

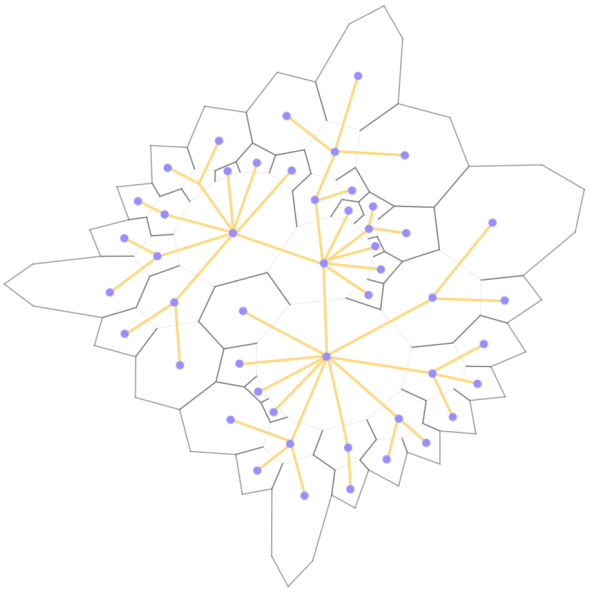

Figure 9: Voronoi Map and network graph

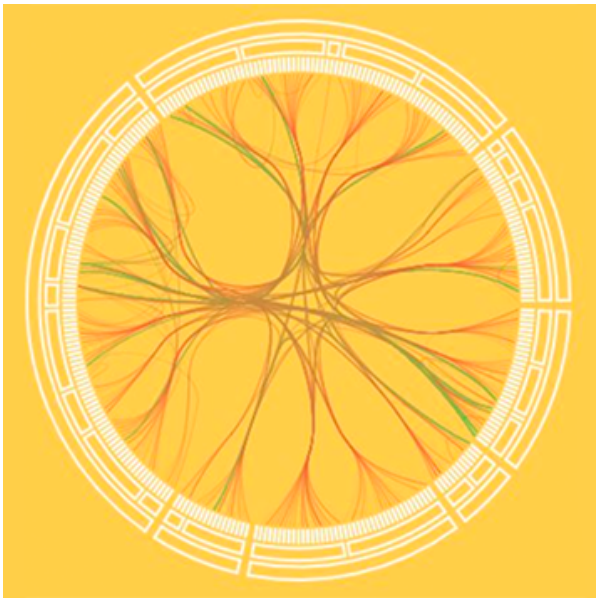

Figure 10: Chord Diagram

Name	OS	IP	MAC	Caption
Patric	Debian 8.2	192.168.238.128	00:0c:29:04:9c:a3	Know nothing
Losa	Ubuntu 15.10	192.168.238.129	00:0c:29:a4:63:4f	Sent spam
John	CentOS 7.1	192.168.238.130	00:0c:29:2f:be:8c	Received spam
Harry	Manjaro 15.9	192.168.238.131	00:0c:29:89:74:8b	Lover of video

Table 2: Network configuration

them. To solve these problems, it is not enough to identify a potential attacker – it is still necessary to provide proof to the person who will make the decision. Thus, the structuring of the detected traffic anomalies into a consistent and clear view also requires a certain amount of time.

For completeness, we give a brief scenario of the attack:

1. Patrick changes own IP address and MAC address;
2. Patrick runs ARP-Spoof and SSL-Strip for capturing the login and password;
3. Lisa enters her account to check mail messages and Patrick captures these confidential data;
4. Patrick uses Lisa's account for sending the spam messages to John;
5. Patrick restores his own previous network configuration and continues normal activity in the network;
6. John and Harry simply create the background of normal behavior in network traffic.

A typical solution of the problem consists of the following steps:

1. Investigator observes that the Lisa's password was transmitted in plaintext without encryption;
2. Criminalist finds an intermediate host with the MAC address 00: 0c: 29: bf: 22: 13 through which the packet is transmitted;

3. Such MAC address is absent in network; therefore, it can be concluded that there was a substitution of address;

4. The Investigator finds out a lot of replies from such host to Lisa's computer which the MAC address is 00: 0c: 29: a4: 63: 4f; such packets are send from the host which pretends to be a router with the IP address 192.168.238.2;

5. Also it becomes known that the attacker has the changed IP address 192.168.238.135 and uses the browser based on the AppleWebKit / 538.15 (KHTML, like Gecko);

6. Such browser is used by Patrick (192.168.138.128) and also, he has the information "how to change MAC address in Linux".

3.3 Visual Analysis Solving

To solve the test using visual analytics, we had made data slices from PCAP-traffic, which can contain traces of the attack. The first slice was formed for routing analysis. It was made from ARP packets and contained 2 fields: Source MAC and Source IP. The second slice was configured to analyze the network activity of each host. It was made on the basis of all packets and contained 4 fields: Source MAC, Packet Type, Packet Length, Packet Timestamp. The third file contained time windows with traffic. For each window, the percentage of packet types in the network was recorded. Let's look at the visualization of these slices.

Let's represent the visualization of these slices.

For the first slice, we choose the model of parallel coordinates (Figure 11). The left column represents the IP-address, and the right – MAC-address. If there is a line between the elements, it means that in the slice there is information about the packets with such a pair of IP and MAC addresses. Obvious feature that one can see is MAC with eight IP addresses (right column, second from the top, light-blue rectangle on Figure 11). This host is a router that broadcasts broadcast-packets, so these seven lines are a kind of data noise in the analyzed slice. If one does not consider the data from the router, then all MAC addresses have one IP, except one. Host with MAC address 00:0c:29:bf:22:13 (the right column, the first from bottom, the pink rectangle on Figure 11) has three IP-addresses. This behavior clearly stands out against the background of the behavior of other network hosts.

Let's look at the second slice. In it, we calculate the number of packets of the same type for each of the devices and represent it using the basic Bar Chart. For all types of packets except ARP (Figure 12) anomalies were not detected. When ARP-packets are displayed, the host with MAC-address 00:0c:29:bf:22:13 is allocated (the first from the bottom bar on Figure 12), which sent ARP packets more than the router (the router is the second from the top bar on Figure 12).

This ARP-anomaly can be detected in the third slice. For its visualization, we used the model of triangular coordinates (Figure 13). The slice was aggregated to three values: percentage of ARP traffic, TCP traffic and all other traffic. Every few seconds of observation are represented by the individual red dot. Most of the observations are on the lower edge (they did not have any ARP traffic). When ARP traffic appears on the network, the dots are slightly climbing. These two dots on the left edge are time intervals in traffic dump that have increased number of ARP packets (20% and 30% of ARP packets in one-time window). We can check the time intervals of these dots and look at them in Wireshark in details.

If one look at the PCAP-file with time packets (Figure 14), when these two points were fixed, traces of ARP-spoofing will be detected.

Let's go back again to the slice number 2 to consider who related to MAC-address 00:0c:29:bf:22:13. Let's represent each user in the form of separate Scatter Plot in which the Y-scale is the packet length,

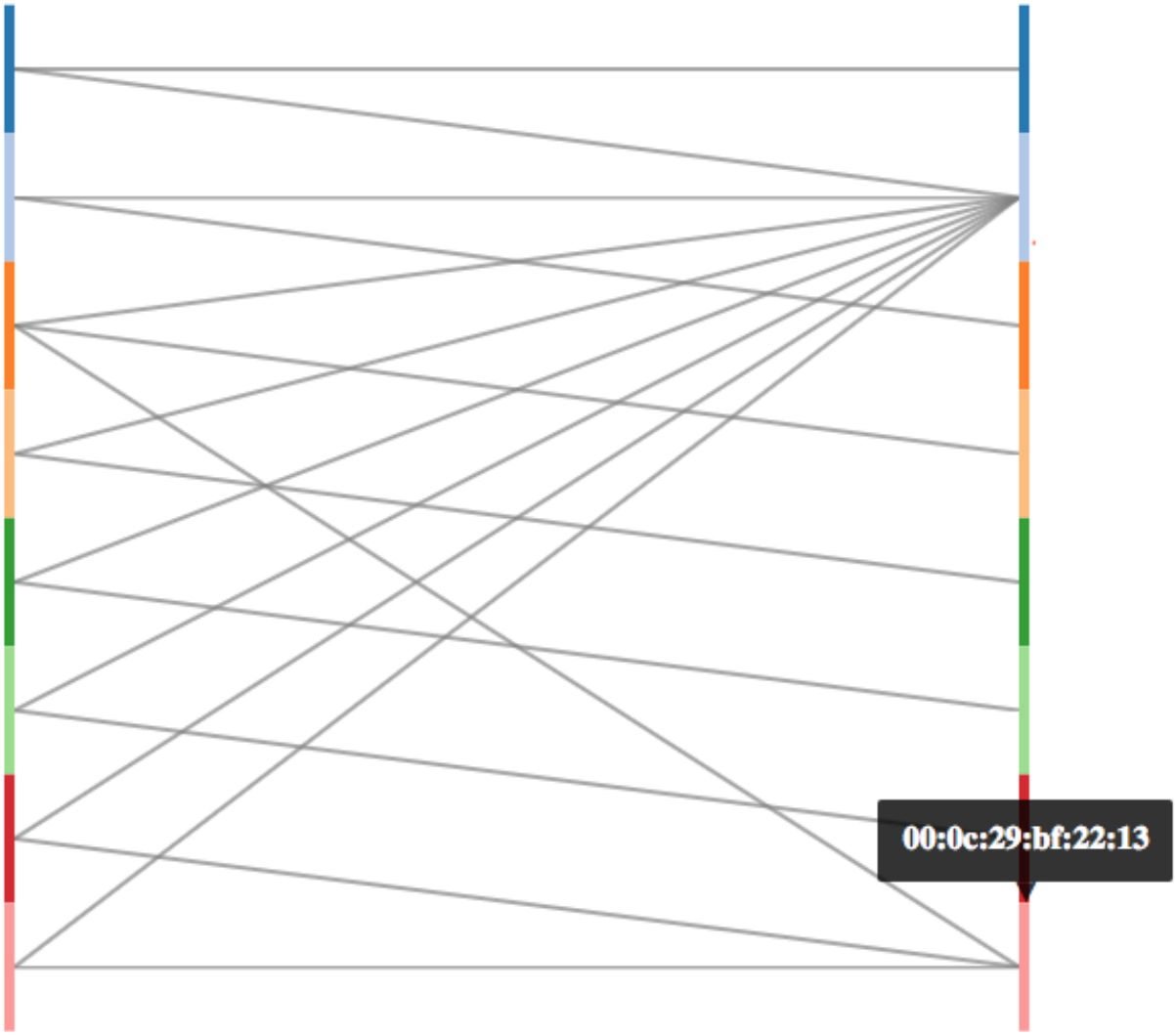

Figure 11: Parallel Coordinates represents IP-MAC pairs anomaly around host 00:0c:bf:22:13

the X-scale is the timeline of the packet, color is the type of packet, point - packet. In this graph (Figure 15), one can see some features of user activity. For example, consecutive vertical lines may indicate the download of files within a single session. Consecutive horizontal lines in which packets of the same length appear at regular intervals may indicate services. But on this plot, the obvious anomaly is that Patrick (the fourth from the bottom Scatter Plot) left the network during the attack. The attacker himself 00:0c:29:bf:22:13 (the first bottom Scatter Plot) was active only at the time when the Patrick was inactive (highlighted in red rectangles).

Consider this discover in a simpler form. Figure 16 shows Scatter Plots without packet lengths in the form of horizontal columns. On it, the sequence of actions is more obvious: the Patrick leaves the network $\rightarrow$ the attacker appears $\rightarrow$ the attacker leaves the network $\rightarrow$ the Patrick appears.

3.4 Discussion

The presented use-case is used in training the network forensics of master students at the ITMO University. It is assumed that a student familiar with the basic tools of Wireshark for 3 hours (2 laboratory

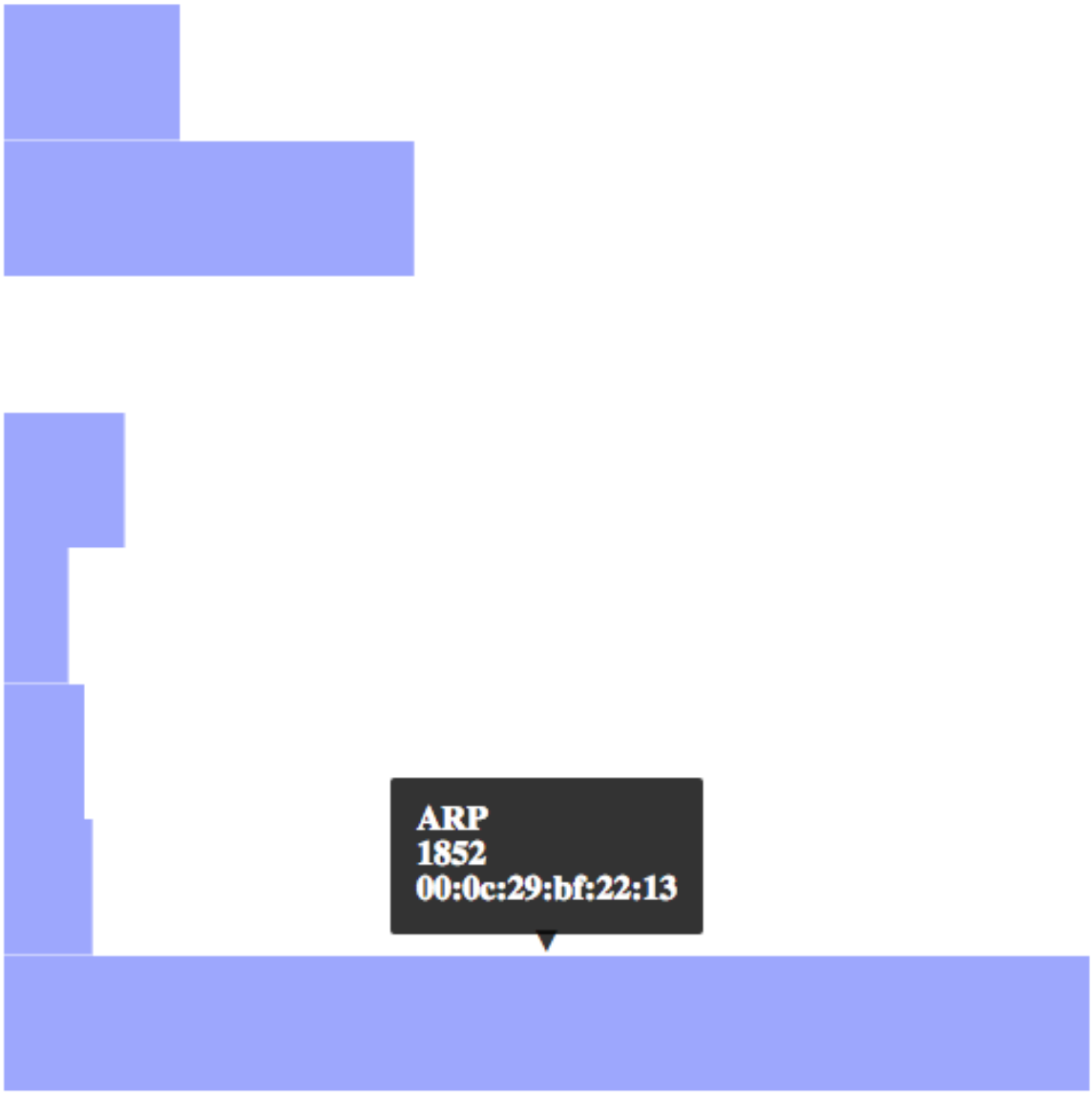

Figure 12: BarChart represents ARP traffic anomaly around host 00:0c:bf:22:13

sessions of 1.5 hours) can find an attacker or at least find anomalies in traffic that can be interpreted as traces of the attack. The student needs not only to find all possible anomalies in traffic, but also to link them into a single story that will show the logic and sequence of actions of the attacker. Students also need to provide a report that should convince the investigator of the weight of the presented arguments.

In practice, to find enough evidence of an attack in 3 hours and form a security breach story can 3 students out of 10, while the rest takes more time. Students also make a report in which the screenshots of Wireshark are used as arguments, with the selection of individual fields and a textual interpretation of their values.

Students were asked to solve this use-case using visual analytics. The teacher has prepared JSON files that can contain traces of the attack. Students developed visualizations on pairs using D3.js. As a result, for 3 hours only 2 students out of 10 failed to complete the assignment. Figures 11, 12, 13, 14,

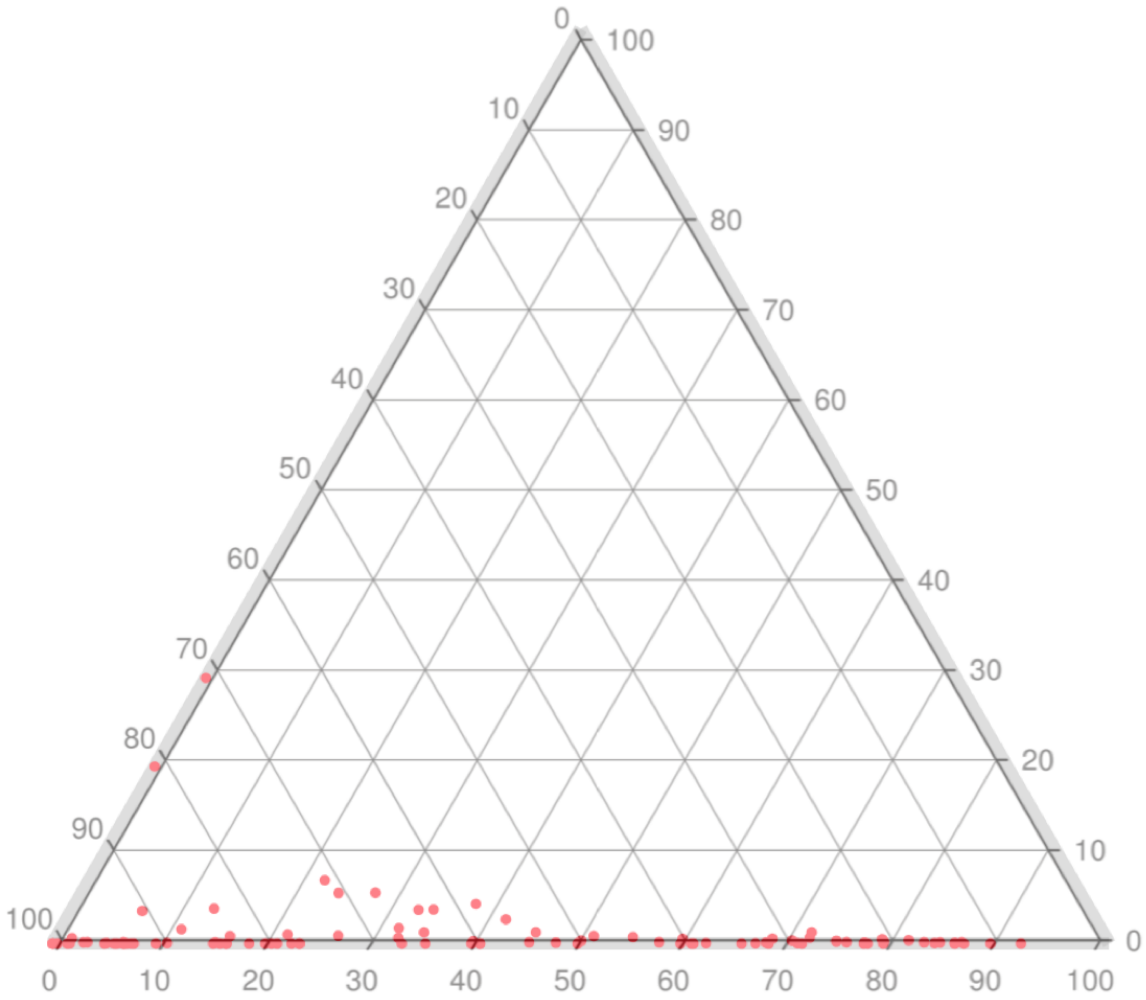

Figure 13: Triange Coordinates represents for ARP traffic

No.	Time	Source	Destination	Protocol	Length	Info
84467	879.231449	192.168.238.130	93.180.6.3	NTP	90	NTP Version 3, client
84468	879.241141	93.180.6.3	192.168.238.130	NTP	90	NTP Version 3, server
84469	879.750142	192.168.238.1	192.168.238.255	NBNS	92	Name query NB NPI403408<00>
84470	880.169586	Vmware_bf:22:13	Vmware_a4:63:4f	ARP	42	192.168.238.2 is at 00:0c:29:bf:22:13
84471	880.500369	192.168.238.1	192.168.238.255	NBNS	92	Name query NB NPI403408<00>
84472	882.170687	Vmware_bf:22:13	Vmware_a4:63:4f	ARP	42	192.168.238.2 is at 00:0c:29:bf:22:13
84473	884.122642	Vmware_89:74:8b	Vmware_e7:06:30	ARP	60	Who has 192.168.238.2? Tell 192.168.238.131
84474	884.122644	Vmware_e7:06:30	Vmware_89:74:8b	ARP	42	192.168.238.2 is at 00:50:56:e7:06:30
84475	884.171761	Vmware_bf:22:13	Vmware_a4:63:4f	ARP	42	192.168.238.2 is at 00:0c:29:bf:22:13
84476	884.450231	192.168.238.131	95.213.132.250	NTP	90	NTP Version 4, client
84477	884.460189	95.213.132.250	192.168.238.131	NTP	90	NTP Version 4, server
84478	886.172755	Vmware_bf:22:13	Vmware_a4:63:4f	ARP	42	192.168.238.2 is at 00:0c:29:bf:22:13
84479	887.128527	192.168.238.130	64.233.164.113	TCP	60	[TCP Keep-Alive] 36023 → 80 [ACK] Seq=2039 Ack=825 Win=16616 Len=0
84480	887.128528	64.233.164.113	192.168.238.130	TCP	54	[TCP Keep-Alive ACK] 80 → 36023 [ACK] Seq=825 Ack=2040 Win=64240 Len=0
84481	888.173610	Vmware_bf:22:13	Vmware_a4:63:4f	ARP	42	192.168.238.2 is at 00:0c:29:bf:22:13
84482	888.760702	192.168.238.130	64.233.164.113	TCP	60	[TCP Keep-Alive] 36024 → 80 [ACK] Seq=936 Ack=347 Win=15544 Len=0
84483	888.760704	64.233.164.113	192.168.238.130	TCP	54	[TCP Keep-Alive ACK] 80 → 36024 [ACK] Seq=347 Ack=937 Win=64240 Len=0
84484	889.044197	192.168.238.131	192.168.238.2	DNS	73	Standard query 0xd3f5 A s.youtube.com
84485	889.044198	192.168.238.131	192.168.238.2	DNS	73	Standard query 0x2a25 AAAA s.youtube.com
84486	889.045004	192.168.238.131	173.194.220.138	HTTP	1099	GET /api/stats/watchtime?st=268.512&fexp=9406851,9408495,9416126,9416179,9416

Figure 14: Representation of the ARP-spoofing attack using Wireshark

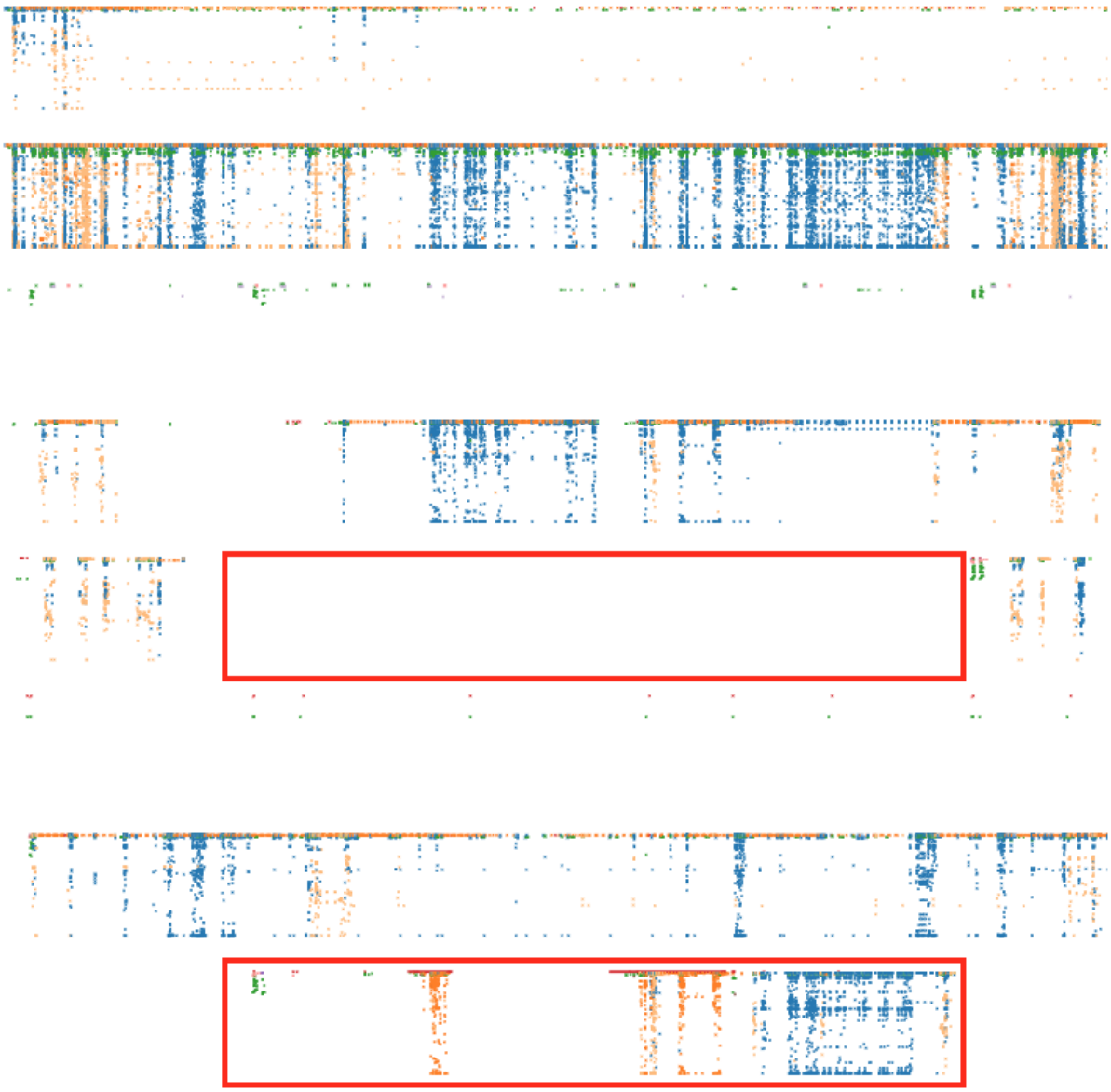

Figure 15: Scatter Plot of network traffic

15 and 16 were formed by ITMO students. It is worth noting that after completing the assignment, the students had graphic material that they used in their reports.

So, we can see that visual analytics allows solving two challenges at once. First, it is used as an analysis tool, which the investigator applies in his work. In this case, the investigator must first create slices of data which may contain information about anomalies. Then, he leads these slices to a format suitable for visualization, depending on the type of slice: if it is numerical data, the investigator can use tables or relational databases; if this is not numerical data, the investigator can use JSON files or NoSQL databases. Then he uses a specific visualization model. The investigator can use different visualization models for one set, depending on the situation.

It is likely that on one visualization model the anomalies will not look obvious, but on other not (compare Figures 15 and 16). Secondly, visualization is used as a proof, which is perfectly suited to

Figure 16: Scatter Plot of network traffic without packet length

the report, allowing one to explain the phenomena not by numbers, but in the form of a graphical image, which is much easier to understand.

Often decisions are made by people who are not experts in information security. It will be difficult to explain them basic technical aspects, especially when it is necessary to keep a lot of data in mind. Visualization simplifies this process.

Obviously, visual analytics tools cannot replace the work with traffic in Wireshark, but they do not pretend to do this. The goal of visual analytics is not to replace the work of investigators with automatic algorithms and the drawing of visual stories. The goal of visual analytics is to give powerful tools to investigators that will save time and simplify the work with data.

The presented example shows how visual analysis effectively allows one to detect an anomaly and localize an anomaly part of the traffic. However, not every anomaly in the data is a security breach. For example, in Figure 11, multiple IP-MAC connections (the right column, second on top, light-blue host) are not a security breaches, but just an ARP broadcast of packets from the router that were forgotten to be excluded from the data set (an intentional oversight on our part).

Determine a security breach can only be done with the help of Wireshark, and the investigator should still explore the detected phenomenon at a lower level at the packet level. But visualization allows one

to localize these phenomena and suggest areas in the traffic that one should look firstly. So, for example, Figure 13 demonstrates narrowing of the search area to just two points. It is therefore necessary to view just a few seconds of ARP-traffic that related to these 2 points to make sure that there was ARP spoofing on the network. At the same time, visualization allows one to show the deviation from the template. Image in Figure 13 allows one to convince a person who does not understand computer networks that the found phenomenon is different from the trends and typical behavior of the computer network. To prove the presence of an attack using bright visual representation becomes easier.

In addition, the images obtained as a result of the analysis can be combined into one visual story. Step by step, one can designate the sequence of attacks in a graphical form. Together with the usual evidence based on the analysis of traffic packets, the visual representation of the events story will help to bring the evidence together, creating solid base for suspicion and will serve as an excellent addition to the investigation report.

4 Conclusion

This paper presents how visual analysis can be used to improve efficiency of network forensics. In the paper, we proposed the technique for classifying data-slices of traffic and the using various visualization models for investigation cyber-attack using visual analytics. We showed the example of using visual analytics to find the attacker by comparing traffic analysis in WireShark and visual analysis.

Studies have shown that visual analytics can be more important in the criminal arsenal. It allows one to quickly find anomalies in a simple and explicit graphical form. In addition, images obtained as a result of visual analysis can also be used in reports. Visual evidence is always perceived much better. At the same time, visual analytics cannot replace the use of WireShark, but it can quickly reveal the place in which to look, show trends and the overall situation. To verify the detected anomalies, it is necessary to use Wireshark, but the scope and search objectives are significantly narrowed.

Future work will focus on simplifying data preprocessing mechanisms for visualization, building a consistent technique for visual analysis, and extending the benefits of visual analysis to areas of digital forensics that are different from forensic science. In this paper, we examined how one can use visual analysis in network forensics.

Acknowledgment

This research was partially supported by grants of RFBR (projects No. 16-29-09482 and 18-07-01488), by the budget (project No. AAAA-A16-116033110102-5), and by Government of the Russian Federation, Grant 08-08.

References

[1] T. V. Lillard, C. P. Garrison, C. A. Schiller, and J. Steele, *Digital Forensics for Network, Internet, and Cloud Computing: A Forensic Evidence Guide for Moving Targets and Data*. Elsevier, 2010.

[2] G. Conti, *Security Data Visualization: Graphical Techniques for Network Analysis*. No Starch Press, 2007.

[3] R. Marty, *Applied security visualization*. Addison-Wesley, 2009.

[4] J. Jacobs and B. Rudis, *Data-driven security*. John Wiley & Sons, 2014.

[5] N. Promrit, A. Mingkhwan, S. Simcharoen, and N. Namvong, "Multi-Dimensional Visualization for Network Forensic Analysis," in *Proc. of the 7th International Conference on Networked Computing (NCM'11), Gyeongju, Korea*. IEEE, September 2011, pp. 68–73.

[6] S. Krasser, G. Conti, J. Grizzard, J. Gribschaw, and H. Owen, "Real-time and forensic network data analysis using animated and coordinated visualization," in *Proc. of the 6th Annual IEEE SMC Information Assurance Workshop (IAW'05), West Point, New York, USA*. IEEE, August 2005, pp. 42–49.

[7] R. Erbacher, K. Christiansen, and A. Sundberg, "Visual network forensic techniques andprocesses," in *Proc. of the 1st Annual Symposium on Information Assurance: Intrusion Detection and Prevention (ASIA'06), Albany, New York, USA*, June 2006, pp. 72–80.

[8] A. Chechulin, M. Kolomeets, and I. Kotenko, "Visual analytics for improving efficiency of network forensics: account theft investigation," in *Proc. of the 3rd Annual International Conference on Information System and Artificial Intelligence. (ISAI'18), Suzhou, China*. Institute of Physics, June 2018.

[9] M. Kolomeec, A. Chechulin, A. Pronoza, and I. Kotenko, "Technique of Data Visualization: Example of Network Topology Display for Security Monitoring," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, vol. 7, no. 1, pp. 58–78, 2016.

[10] J. C. . Co, "Report of JPMorgan Chase & Co. Management Task Force Regarding 2012 CIO Losses," http://files.shareholder.com/downloads/ONE/2272984969x0x628656/4cb574a0-0bf5-4728-9582-625e4519b5ab/Task_Force_Report.pdf [Online; accessed on June 20, 2018], 2013.

[11] M. Kolomeec, G. Gonzalez-Granadillo, E. Doynikova, A. Chechulin, I. Kotenko, and H. Debar, "Choosing Models for Security Metrics Visualization," in *Proc. of the 2017 International Conference on Mathematical Methods, Models, and Architectures for Computer Network Security (MMM-ACNS'17), Warsaw, Poland*, ser. Lecture Notes in Computer Science, vol. 10446. Springer, Cham, August 2017, pp. 75–87.

[12] S. Few, "Save the pies for dessert," from:https://www.perceptualedge.com/articles/visual_business_intelligence/save_the_pies_for_dessert.pdf [Online; accessed on June 20, 2018], 2007.

[13] S. Tricaud and P. Saadé, "Applied parallel coordinates for logs and network traffic attack analysis," *Journal in Computer Virology*, vol. 6, pp. 1–29, February 2009.

[14] H. Choi and H. Lee, ""PCAV: Internet Attack Visualization on Parallel Coordinates"," in *Proc. of the International Conference on Information and Communications Security. (ICICS'05), Beijing, China*, ser. Lecture Notes in Computer Science, vol. 3783. Springer, Berlin, Heidelberg, 2005, pp. 454–466.

[15] R. Whitaker, "Applying information visualization to computer security applications," Master's thesis, Utah State University, 2010.

[16] D. Arendt, D. Best, R. Burtner, and C. L. Paul, "CyberPetri at CDX 2016: Real-time network situation awareness," in *Proc. of the 2016 IEEE Symposium on Visualization for Cyber Security (VizSec'16), Baltimore, Maryland, USA*. IEEE, November 2016, pp. 1–4.

[17] "National Security Agency Information Assurance Directorate Cyber defense exercise," https://www.iad.gov/iad/programs/cyber-defense-exercise/ [Online; accessed on June 20, 2018].

[18] B. Bederson, B. Shneiderman, and M. Wattenberg, ""Ordered and quantum treemaps: Making effective use of 2D space to display hierarchies"," *ACM Transactions on Graphics*, vol. 21, no. 4, pp. 833–854, October 2002.

[19] I. Kotenko, E. Doynikova, and A. Chechulin, "Security Metrics Based on Attack Graphs for the Olympic Games Scenario," in *Proc. of the 22nd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP'14), Turin, Italy*. IEEE, February 2014, pp. 533–543.

[20] I. Kotenko and E. Novikova, "VisSecAnalyzer: A Visual Analytics Tool for Network Security Assessment Security," in *Proc. of the International Conference on Availability, Reliability, and Security (CD-ARES'13), Regensburg, Germany*, ser. Lecture Notes in Computer Science, vol. 8128. Springer, Berlin, Heidelberg, September 2013, pp. 345–360.

[21] M. Kolomeets, A. Chechulin, and I. Kotenko, "Visualization Model for Monitoring of Computer Networks Security Based on the Analogue of Voronoi Diagrams," in *Proc. of the International Conference on Availability, Reliability, and Security (CD-ARES'16), Salzburg, Austria*, ser. Lecture Notes in Computer Science, vol. 9817. Springer, Cham, August 2016, pp. 141–157.

[22] "PCAP-file and description of investigation," http://comsec.spb.ru/files/ForensicLab.htm [Online; accessed on June 20, 2018].

Author Biography

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 350 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.

Maxim Kolomeets is currently a PhD student of ITMO University and a researcher 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. He is the author of more than 20 refereed publications.

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 70 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.

Yannick Chevalier is a former mathematics and computer science student of ÉNS Lyon. He has received a PhD from University Nancy 1 in 2003, and is since 2004 an associate professor at University Toulouse 3. He has co-authored 13 journals and 27 conference papers, and has participated in the European AVISS, AVISPA, and Avantssar Projects. His work was cited more than 2000 times according to Google Scholar.