C O V E N T R Y
U N I V E R S I T Y

Faculty of Engineering and Computing

Department of Computing

MSc. Software Development

M08CDE – Individual Project

Visualisation of Cyber Security Data With Data Mining to Facilitate More Accurate and Relevant Analysis

Author: Yavuz Atlas

SID: 4662475

Supervisor: Robert Bird

Submitted in partial fulfilment of the requirements for the Degree of Master of Software Development

Academic Year: 2013/14

Declaration of Originality

This project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet, etc) has been acknowledged within the main report to an entry in the References list.

I agree that an electronic copy of this report may be stored and used for the purposes of plagiarism prevention and detection.

I understand that cheating and plagiarism constitute a breach of University Regulations and will be dealt with accordingly.

Copyright

Signed: Date: 18/08/2014

Office Stamp

Abstract

With increasing of cyber security data size new automatize and manual analysing methods are being developed. Cyber security visualization tools are relatively new and effective approach in manual analysing methods. On the other hand data mining based security tools are effective approach in automatize analysing methods with lack of human interaction.

In this report a hybrid approach involves information visualization and data mining is suggested and a software is developed to evaluate how useful this approach is. A clustering algorithm named DBSCAN is preferred and different colours are used to encode different clusters on a scatter plot.

First remarkable thing in this project is that seven different attributes of an IP packet are visualized in a scatter plot in a software. Each unique source IP and destination IP pair is directly mapped to system. Number of unique destination port of the each IP pair is encoded with size attribute. And length, windows size, sequence NO and TTL values used in data mining process and reflected to display with colour attribute.

Second point is that a data mining algorithm is used to assist security analysers. Data mining algorithms in cyber security is used in many projects. But they do everything automatically which is reducing reliability of these systems. But in this project data mining is used only as an advice tool which is also appropriate for information visualization theory.

Table of Contents

Abstract 2
Table of Contents 3
Additional Materials on the Accompanying CD 4
1 Introduction 5
1.1 Background to the Project 5
1.1.1 Overview of Information Visualization 6
1.1.2 Current Intrusion Detection Approaches 10
1.1.3 Data Mining 11
1.2 Project Objectives 12
2 Literature Review 13
2.1 Cyber Security Visualization Tools 13
2.2 Data Mining in Intrusion Detection 15
2.3 Chapter Summary 16
3 Methodology 17
3.1 Process Model 17
3.2 Secondary Research 18
3.3 Chapter Summary 18
4 Requirements 19
4.1 Functional Requirements 19
4.2 Non-Functional Requirements 19
5 Analysis 21
5.1 Use Case Diagram 21
5.2 Chapter Summary 22
6 Design 23
6.1 Class Design Diagram 23
6.2 Sequence Diagram 24
6.3 User Interface 25
6.4 Chapter Summary 28
7 Implementation 29
7.1 Implementation Process of the Software 29
7.2 Chapter Summary 30
8 Testing 31
8.1 Port Scanning 32
8.2 DOS Attack 35
8.3 Password Attack 38

8.4 Chapter Summary 39
9 Project Management 40
9.1 Project Schedule 40
9.2 Risk Management 41
10 Critical Appraisal 42
11 Conclusions 44
11.1 Achievements 44
11.2 Future Work 44
12 Student Reflections 45
Bibliography and References 46

Additional Materials on the Accompanying CD

Accompanying CD includes following folders:

Code: Contains code of the software.
Datasets: pcap files and processed CSV files are in this folder.
Project plan: Project plan in pod format is in this folder.
Report: PDF version of this report.
UML diagrams: UML diagrams which can be seen with StarUML.

1 Introduction

1.1 Background to the Project

While demand on information systems increasing security concerns on them become more serious (Song et al. 2011). Most of the information systems keep individuals' personal information on their databases and send the data to other machines by network cables. Security of these data and accessibility of the systems are becoming more crucial day to day. In this context cyber security researchers working to find better methods to prevent, detect and react to cyber attacks that aim to exploit security vulnerabilities. But it becomes harder day to day since while complication level of cyber attacks is increasing, knowledge to carry out them is decreasing (Goodall 2008).

With both rise of the usage of information systems and security concerns on them, a new concept is occurred named cyber security (information security) which is now a core element of information technologies. Cyber security can be described as securing information that is in digital environments like hard disks, rams or network cables (Ciampa 2011).

There is a big tendency to carry applications to web which directly depend to internet instead of clients. Also most of the client applications use internet for different purposes and mostly without asking to users. Under those conditions velocity of the internet data is growing. Despite this tendency is beneficial for regular users in terms of usability, in cyber security side it contributes problem. Especially internet data used by big corporations are increasing drastically in this context. There are some automatic tools to distinguish normal and malicious usage for instance firewall, antivirus, IDS and IPS.

Log and traffic analysing is a fact of cyber security because of inadequacy of automated security tools. But when considered amount of data reading every log line which can be millions of lines in some cases is not a realistic idea. Analysts mostly use their experiments, knowledge and senses to cope with cyber security data (Fink et al. 2009). Another approach to analyse cyber security data is information visualization. Visualization's strength is taking advantage of human perception. Human brain is a very successful pattern recogniser (Ware 2004). Patterns can be detected easily by analysers instead of automated tools with information visualization approach. Thus with visualization, analysts can process data and detect anomalies and patterns easier than traditional text based analysis techniques (Conti et al. 2005)

Visualization tools have some handicaps though. When data is too big information visualization systems do not let a healthy analyse. What this project offers is that using a clustering algorithm to separate cyber security data rationally to make visual analysing more efficient.

1.1.1 Overview of Information Visualization

Human brain and sense of sight provide a powerful pattern seeking system (Ware 2004). Information visualization is a discipline that searches ways to exploit this system to fill gap between human cognition and textual data with help of other disciplines like statistics, computer sciences, psychology, and geometry (Kmeťová 2010). Information Visualization can be used for exploring and analysing data and taking decisions with that information (Goodall 2008).

Information visualization is likely about abstract data. In many cases it is impossible to map data directly to visual display. Researches are mainly focused on finding better ways to present abstract data to users with graphics. Also usually users of information visualization systems do not have to be expert. But there is a broad confusion about information visualization and scientific visualization though. Scientific visualization directly reflects a picture of a fact such as data of a molecule and those systems are mostly developed for experts in contrast to information visualization (Gershon, Stephen, & Card 1998).

Understanding the way that how human perception works is quite important to develop high quality visualization systems. Human perception works with its own rules and a visualization system must be developed by taking into account those rules. Many researches were conducted to discover features of human perception. Phycology researchers discovered that humans can detect some properties of an image without a focussed examination. Visual properties that can be detected by human visual system in less than 200-250 milliseconds are called pre-attentive properties (Healey & Enns 2012). For example a red dot in a display that full with blue dots can be detected with pre-attentive processing (figure 1).

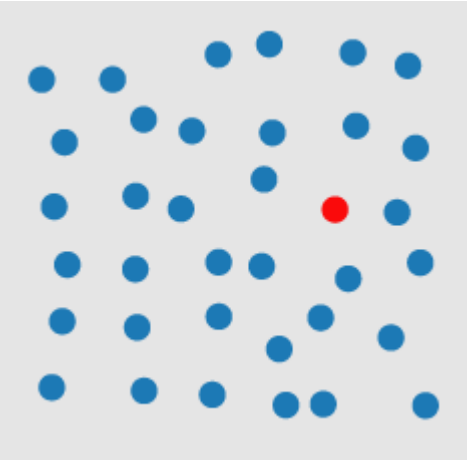


Figure 1. Pre-attentive processing (Healey 2009)

Pre-attentive visual properties can be grouped as four categories: Form, colour, motion and spatial position. All of those visual properties also have attributes. For instance size, blur, line length, line width are attributes of form while hue and intensity are attributes of colour (Ware 2004). Each attribute reflects one dimension of data. Also attributes can be combined to reflect more than one data dimension. For example a blue triangle and a bigger dot can be used in a display that full with red dots. In this case display has two attributes: shape and size.

Only appropriate attributes must be used together. When inappropriate attributes are combined it violates pre-attentive processing and sometimes information that is tried to express becomes invisible. Attributes those do not work well together are called as integral dimensions and attributes which can reflect data together by pre-attentive processing are called separable

dimensions (Marty 2008). For instance colour and position are separable dimensions while height and width attributes are integral dimensions (figure 2).

Figure 2. Integral and separable dimensions (Marty 2008:11)

One of the researches on human perception is carried out by German psychologists named Max Wertheimer, Wolfgang Köhler, and Kurt Koffka in about 1910. After researches they developed Gestalt psychology (Nesbitt & Friedrich 2002). Seven Gestalt principles are below (Rusu et al. 2011):

1. Proximity: According to this rule things close together are perceived as in same group. Also human perception tended to grouped visual elements based on their densities. Thus related data should be visualized close to each other.
2. Similarity: Objects look similar to each other are tended to be seem as a group. That means if same attributes are used for certain objects, human visual system perceives them in same group.
3. Closure: Incomplete elements are completed by human perception. For instance an incomplete square's missing place is completed by the perception.
4. Symmetry: Symmetric objects are perceived as in same group even if there is distance between them.
5. Continuation: When objects are occluded each object is perceived as separately.
6. Common fate: Objects move together are perceived as part of the same group.
7. Figure-ground: figures which are like an object are perceived as at foreground.

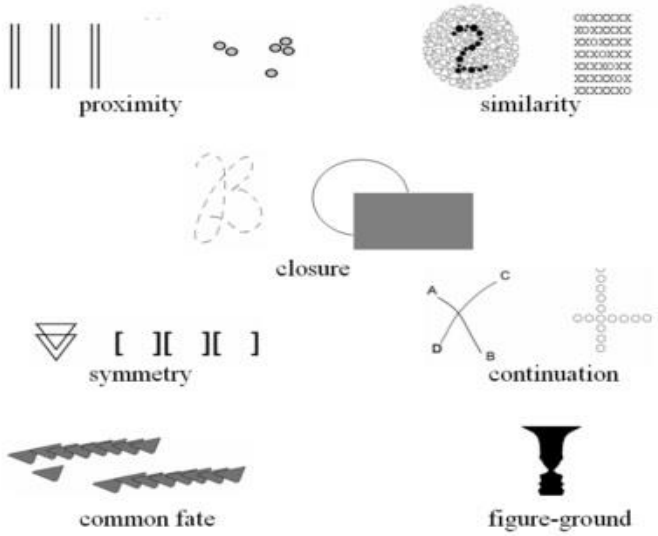


Figure 3. Gestalt Principles (Rusu et al. 2011:488)

Every visualization system needs a dataset to process. Data types which are processed by visual systems must be known before developing the system. Mostly used data types are:

categorical, ordinal, interval, ratio, hierarchical and graph data (Conti G. 2007). Categorical variables do not have an ordering and they are name or category based variables. Internet protocol name is an example for categorical variables. Difference of ordinal and categorical values is that they have an ordering. However their ordering is not based equal parts. For example priority of IDS alerts have an ordering but it cannot be said that distance between low and medium alerts is same with distance between medium and high. If distances were same, variable would be called interval. Interval variables are quantitative and ordered. Ratio variables have a real 0 point. If data is at 0, that means there is not any of that data (Stevens 1946). Packet size is a ratio variable. Hierarchical and graph are advanced data types. Directory structure in computers is an example for hierarchical variables while network packets can be took as a graph data because a network packet can follow different network routes (Conti 2007).

There are many ways to represent data graphically. A graph can express information if follows two criteria: expressiveness and effectiveness (Mackinlay 1986). Expressiveness criteria is about whether or not a graphic express all and only desired information. Effectiveness criteria is for determining which graphical approach is better to reflect a certain situation. While preparing a graphical display below steps can be followed (Schriger & Cooper 2001):

1. Define data which will be depicted carefully
2. Determine which one is the best way to depict data: text, table or graphic
3. Determine purpose of the graphic.
4. Determine main variables to depict information.
5. If there is pairing depict it.
6. Select most appropriate graphical layout for current situation.
7. Use clearly defined labels for axes and consistent metric.
8. Do not use unnecessary ink and exploit each graphical element at maximum level

Selecting a graphical layout is a challenging work. Inappropriate graphics obviously violate expressiveness of the visualization. There are many graphical layout types and each of them has its own purpose, advantages and disadvantages. Bar charts work well for comparison purpose. For representing time series, bar charts can be used besides line charts and radar plot. To compare a certain part of a data to whole part stacked bar chart, multiple bar chart, pie chart and treemap can be used. To show deviation of data bar chart and line chart is appropriate. Histograms work well at distributions and scatter diagram and bubble chart are ideal for correlation presentation (Jonge 2012).

Above layouts are simple and common ones. Also there are more sophisticated graphical layouts. One of them is parallel coordinates. Parallel coordinates approach is especially useful for representing multi-dimensional data. With this approach n dimensional data is showed at 2D layout. Axes that show dimensions are draw parallel and in theory there is no limit for axe (or dimension) number in this approach (Berthold & Hall 2003).

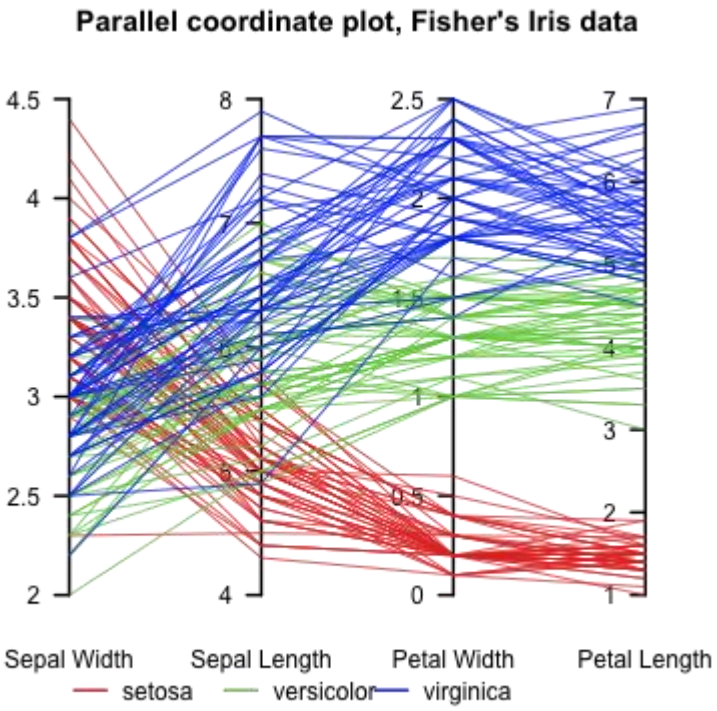


Figure 4. Parallel Coordinates (Wikipedia 2014)

Despite considering to exploiting human perception at highest level is one of the most important tasks of information visualization developers, it is not enough to provide usable systems to analysers. Visualization systems have to provide ability to drill down to investigate data detailed with interaction techniques (Fink et al. 2009). Ben Shneiderman suggests Visual Information-Seeking Mantra to prepare useful graphical interfaces (Shneiderman 1996). This method can be summarized with “overview first, zoom and filter, then details on demand” formula. According to Visual Information-Seeking Mantra a visualization system must provide an overview of the situation firstly. After the user is aware of the overview of the situation, he/she can focus on a specific part of the big picture with zooming and filtering abilities which are provided by the system. And finally detailed information can be gathered by selecting a part of the focussed area.

To reach Shneiderman’s Visual Information-Seeking Mantra interaction techniques must be provided. Interaction techniques for visual systems are below (Keim 2002) :

1. **Interactive Filtering:** While examining data it is important to focus on a specific part of the data. To obtain this there are mainly two kind of filtering method: directly selecting a subset (browsing) and specifying properties of the desired data (querying). First one is difficult on large datasets while getting satisfying results is hard with latter.
2. **Interactive Zooming:** Besides providing overview of data, also giving a larger and detailed picture is a useful feature for visualization systems. While giving a detailed picture some additional information like labels can be added also.
3. **Interactive Distortion:** With this technique a visualization system shows an overview of the picture while providing a detailed view at same time.
4. **Interactive Linking and Brushing:** Every graphical layout has advantages and disadvantages. To overcome disadvantages of them graphical layouts can be used together. In interactive linking and brushing technique displays must be connected each

other interactively (linking) and when an area is selected in a display (brushing) other displays data must be changed according to selecting action.

1.1.2 Current Intrusion Detection Approaches

If a user takes an illegitimate action while using an information system, it is called intrusion (Jones & Sielken 2000). There are many methods and tools to detect intrusions both on network and hosts automatically. Intrusion detection approaches can be divided into two main groups: misuse and anomaly based detection.

In misuse detection approach, malicious usage in cyber security data is detected by predefined signatures. This approach is successful at detecting well known attacks. With these systems well known attacks can be detected with low false positive rates, but on the other hand misuse detection approach is weak at zero day attacks (Song et al. 2011). Signature databases of misuse detection based systems must be always up to date since these systems are completely dependent to their signature databases. There are two types of misuse detection technique (Karthikeyan 2010):

1. **Expression Matching:** Expression matching is the simplest way of misuse based intrusion detection. The idea is determining a certain expression and searching it at the cyber security data like system logs or raw network packets. This expression can be a system command that means someone tries to reach password file or an expression which is included by a known malware.
2. **State Transition Analysis:** In this method attacks are defined as actions that are taken in sequence. When every action is examined separately it may not mean anything but a set of actions is matched with a signature, intrusion detection system labels it as an intrusion attempt.

Anomaly detection based systems try to distinguish abnormal data from normal cyber security data and abnormal data is considered as an intrusion attempt. This approach gives worse result at detecting well known attacks than misuse based detection approach. But it is better at detecting unknown or undefined attacks. Anomaly detection techniques usually consist of two phase: training and testing. In training phase, the system must be trained with attack-free (normal) cyber security data. After training phase the system generates functions that represent normal data and uses those functions to distinguish normal and abnormal data. Determining if a cyber security data is attack-free is not easy in real environment thus training these systems is a crucial problem. On the other hand because of these systems are customized for their own environments, it is quite difficult for an attacker to understand behaviours of anomaly based detection systems which causes security level of the systems to increase (Patcha & Park 2007). There are three types of anomaly detection techniques (Karthikeyan 2010):

1. **Protocol Anomaly Detection:** Protocol anomalies are related with every exception about protocol format and protocol behaviour. Protocol anomalies include network, transport and application layer anomalies.
2. **Application Payload Detection:** This anomaly detection technique must be supported by application protocol analysis. Application payload anomaly based detection techniques are about content of the cyber security data. For instance an illegitimate entry in a form area at a web site can be detected with this technique.
3. **Statistical Anomaly Detection:** Difference between actual and expected cyber security data can be exposed with statistical properties like mean and variance. Intrusion

detection system gives score to actual traffic, based on a statistical method in these techniques and if this score is bigger than a pre-defined threshold the traffic is labelled as intrusion (Gyanchandani et al. 2012).

Also Gyanchandani et al. (2012) discussed machnie learning and data mining based detection techniques under anomaly detection approach. Machine learning is a method that programs which use this method improve their own performance over time. Machine learning alghorithms change their execution way during their working time. Their biggest disadvantage is that these techniques use system resources at high level. Bayesian approach, neural networks, fuzzy logic, genetic alghorithms and support vector machines are examples of this method. Data mining methods are examined below more detailed.

Common points of the above approaches are that they undervalue contribution of humans. Computers can process big amount of cyber security data with automated methods but they cannot reach human analytical skills. Thus providing opportunity to analysts processing big cyber security data with their own skills is a need. In this context information visualization (or cyber security visualization) is another approach at intrusion detection (Goodall 2008).

1.1.3 Data Mining

Data mining (knowledge discovery) is pattern or knowledge extraction from big amount of data by help of statistics, information retrieval, machine learning, and pattern recognition (Gudadhe, Prasad, & Kapil 2010). Data mining is used in many areas such as finance, medicine, social sciences and cyber security.

There are mainly two different approaches in data mining: supervised and unsupervised. Supervised methods use training data. In supervised methods data mining application must be trained with labelled data. Then unlabelled data is used to test success rate of algorithm. On the other hand knowledge is extracted from data without a training phase in unsupervised methods (Du 2011). Every data mining method uses one of these two approaches. There are six different data mining method (Fayyad 1996):

1. **Classification:** Classification is supervised data mining method. With classification a data item's class is determined by a function generated by help of training process.
2. **Regression:** Regression is used to find a function that map to a dataset. Regression is a prediction method also.
3. **Clustering:** Clustering is a descriptive and unsupervised data mining method. With this method a dataset is divided into clusters based on similarity of data.
4. **Summarization:** Set of methods that aim to describe sub sets of whole data.
5. **Dependency Modelling:** Dependency modelling exposes dependencies between variables.
6. **Change and Deviation Detection:** This method aims to reveal significant differences between two different conditions of the same data set.

As mentioned unsupervised data mining methods do not need a training phase with labelled data and this feature makes these methods quite attractive especially in cyber security area. They only need some parameters to work. Obtaining labelled data for training is an important problem because in real world it is hard to guarantee that a dataset does not include malicious

data, especially when we consider about zero day attacks (Song et al. 2011). Also even if an attack-free network traffic data is obtained for training of an unsupervised system if the network environment is changed normal traffic pattern must be calculated again (Du 2011). Thus supervised data mining algorithms are not as reliable as unsupervised algorithms at changing network environments.

1.2 Project Objectives

The project is, developing a cyber security visualization tool that uses a data mining algorithm to separate cyber security data rationally to facilitate easier visual analysis. Objectives of the project are:

- Research information visualization theory and tools.
- Research and compare data mining algorithms to find appropriate one for the project.
- Research network security analysing techniques.
- Determine most appropriate visual elements to present information in this project.
- Develop a cyber security visualization tool.

2 Literature Review

2.1 Cyber Security Visualization Tools

RUMINT (Conti et al. 2005) is a well-known cyber security visualization tool. This tool is available on the internet for free. RUMINT has ability to capture network traffic. However RUMINT developed for Windows environment and worked in Wine in Linux/Unix environment, thus live capture mode is only available in Windows. Also RUMINT has capability to import pcap files for forensic analysis. Maximum packet capacity of RUMINT is 30000. That means bigger traffic than 30000 packets must be analysed in the tool separately. After importing traffic data file, RUMINT shows data simultaneously as capturing the traffic in that time. That makes easier to detect changes in the traffic because when whole traffic is imported to the tool, visualization screens suffer with occlusion and labelling problem usually and that final picture can be less useful than animated visualization. RUMINT has filtering option also. RUMINT's one of the most remarkable points is its seven different visualization windows. That provides analysts to analyse traffic with different perspectives and correlate them. Especially parallel plot view is quite useful. Also it is very easy to be familiar with RUMINT because of its VCR interface that people already familiar from music players.

Visual Information Security Utility for Administration Live (VISUAL) (Ball, Fink, & North 2004) is a tool which aims to visualize communication between external and internal hosts to expose communication patterns. VISUAL uses data captured by tools like TCP-Dump. In this tool, internal hosts are represented by a grid that every cell of it symbolizes one internal host. Those grids' sizes change depends on their activity volume. External hosts are shown by yellow squares around the grid. Port numbers of external hosts which are used for communication with the internal network are showed with horizontal lines in the squares. Low numbered ports' lines are at up of the square while high numbered ones at down of it. And there are straight lines between squares and cells to show communications. Colours of lines denote some information about communication. For instance communication from an external host to an internal host without response is showed by red line, while if the internal host responses, this line appear with blue. VISUAL uses visualization mantra of Shneiderman. Analyst can focus on several or one host on demand after overview of network. Also analysts can reach; textual information of IP address of a selected host, IP address of the hosts which are communicated with selected host, used TCP and UDP ports and traffic percentage of a the host. VISUAL is able to process up to 2500 internal and 10000 external hosts. The tool has an occlusion problem. Each connection is represented by single line to reduce occlusion and also lines are transparent to prevent overlapping squares of external hosts.

PortVis (McPherson 2004) is a specialized tool for analysing port activities. Goal of PortVis is providing an environment for analysts to detect only high level port activities. For instance a port scan activity that probes all ports of a host can be detect with PortVis easily, while an intrusion that consists of small number of connections cannot be detected. Also traffic that causes abnormal pattern cannot be perceived with PortVis. PortVis' main visualization display is a 256 X 256 grid that each dot symbolizes a port. Another display provides a magnified view of selected area at main display. When a port is selected from that magnified view, port details are showed in a 3D view named port detail. This display shows time and traffic volume information. It is remarkable that all screens are connected to each other. Also there are filtering features.

Besides analysing low level incidents, being aware of big picture of network is quite crucial. VisAlert (Livnat 2005) is a cyber security visualization tool that aims to provide situational

awareness. VisAlert visualizes alerts from different sources such as IDS and system log files and let analyser to correlate them. According to developers of VisAlert an alert must include at least three attribute: host information (where), incident time (when) and alert type (what) which is authors called as w3. Alerts can be grouped by these three attributes in the tool. VisAlert uses a radial visualization technique that displays network topology centre of it. Alerts are showed around the ring and they are grouped by their sources and ring width shows alert times. Despite this tool facilitates to be aware of big picture it does not have capability to drill down through alerts.

NVisionIP (Lakkaraju et al. 2005) is another tool to contribute situational awareness of analysts. The tool shows all B class network in a single screen. In the main screen (galaxy view) subnets are showed at horizontal axis while hosts are showed at vertical axis. Host activities like traffic volume and port number are represented by colourful dots but there is not a capability to change colour set. Also it is impossible to recognise host communication from the visualization screen. Some filtering options based on IP address, ports, protocols and volume of traffic can be applied to this screen. Another feature of the tool is a magnifier function which provides analysing a specific part of the network more detailed and quickly. NVisionIP has two more views except galaxy view: small multiple view and machine view. While data is being analysed in galaxy view analyst can focus on a specific area with small multiple views which show information of several hosts. Most detailed view is machine view that lets analyser to focus on a single host. There is visual statistical information about protocol and port information of the host's data flow in this view. It can be said while NVisionIP is committing a big picture of a B class network it is also providing visualization about low level situation of the network. The tool has lack of performance while processing big datasets; it is a negative point for a tool targeting B class networks.

TNV (Goodall 2005) is a free and downloadable cyber security analysis tool. Main idea of TNV is providing analysis opportunity of packet captures without losing overview of context. TNV uses pcap files. TNV classifies hosts as local and external after entering home network information. TNV uses a matrix view that shows local hosts at rows and time interval at columns. Also each cell shows incoming and outgoing packets with colourful arrows. Packets' colours indicate protocol and flag information of TCP packets. Remote hosts are at left side of the matrix that are connected to local hosts with curvy lines. Thus it can be seen easily which remote/local host communicating with which local host, distribution of packets based on time and packet protocols. There is a time slider bottom of the screen that lets adjust time period for analysing, thus a more specific time can be analysed. Port activities are showed at rightmost display. This screen does not have interactivity with main matrix screen but packet details screen. When right clicked on a packet row at packet details screen there is an option to show port activity related with that packet. At rightmost display there are two more pane except port activity pane. One of them is for configuring display properties. This pane is providing too limited configuration facility. Other pane is for filtering data which is too limited also. Despite TNV does not have a packet importing limit, it has problem with loading big files and that prevents analysing big amount of data.

Vast majority of the visualization systems use 2D visualization, however there are 3D visualization systems also. One of them is The Spinning Cube of Potential Doom (Lau 2004). This tool is an animated tool which focuses on port scans. The system is only using data collected by an intrusion detection system named Bro. There is one display in the tool that uses a 3D scatter plot. In the scatter plot each axis represents certain information of a Bro data. Local addresses are represented by X axis while Z axis shows whole ip space except local addresses and Y axis shows port numbers that are used connection between local and remote hosts. Because of Z axis is for whole ip space between 0.0.0.0 and 255.255.255.255 it is obvious that

there is a de facto limit for the tool, despite there is not such limit algorithmically. Dots in scatter plot represent TCP connection. White dots mean three way handshake is completed and other colours mean the connection is incomplete. Colours for incomplete connections varying by port number which decided based on rainbow colour map. Occlusion is a big problem for especially 3D visualization systems. The Spinning Cube of Potential Doom is not an exception at this point. The tool has rotating capability to overcome this common problem. Animating capability provides time information of connections. The tool is useful to understand high level network activities but it is not suitable to go deeper. Also only purpose of the tool is detecting port scans but other intrusion attempts.

2.2 Data Mining in Intrusion Detection

MADAM ID (Mining Audit Data for Automated Models for Intrusion Detection) (Lee 2000) is one of the first studies that data mining is used for intrusion detection. MADAM ID uses system audit data instead of network data to detect malicious usage. This framework has classifier and meta-classifier, association rule and frequent episode components. Process of MADAM ID starts with applying data mining to audit data for finding frequent patterns. Than extracts predictive features from them. Finally classification algorithms are used to generate intrusion detection models.

ADAM (Audit Data and Mining) is a testbed to show how successful data mining in intrusion detection (Barbará et al. 2001). This system uses TCPdump data. Association rules and classification applied for anomaly detection in ADAM. ADAM uses both labelled and attack free network traffic data in training phase. Firstly the system is trained with attack free network data which is provided by help of misuse detection systems. After then it is trained with entire dataset to detect attacks that are not included by first phase. And eventually ADAM uses a classifier to allocate each suspicious connection as a known attack, unknown attack or false alarm. With this hybrid method it is aimed to reduce false alarm rate. ADAM's biggest disadvantage is that it has dependency to an attack-free network data which is hard to obtain from real world network traffic.

MINDS (Ertoz, et al. 2004) is a data mining suite which is developed by Minnesota Intrusion Detection System Group for intrusion detection. MINDS has different modules for different intrusion activities such as scan detector and anomaly detector. In MINDS' anomaly detector an algorithm named Local Outlier Factor (LOF) is used. MINDS monitors network traffic and assigns an anomaly score (LOF score) to each network connection. Then a cyber security analyst checks connections that have highest LOF score to determine whether or not connections are intrusion. This approach does not ignore human contribution to intrusion detection process but it can be unsuitable at examining big amount of real time network data. Also MINDS is very successful on detecting intrusions including attacks that are not detected by signature based IDSes.

Jiong and Zhang (2006) develop an anomaly based intrusion detection framework. In this study random forest algorithm which is an unsupervised outlier detection algorithm is applied to network data. The system captures network traffic data by itself and after this point offline processing phases are started. A dataset is generated after pre-processing of network data. Then random-forest algorithm is used to build service-based patterns. Outliers can be found with those patterns. When an outlier is detected the system raises alert. This method does not need an attack free training data and outliers are obtained from regular network traffic data. Despite the framework has successful results over KDD'99 dataset, it cannot perform real time

detection because of performance issues. Also the framework is not successful over attack types that carry out with high number of connections since these connections are not outlier.

Horng et al. (2011) have developed a hybrid system that uses both SVM and hierarchical clustering algorithms. In this study SVM algorithm uses training data which is provided by hierarchical clustering algorithm. While training time is reduced, SVM algorithm's performance is increased with this method.

Liao and Vemuri used a text mining method to detect intrusions based on frequency of system calls. They used KNN classifier which is popular at document categorization. In this approach every system call is represented as a word while every system calls collection which belongs to a program is represented as a document. Programs are classified as normal or intrusive after applying text mining methods to system calls. Authors evaluated the method with 1998 DARPA BSM audit data and results showed that the method worked efficient with low false positive rates. In normal cases a real document set which consists of real documents and words includes about 1500 distinct words. One reason of why the method is successful is that there are only 100 distinct system calls in DARPA dataset, thus there is no need for producing a dictionary which involves only a part of the system calls (Liao & Vemuri 2002).

2.3 Chapter Summary

In this chapter, a critical review is conducted on cyber security visualization tools and intrusion detection tools that use data mining. Since it is not accomplished to find a significant study that covers cyber security, data mining and information visualization this method is preferred.

3 Methodology

3.1 Process Model

This project basically aims to develop a software product to reflect information of cyber security data by exploiting human perception and carrying out a research to have corresponding knowledge to develop the product. Despite surveys to understand needs of cyber security analysers and studies on perception because of the nature of the area probably some changes will be conducted over visual elements after first development. Also probably some new information will be learnt during development phase and this information can help to enhance actual product. Thus development process model must have flexibility to change the product. These reasons show that waterfall model is not appropriate for such a project. Requirements must be certain in waterfall model because this model is not suitable to make changes on product during development.

First tasks in this project are research and literature review which were conducted in first two chapters. In light of the knowledge which was gathered in previous chapters a new research will be carried out for analyzing requirements of cyber security researchers. Then proof of concept codes will be implemented and after this point software development phase will be started. In this project throw away prototyping will be used because of the conditions of the project which are mentioned above.

Software development process models are abstract representations of software development processes (Munassar & Govardhan 2010). Process models do not include anything about how software to be developed. They are only abstractions of software development phases. One of the software development process models is Prototyping. Prototyping is a process model that, ensure developers to develop software which customer wants by filling gaps between developers and customer. This is provided by presenting several working but incomplete versions of the system to customers. These incomplete versions are called prototype. Customer can check this system and after makes clearer his/her needs, can ask to change features of the system. This amending and demonstration loop continues until the system does what customer wants. Thus usually there are more than one prototype in software projects.

Prototypes are basic versions of a complete system to represent concepts of the system which are developed quickly (Sommerville 2010). Prototypes can be thrown away after demonstration. These types of prototypes are called throwaway prototypes. Throw away prototypes are usually implemented by high level programming languages. Second type of prototype is evolutionary prototypes. Evolutionary prototypes are initial version of the final software product (Bell 2005).

According to Douglas Bell (2005) goals of software engineering are: meeting needs of users, low cost of production, high performance, portability, low cost of maintenance, high reliability and delivery on time. Despite these goals indicate commercial projects, especially some of them can be mapped to this project easily. Because it is a requirement for this project to meet needs of cyber security analysts, also high performance and delivery on time.

While testing prototypes need of analysts will emerge clearly and changes will be made in minimum time. Performance issues will be spotted on prototypes thus changes can be done at early stages of the project. And also new features can be added after the subject is understood better by the developer.

Despite prototyping seems ideal for this project there are some pitfalls also. For instance, management of a prototyping project may be difficult because predicting how many prototypes will be developed can be hard (Maheswari & Jain 2012).

3.2 Secondary Research

This projects' intended user group is cyber security analysts. For that reason their opinions about cyber security visualization, their analysis methods and their habits are quite important for this project. There are surveys on both information visualization and cyber security visualization areas and this project will take advantage of these researches.

Second important resource is research on information visualization, intrusion detection approaches and data mining which is involved by first chapter of this report. And finally strong and weak sides of cyber security visualization tools and data mining based intrusion detection tools which are involved by literature review chapter will be taken into account.

3.3 Chapter Summary

In this chapter requirements of the project are told and in light of it explained that why waterfall process model is not suitable. Then prototyping model's features that are related with requirements of the project are introduced. Also this process model is evaluated with goals of software engineering. And finally given information about secondary research of the project.

4 Requirements

A new research which involves secondary sources is conducted to obtain requirements of cyber security researchers. Some surveys obtained these requirements by interviewing and observing analysts and others investigated tools and literature to understand needs of the area. Results of this study will be given under two subtitles:

4.1 Functional Requirements

Functional requirements of cyber security analysts are usually indicating features that provide flexibility to them in analysing process.

- Conventional Cyber Security Visualization tools aim to provide situational awareness and leave finding critical events to analyser. But also pointing out critical events could be useful, which is a current trend that is followed by new visualization developers (Shiravi, Shiravi, & Ghorbani 2012).
- Visualization system should be able to use different types of data format to facilitate collective work with other tools (Fink et al. 2009).
- Different types of data can be processed by visualization tool to provide correlation between them without opening new windows over and over (Fink et al. 2009). For instance if a cyber security visualization tool can handle both IDS alerts and IP packet headers, it can be useful for analysts.
- Data which is imported to visualization tool can be manipulated without changing original data (Fink et al. 2009).
- Most of the cyber security analysts think that visualization tools hide really what is going on because the tools mostly do not provide to reach real data (Fink et al. 2009).
- Finally information visualization mantra which is covered in introduction chapter in this report is an important guideline for all information visualization developers. It can be summarized with “Overview first, zoom and filter, then details-on-demand” (Shneiderman 1996).

4.2 Non-Functional Requirements

Non-Functional requirements of cyber security visualization tools are usually related with size of cyber security data. While cyber security data size is growing day to day, also screen resolutions and performance of computers are growing. However growing of latters is not as fast as former.

- Visualization tools should be able to cope with data noise which is quite successful to ruin visualization and hide critical information (Conti, Ahamad, & Stasko 2005).
- Occlusion of visual elements is another and common problem of visualization tools (Conti, Ahamad, & Stasko 2005). Especially 3D visualization tools suffer with occlusion problem (Shneiderman 1996).
- Visualization tools should beware of label occlusion problem. It must be determined which visual data need label in which case carefully to reduce labelling occlusion which does not have a certain solution (Conti G. 2007).
- Data should be scaled by visualization tools appropriately (Shiravi, Shiravi, & Ghorbani 2012).
- Cyber security visualization tools must be fast enough to serve to a security analyst. When considered how much data must be processed it is a very important point (Fink 2009).

- Importing data to and exporting data from a visualization tool must be easy enough (Fink 2009).
- Time of importing data and data size to store must be considered carefully (Fink 2009).

5 Analysis

5.1 Use Case Diagram

Use case diagrams depict interaction between systems and users. Use cases only include main usage scenarios of the system. Thus they are useful tools to both clarify requirement of users and clarify thoughts of the project team. Figure 5 shows use case diagram of the cyber security visualization software.

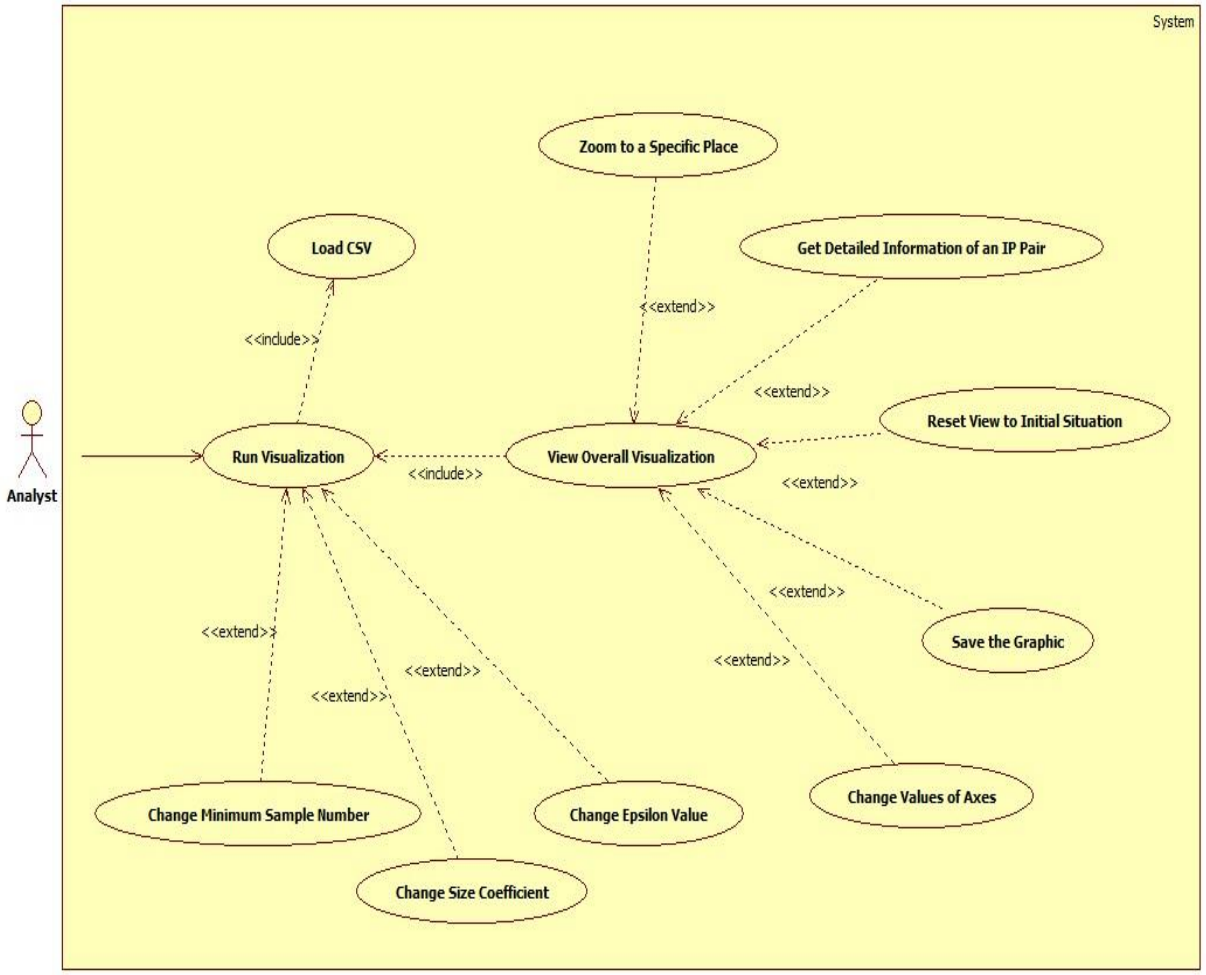


Figure 5. Cyber Security Visualization System Use case Diagram

In this system there is one actor which represents cyber security analysts. After system is started by analyst some values must be determined. Because of “minimum sample number”, “size coefficient” and “epsilon value” have default values analyst does not have to enter new values for them but if he/she wants these values can be changed. Loading a CSV file is a mandatory part of the system.

After running the system, analyst reaches new screen that shows visualization of data. In this screen analyst have other optional use cases. First of all on initial view of visualization screen analyst will be able to see overall picture. Analyst will be able to investigate a specific part of the view by using magnifier function. Then analyst will be able to get more detailed information about an IP pair when demanded. After his/her work is finished or want to see overall picture

again, analyst can reset visualization and start analysing from the beginning. Also analyst can take pictures whenever he/she wants. And also analyst will be able to change values of axes.

5.2 Chapter Summary

Main usage scenarios of the software are explained in this chapter. A use case diagram is used for this reason and also the diagram explained briefly.

6 Design

6.1 Class Design Diagram

Design class diagrams show software's classes, their attributes, methods and relations among them. They provide a static view of software design structure.

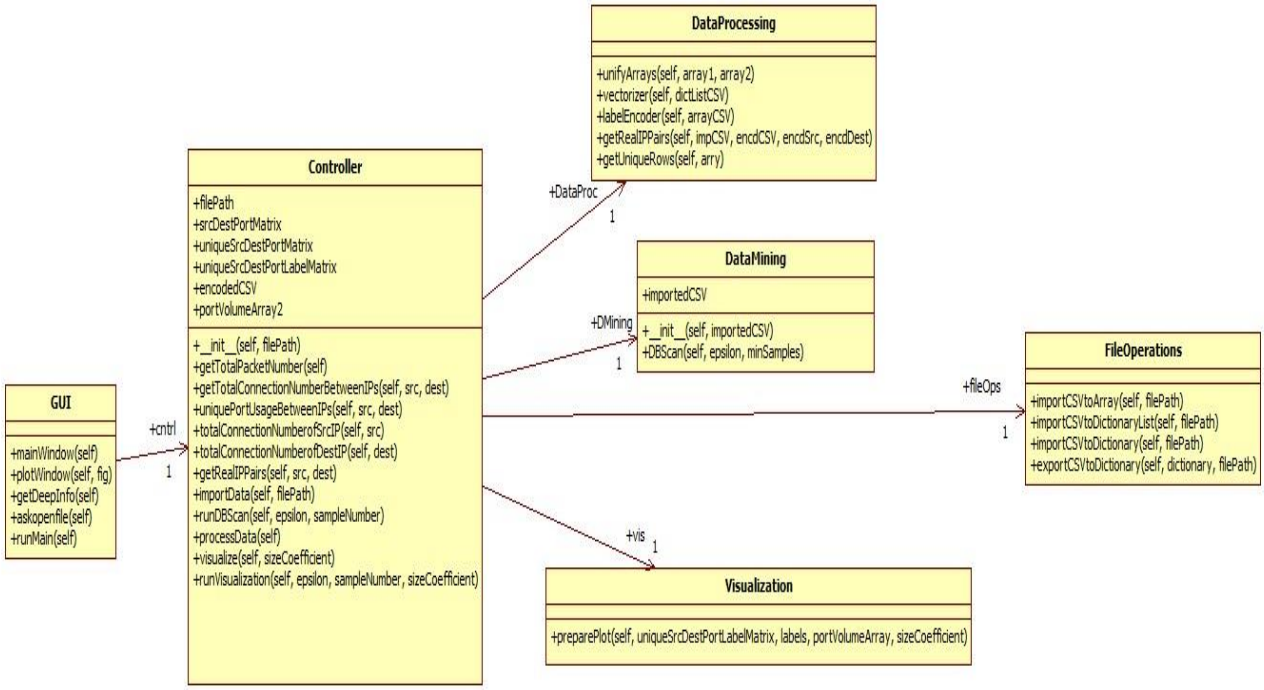


Figure 6. Cyber Security Visualization System Design Class Diagram

This product is designed with four layer architecture. Layers are; user interface layer, application logic layer, domain layer and persistence layer.

There is one class in user interface layer which is named GUI. GUI class is responsible of taking commands from user and sending messages from the system to user. It has only one object to send messages to Controller class. Thus UI layer only communicates with Controller.

Controller class is a facade controller which is representing all application logic. Controller class is a gate between UI Layer and domain layer. Thanks to Controller, UI Layer classes can be changed easily. It is reducing dependence of the software to GUI.

Third layer is domain layer. This layer classes are DataProcessing, DataMining and Visualization. These classes are only used by Controller class which is in above layer.

Last layer is persistence layer. In this software there is one class in persistence layer which is responsible by file operations. In layered architectures there are two approaches: strict layered and relaxed layered architectures. In strict layered architectures every layer can only communicate with layer directly below it. But in relaxed layered architectures a layer can communicate with every layer below it (Larman 2004). In this design it is obvious that a relaxed layer is used. Controller class which is an application layer class sends messages to both domain layer and persistence layer classes. These layers are below application layer.

While examining diagram one point must be taken into account. This design class diagram belongs to a software implemented with python. For that reason methods do not have return types.

6.2 Sequence Diagram

Sequence diagrams show messages between software objects. These diagrams give a dynamic picture of the design.

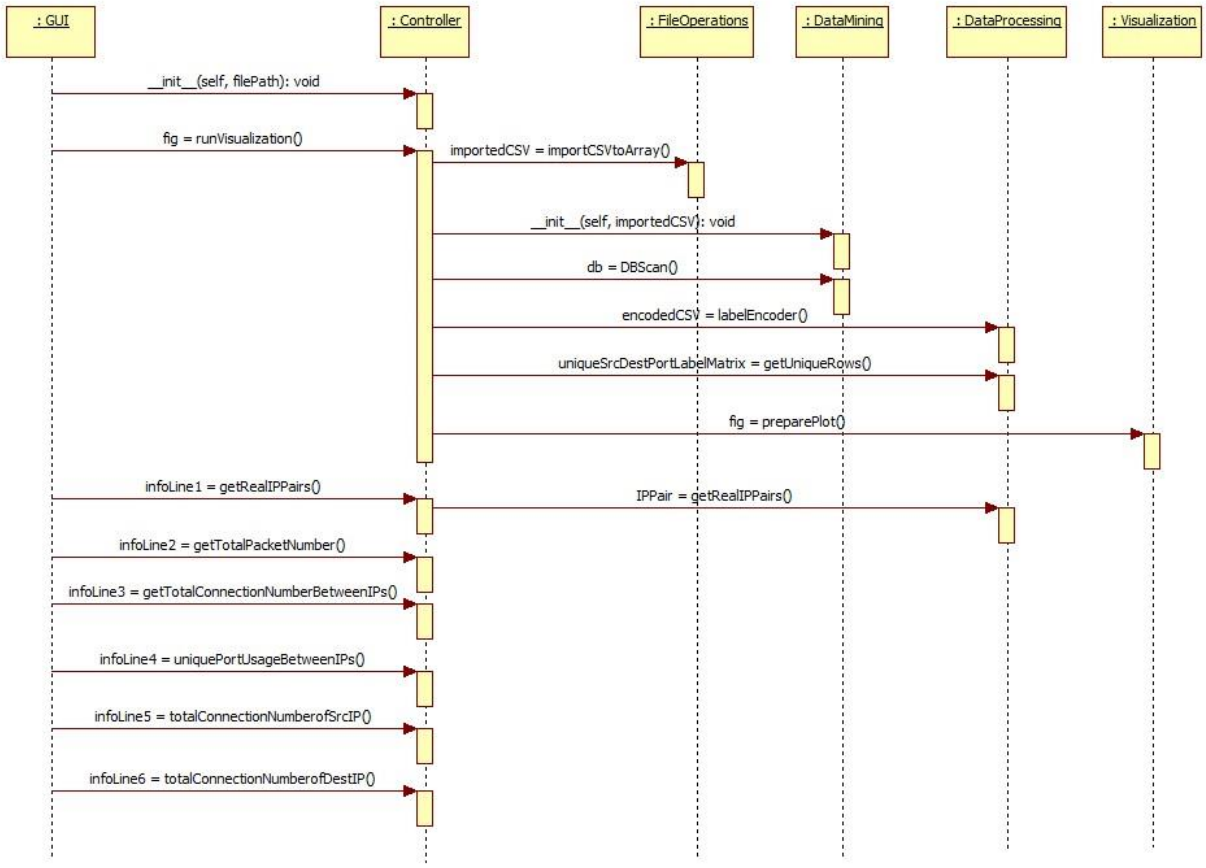


Figure 7. Sequence Diagram of Cyber Security Visualization System

In this software every message is triggered by GUI objects. First of all GUI assigning initial values of a controller object with __init__() method. Then sending runVisualization() message to Controller and after that message, Controller object is sending messages to FileOperations, DataMining, DataProcessing and Visualization objects which are generated by Controller. GUI object also sends six other messages to Controller. One of these messages is triggering another message that is sent to DataProcessing object, while others are handled by only Controller objects.

6.3 User Interface

While designing user interface especially functional requirements which are covered by this report are considered. Main idea was providing enough flexibility to analysts. Also Shneiderman's information visualization mantra is used as a guideline.

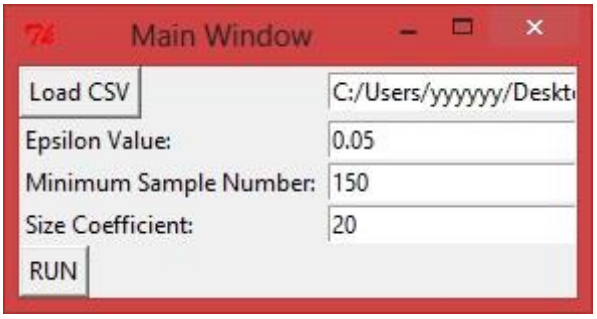


Figure 8. Main Window of Cyber Security Visualization System

After running the software a simple main window appears. In this window epsilon value, minimum sample number and size coefficient values have default values. A CSV file's path can be defined by pushing "Load CSV" button also. After that simple configuration is completed visualization can be worked by pushing "RUN" button.

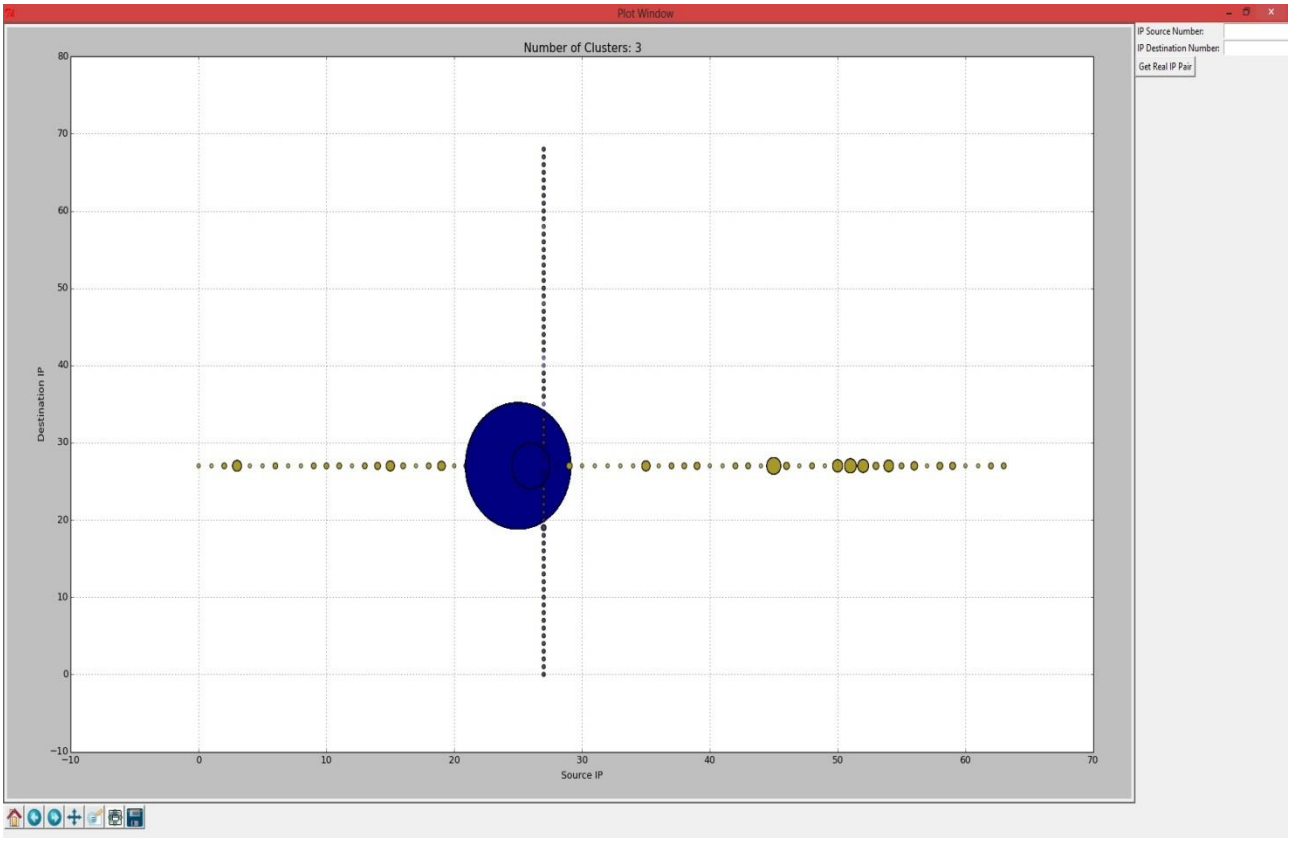


Figure 9. Visualization Window of Cyber Security Visualization System

After a while a new visualization screen appears without closing main window. This is a useful feature because analyst can visualize different datasets or same dataset with different configurations and correlate them easily. A scatter plot with size and colour visual attributes is used in visualization. X axe shows source IP numbers and Y axe shows destination IP numbers.

Instead of mapping real IP addresses to axes relative numbers are used. It is built in feature of matplotlib library. This feature prevents label occlusion.

Every circle shows at least one connection between corresponding IPs. Even if there is one connection or one million connections between two IPs there will be one circle. With this method occlusion numbers of circles have been reduced. Also circles have transparency which is another feature to reduce occlusion level. Size of circles shows how many different ports are used by an IP pair during whole connection time.

Number of clusters is written top of the plot. Colour attribute is used to encode clusters. In this example there are blue, brown and red circles. Clustering is a novel approach which is used to attract analyst's attention to different parts of the graphic.

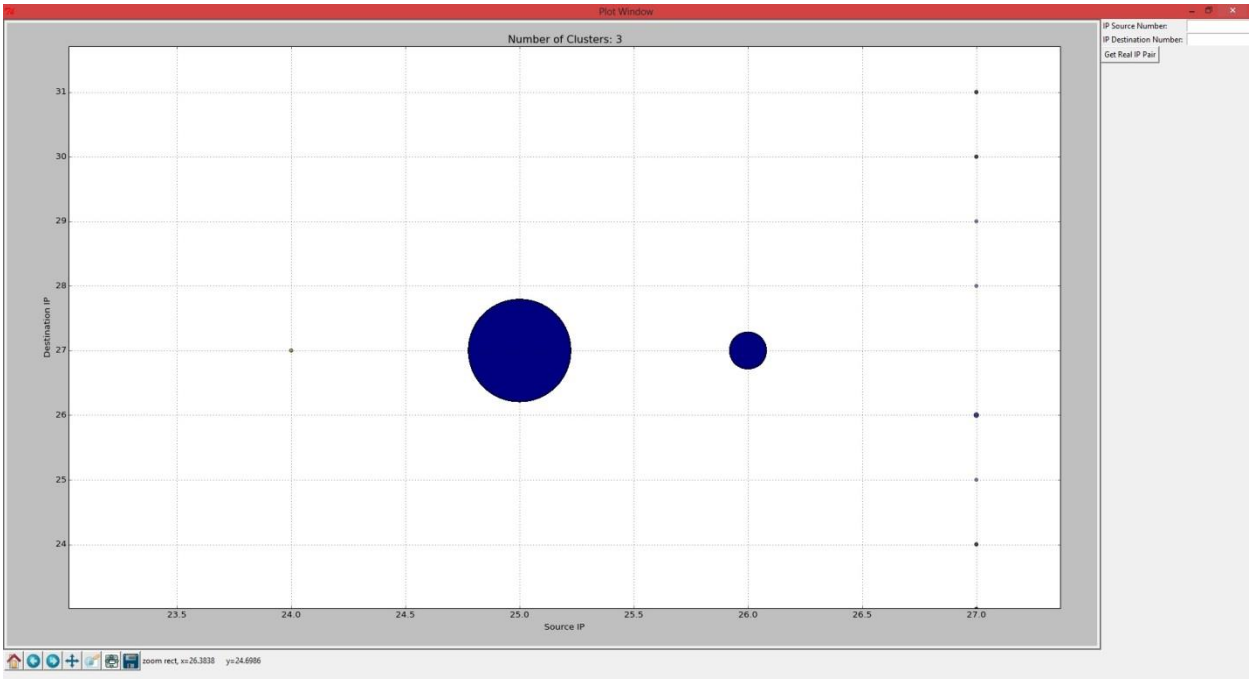


Figure 10. Zoomed View

Analyst can focus on a specific part of the plot by using magnifier which is in a toolbar below of the screen (Figure 10).

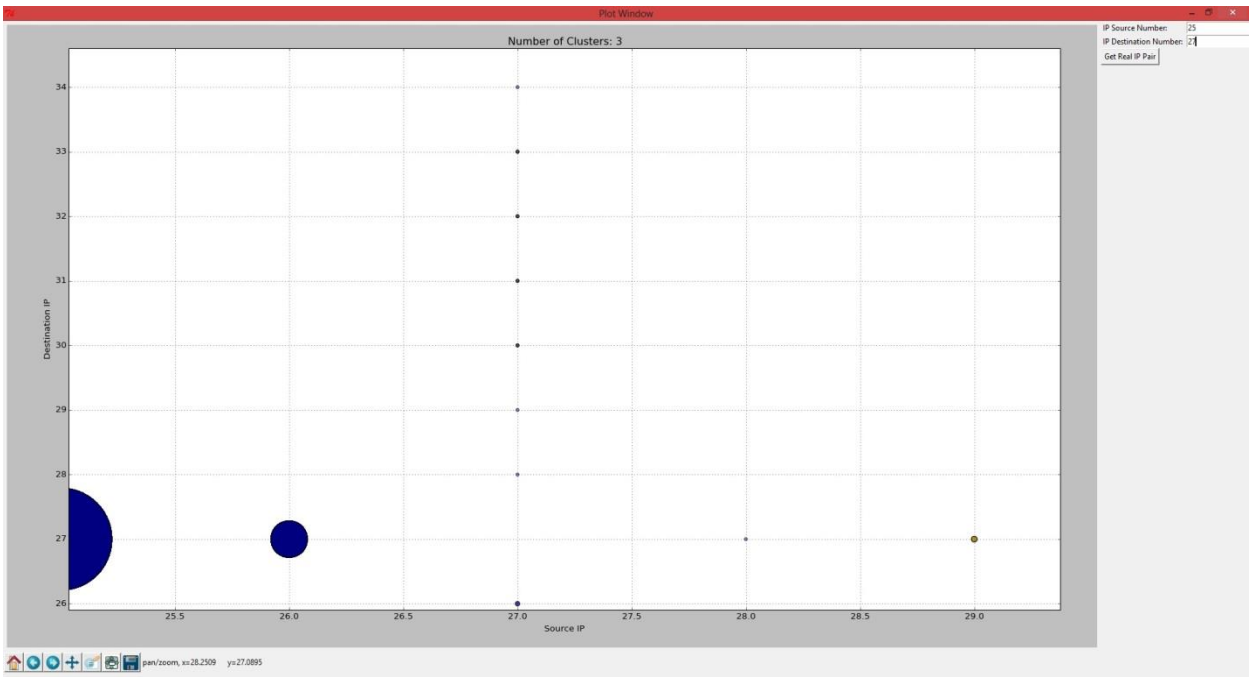


Figure 11. Changed Coordinates of the View

Analyst can change the coordinates of the zoomed view easily by pushing to corresponding button in the toolbar. In figure 11 righter part of the plot is examining.

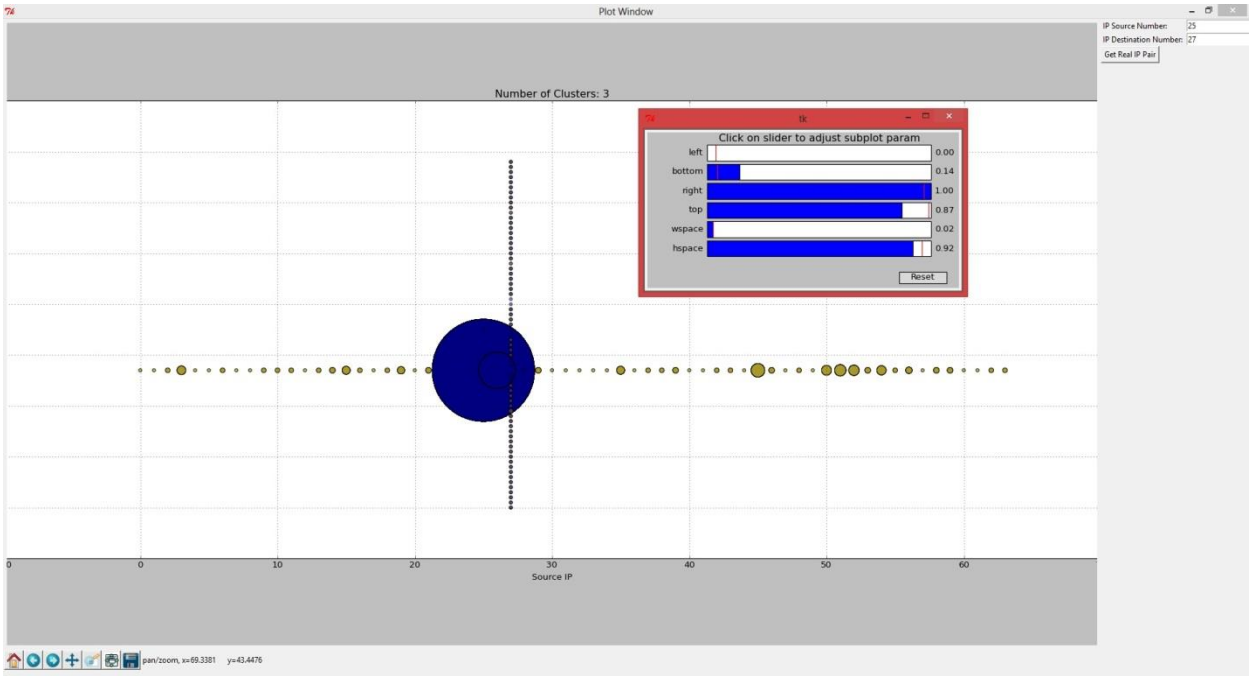


Figure 12. Axe Adjustment

Also some axe adjustments can be carried out in a window.

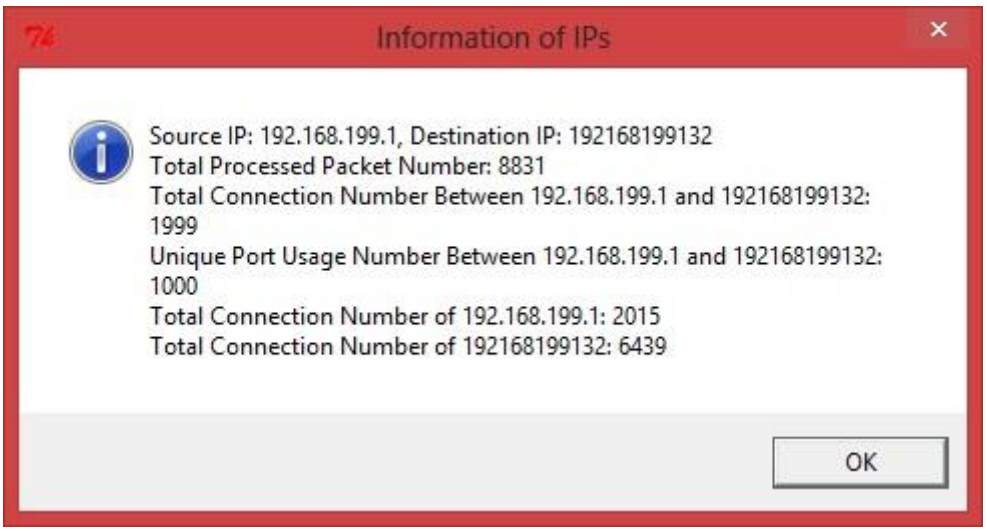


Figure 13. Detailed Information of an IP Pair

Finally after investigation if analyst wants to have detailed information about an IP pair, analyst should enter relative integer values of IP pairs to “IP Source Number” and “IP Destination Number” areas righter side of visualization screen. After pushing “Get Real IP Pair” button a message box appears. In this message box analyst can see real IP addresses and other important statistical information.

6.4 Chapter Summary

Firstly object oriented design of the software is explained in this chapter. To do this, two different diagrams are used. First diagram shows static and second one shows dynamic view of the software design. Then interface of the software is introduced by indicating corresponding requirements.

7 Implementation

7.1 Implementation Process of the Software

In this project prototyping is used as process model as mentioned in methodology chapter. But before implementing prototypes two proof of concept (POC) phases are carried out, because of need of examining both visualization methods and data mining algorithms. For this purpose a free data mining tool, RapidMiner is used. RapidMiner provides an easy to learn and easy to use environment which are obviously crucial at POC phase. RapidMiner has various data mining and visualization methods with easy data pre-processing features. Also algorithms of Weka which is another well-known data mining tool can be added to RapidMiner with a plugin.

After examining data mining and visualization methods second POC phase is carried out. Purpose of this phase was examining corresponding libraries of Python which was going to be language of the final product. Python is a high level programming language with high readability and many useful libraries. After examining python libraries, determined to use these libraries:

- **NumPy:** NumPy is a library that provides a multidimensional array object and ability to operate basic linear algebra on it.
- **scikit-learn:** A data mining and machine learning library for Python. It has various algorithms, besides interoperation ability with NumPy.
- **matplotlib:** matplotlib is a plotting library for Python. It can use NumPy arrays.
- **Tkinter:** This module is the de-facto standard Python interface to the Tk GUI toolkit.

Besides them an IDE is used in implementation process. Microsoft Visual Studio Express 2012 is preferred in this point because this IDE provides high usability with its IntelliSense feature. Also debug operations and project management are quite simple in this IDE. Other important thing is that express edition of Microsoft Visual Studio is free. Despite Visual Studio supports various programming language built-in, Python is not one of them. For that reason Python Tools for Visual Studio (PTVS) is used to provide Python support in Visual Studio.

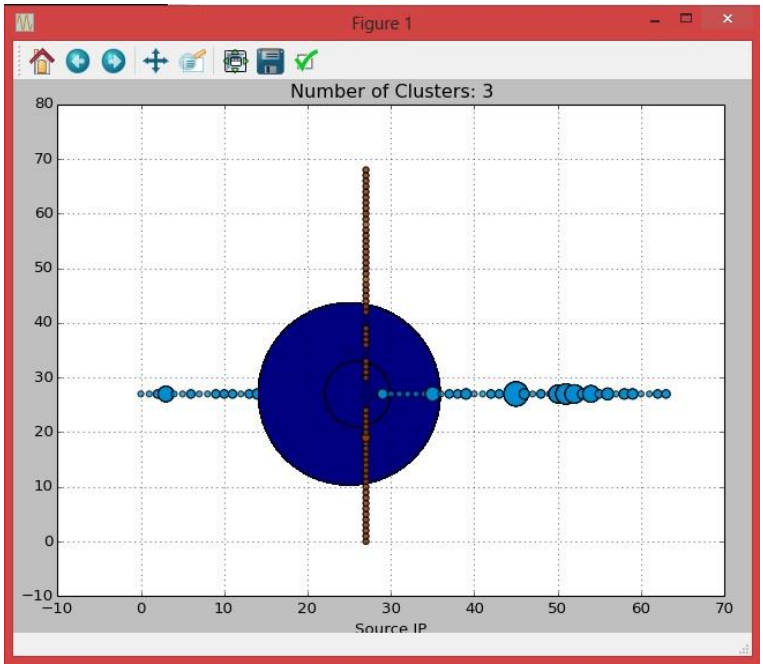


Figure 14. Prototype of the System

After POC phases development of the software is started. In prototyping it is not certain that how many prototypes will be implemented in the beginning of the project. In this project after one throw away prototyping, final product is implemented. The prototype had same data mining and visualization features with final product. But it was working with unchangeable CSV, epsilon, minimum sample and size coefficient values. It did not have ability to show detailed information of IP pairs. The prototype was working in one window and using matplotlib's plotting feature instead of Tkinter. Also class design is revised after the prototype.

Final product is a data mining aided cyber security visualization tool. It uses packet captures in CSV format that have following columns respectively: source IP, destination port, destination IP, length, windows size, sequence NO and TTL. Values on axes represent source IP and Destination IP. Size of the circles represents number of unique destination port usage between IPs. Conventional information visualization methods are used so far. After this point length, windows size, sequence NO and TTL values which are numerical values of IP packet headers are used to apply data mining. Results of data mining process are encoded with colour attribute.

For data mining process a clustering algorithm named DBScan is preferred because of three main reasons. First reason is that DBScan does not need to define numbers of clusters. Since analysts usually does not know how many clusters will be in a dataset before examining it, this feature of DBSCAN makes it suitable for this project alone.

Also there are only two parameters to define, in this algorithm: number of samples and epsilon values. Number of samples value is for determining that how many samples a cluster must have minimum. Epsilon value shows the distance between two samples to be considered as in the same cluster. This increases usability of this algorithm by cyber security analysts because it is not realistic to wait them to have deep information about data mining. Therefore providing them an easy to use algorithm is crucial.

Lastly DBSCAN is very robust at detecting outliers. Clusters are especially good at attacks that have enough connection numbers to detect like port scanning, or DOS. But attacks that have very limited connections show outlier behaviours in most cases. Therefore outlier detection feature of DBSCAN is another reason to use it as data mining algorithm.

First remarkable thing in this project is that seven different attributes of network trace is visualized in a scatter plot in this system. Second point is that a data mining algorithm is used to assist analyser. Data mining algorithms in cyber security is used in many projects. But they do everything automatically which is reducing reliability of these systems. But in this project data mining is used only as an advice tool which is also appropriate for information visualization theory.

7.2 Chapter Summary

Development process of the software is covered in this chapter. Tools, languages and platforms which are used in this project are introduced and justified. Also the data mining algorithm is explained besides why and how used it is.

8 Testing

There are some datasets to evaluate this kind of projects on the internet. However these datasets are not used in this project. Some of datasets have big amount of data which is not suitable for this project. Because conducting data mining process on big amount of data needs high hardware requirements. Also because of the data mining algorithm that is used in this study, attack data must form a small percentage of whole data. But these datasets include many types of attack.

On the other hand some datasets on the internet have small size and include one attack type in it. However they are not used in testing phase also, because that kind of datasets only include attack data without normal traffic and it is not appropriate for this tool again.

Since it is hard to find suitable datasets, they are generated. All datasets have attack traffic besides normal traffic which forms higher percentage of it. Sizes of datasets are small enough to apply data mining algorithms in a regular computer. Traffic is captured by victim machine with Wireshark (Figure 15).

Last dataset is taken from Laura Chappell's Wireshark University DVD. This dataset only includes a password cracking traffic. Therefore after exporting this traffic to a CSV file a normal traffic is added and also IP addresses are changed.

The tool is used over three different datasets. First dataset includes a port scan activity, second one includes a DOS attack and last dataset includes a password crack attempt. Attack types are determined based on two factors: importance of attack type and easiness to produce the traffic.

Port scanning is initial point of the attacks in most of the time. If an analyst can detect port scanning activity, he/she can figure out which hosts and services are under risk and take measures to prevent (Simon et al. 2006). For that reason it is crucial to evaluate performance of this tool on port scanning activities. And it is easy to generate a network traffic that includes port scanning data.

DOS and password based attacks have high importance also. These attack types are two of the most encountered attack types (Microsoft 2014). And generating dataset for these attack types is easy also.

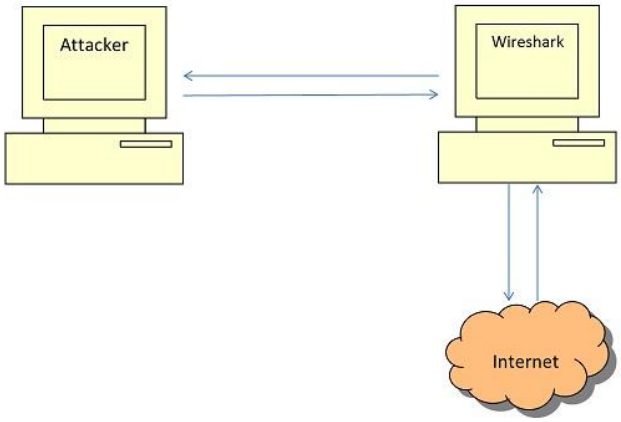
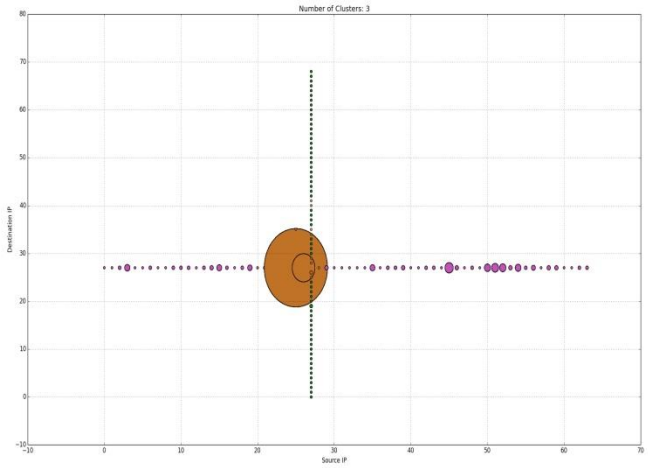


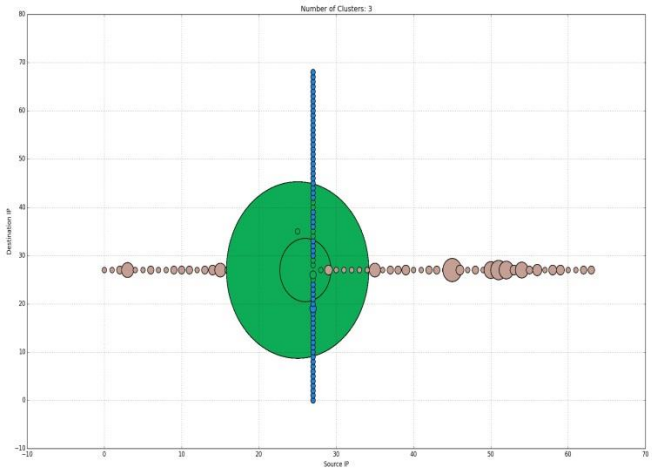
Figure 15. Test Environment

8.1 Port Scanning

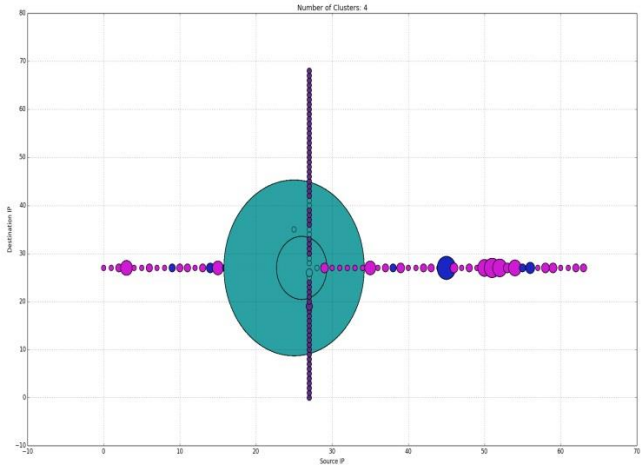
Figure 16 shows visualization window when the tool worked with different parameters. Every graphic has a different colour scheme because of automatic colour assignment. When started with default values to visualization (shown in Figure 16(a)) there are 3 different clusters with small sizes except a big circle in the middle of the graphic. When setting a bigger value for size coefficient value (shown in Figure 16(b)) small circles are becoming more visible again with 3 clusters. If minimum sample number value is become 120, cluster number appears as 4 (shown in Figure 16(c)). And finally if analyst makes epsilon value 1 with other default values cluster number becomes 4 again (shown in Figure 16(d)).



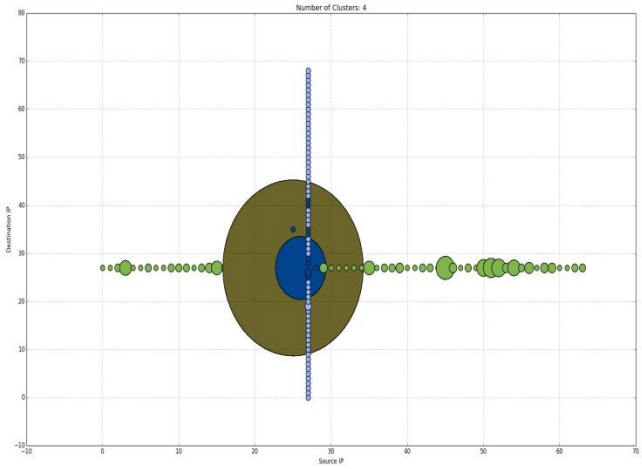
(a) epsilon value: 0.05, minimum sample number: 150, size coefficient: 20



(b) epsilon value: 0.05, minimum sample number: 150, size coefficient: 100



(c) epsilon value: 0.05, minimum sample number: 120, size coefficient: 100



(d) epsilon value: 1, minimum sample number: 150, size coefficient: 100

Figure 16. Visualization window with different values

If analyst goes thorough values in Figure 16 he probably zooms in big circle with a different colour in the middle of the graphic which is obviously the most remarkable thing in the graphic. Because both data mining result (with colours) and unique port usage information (with size) show that there is a different incident then other places. Since data mining function uses different attributes than destination port, it is really important to see both size and colour attributes show unique behaviours in same places. In zoomed view of the graphic (shown in Figure 17) it can be seen that source IP number is 25 and destination IP number is 27.

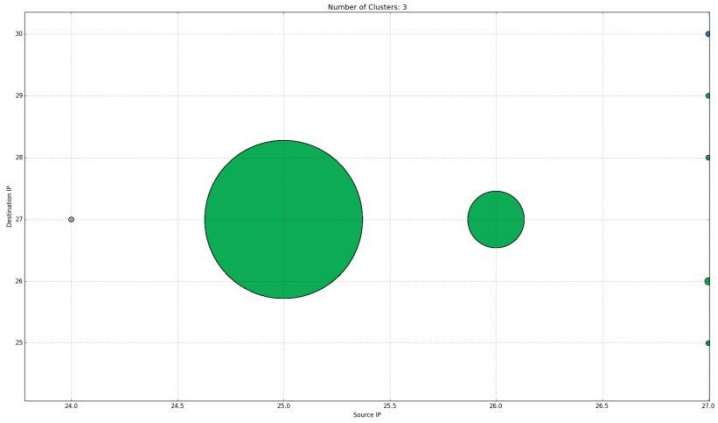


Figure 17. Zoomed visualization window

After investigating zoomed view if the analyst enters corresponding values to “IP Source Number” and “IP Destination Number” areas real IP values with some statistical information appears. According to this window IP addresses are 192.168.199.1 and 192.168.199.132. Total packet number in whole dataset is 8831 and 1999 packets belong to these two IPs. Thus they have almost quarter of all packets. Also these two IPs use 1000 unique ports in 1999 packets. Based on this analysis, it can be said there is a port scanning activity.

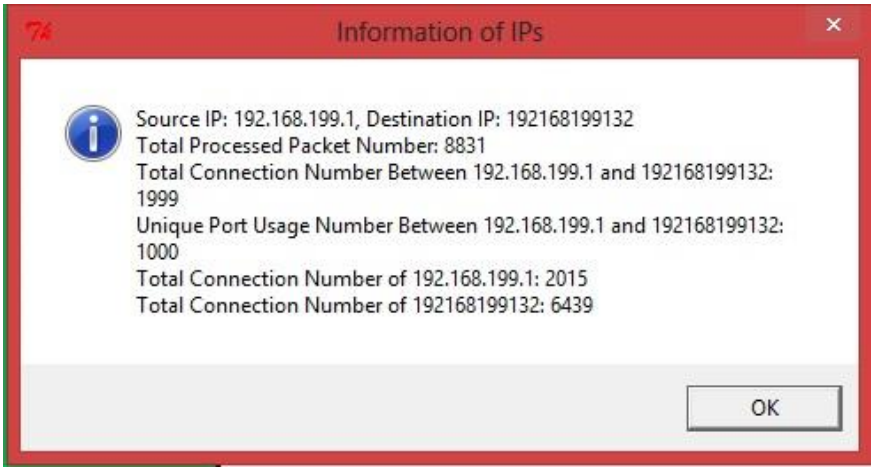


Figure 18. Information of IPs

If these two IPs investigated in Wireshark port scan activity can be spotted. A picture from Wireshark which shows part of activities of these two IPs can be seen in figure 19.

No.	Time	src port	Source	dest port	Destination	Protocol	Length	flags	windows size	sequence no	TTL
1864	135.399938	42754	192.168.199.1	1864	192.168.199.132	TCP	60	0x0002	1024	0	
1864	135.502662	42755	192.168.199.1	1864	192.168.199.132	TCP	60	0x0002	1024	0	
9917	135.502664	42755	192.168.199.1	9917	192.168.199.132	TCP	60	0x0002	1024	0	
1057	135.502665	42755	192.168.199.1	1057	192.168.199.132	TCP	60	0x0002	1024	0	
1022	135.502666	42755	192.168.199.1	1022	192.168.199.132	TCP	60	0x0002	1024	0	
3920	135.502666	42755	192.168.199.1	3920	192.168.199.132	TCP	60	0x0002	1024	0	
8333	135.502727	42755	192.168.199.1	8333	192.168.199.132	TCP	60	0x0002	1024	0	
2809	135.502728	42755	192.168.199.1	2809	192.168.199.132	TCP	60	0x0002	1024	0	
1084	135.502728	42755	192.168.199.1	1084	192.168.199.132	TCP	60	0x0002	1024	0	
8000	135.502729	42755	192.168.199.1	8000	192.168.199.132	TCP	60	0x0002	1024	0	
5004	135.502729	42755	192.168.199.1	5004	192.168.199.132	TCP	60	0x0002	1024	0	
631	135.603516	42754	192.168.199.1	631	192.168.199.132	TCP	60	0x0002	1024	0	
49176	135.603517	42754	192.168.199.1	49176	192.168.199.132	TCP	60	0x0002	1024	0	
1461	135.603518	42754	192.168.199.1	1461	192.168.199.132	TCP	60	0x0002	1024	0	
416	135.603519	42754	192.168.199.1	416	192.168.199.132	TCP	60	0x0002	1024	0	
3333	135.603519	42754	192.168.199.1	3333	192.168.199.132	TCP	60	0x0002	1024	0	
5054	135.603520	42754	192.168.199.1	5054	192.168.199.132	TCP	60	0x0002	1024	0	
3784	135.603521	42754	192.168.199.1	3784	192.168.199.132	TCP	60	0x0002	1024	0	
16993	135.603522	42754	192.168.199.1	16993	192.168.199.132	TCP	60	0x0002	1024	0	
5666	135.603522	42754	192.168.199.1	5666	192.168.199.132	TCP	60	0x0002	1024	0	
2910	135.603523	42754	192.168.199.1	2910	192.168.199.132	TCP	60	0x0002	1024	0	
2910	135.704291	42755	192.168.199.1	2910	192.168.199.132	TCP	60	0x0002	1024	0	
5666	135.704293	42755	192.168.199.1	5666	192.168.199.132	TCP	60	0x0002	1024	0	
16993	135.704336	42755	192.168.199.1	16993	192.168.199.132	TCP	60	0x0002	1024	0	
3784	135.704337	42755	192.168.199.1	3784	192.168.199.132	TCP	60	0x0002	1024	0	
5054	135.704337	42755	192.168.199.1	5054	192.168.199.132	TCP	60	0x0002	1024	0	
3333	135.704338	42755	192.168.199.1	3333	192.168.199.132	TCP	60	0x0002	1024	0	

Figure 19. Wireshark Proof

8.2 DOS Attack

Visualization In figure 20 is produced with a dataset which includes UDP flood attack besides normal traffic. In this case following parameters are used: epsilon value is 0.05, minimum sample number is 150 and size coefficient is 100. With these parameters the tool calculated 3 clusters. When initial visualization window is examined it can be seen that there are 2 different colours are visible: blue and yellow. Also there is a black horizontal line. Due to big number of connection, occlusion causes them to seem black and real colour is not clear.

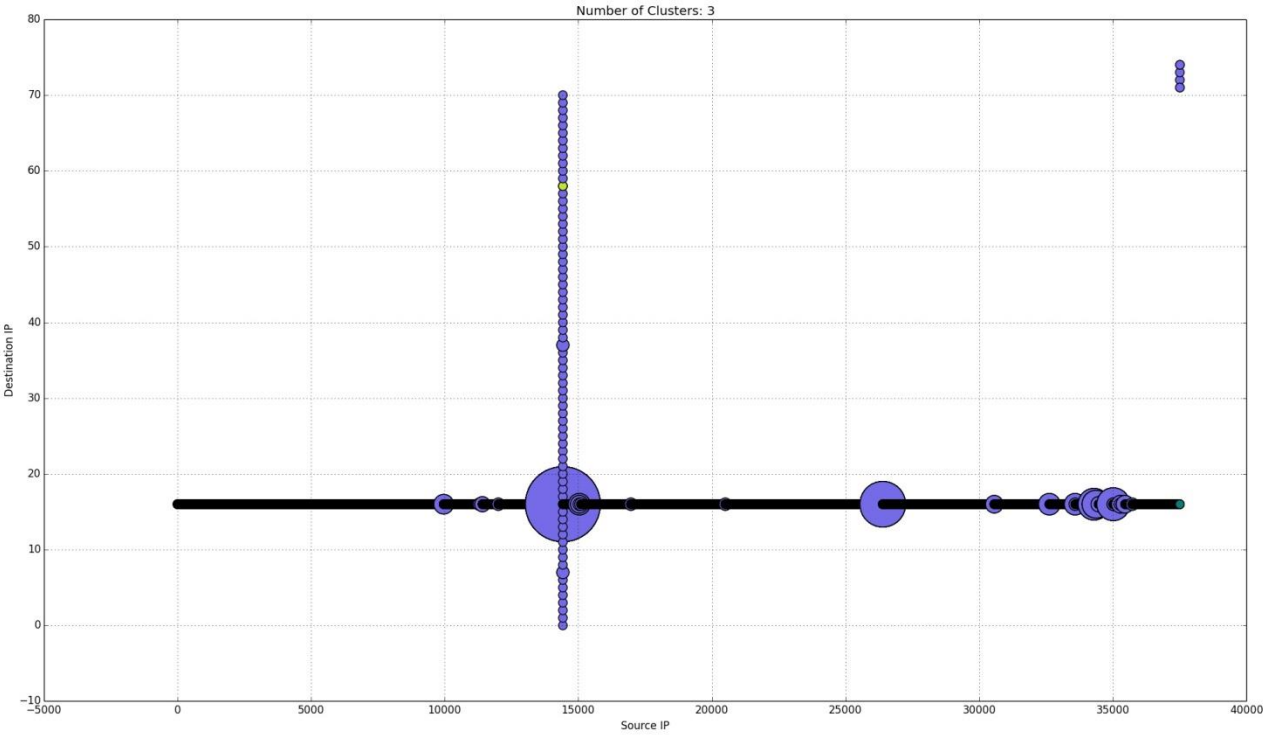


Figure 20. Initial visualization window

When zoomed in to black area it can be seen that the colour is green. So these horizontal series are a different cluster.

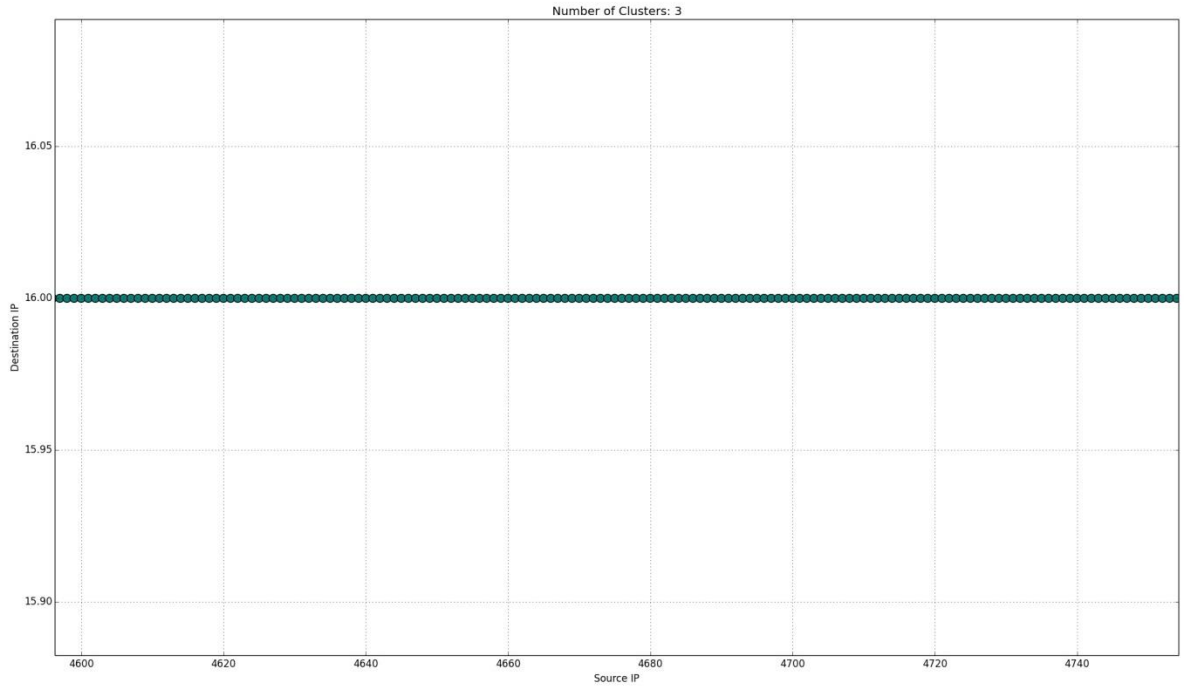


Figure 21. Zoomed view

If IP addresses are entered to “IP Source Number” and “IP Destination Number” areas detailed information can be seen (Figure 22). After this point analyst can carry out a detailed investigation with manual way.

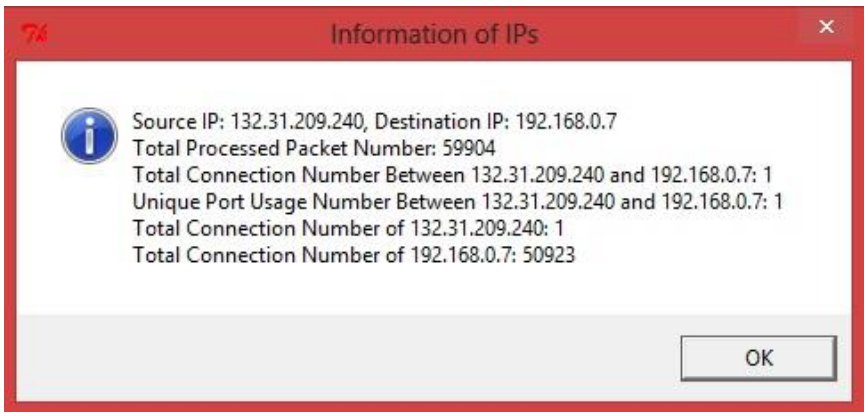


Figure 22. Information of IPs

If the analyst finds one of the suspected source IPs in Wireshark UDP flood activity can be detected easily (Figure 23).

No.	Time	src port	Source	dest port	Destination	Protocol	Length	flags	windows size	sequence no	TTL
	0 76.488843	12607	175.96.129.19	0	192.168.0.7	UDP	60				
	0 76.488936	12608	7.201.52.27	0	192.168.0.7	UDP	60				
	0 76.489166	12609	226.96.140.188	0	192.168.0.7	UDP	60				
	0 76.489345	12610	39.84.253.156	0	192.168.0.7	UDP	60				
	0 76.489494	12611	175.96.244.219	0	192.168.0.7	UDP	60				
	0 76.489602	12612	244.176.144.243	0	192.168.0.7	UDP	60				
	0 76.489645	12613	5.106.22.203	0	192.168.0.7	UDP	60				
	0 76.489742	12614	7.156.61.101	0	192.168.0.7	UDP	60				
	0 76.490203	12615	5.57.198.75	0	192.168.0.7	UDP	60				
	0 76.490448	12616	3.253.89.184	0	192.168.0.7	UDP	60				
	0 76.490693	12617	101.85.246.76	0	192.168.0.7	UDP	60				
	0 76.490946	12618	52.205.187.218	0	192.168.0.7	UDP	60				
	0 76.491210	12619	95.128.7.240	0	192.168.0.7	UDP	60				
	0 76.491466	12620	123.88.247.45	0	192.168.0.7	UDP	60				
	0 76.491627	12621	132.31.209.240	0	192.168.0.7	UDP	60				
	0 76.491775	12622	221.5.244.76	0	192.168.0.7	UDP	60				
	0 76.491873	12623	77.26.167.15	0	192.168.0.7	UDP	60				
	0 76.491969	12624	164.101.251.155	0	192.168.0.7	UDP	60				
	0 76.492113	12625	246.156.57.122	0	192.168.0.7	UDP	60				
	0 76.492325	12626	26.189.5.99	0	192.168.0.7	UDP	60				
	0 76.492474	12627	55.137.158.96	0	192.168.0.7	UDP	60				
	0 76.492583	12628	132.118.255.33	0	192.168.0.7	UDP	60				
	0 76.492674	12629	184.96.243.5	0	192.168.0.7	UDP	60				
	0 76.492764	12630	5.212.56.146	0	192.168.0.7	UDP	60				
	0 76.492847	12631	201.7.156.27	0	192.168.0.7	UDP	60				
	0 76.492991	12632	237.155.39.111	0	192.168.0.7	UDP	60				
	0 76.493019	12633	212.156.76.249	0	192.168.0.7	UDP	60				

Figure 23. Wireshark Proof

8.3 Password Attack

Graphic in figure 24 is obtained when epsilon value is 0.05, minimum sample number is 450 and size coefficient is 100. There are three different colours and one of these colours is in only one circle with a big size. Thus again both size and colour visual attributes are showing same direction.

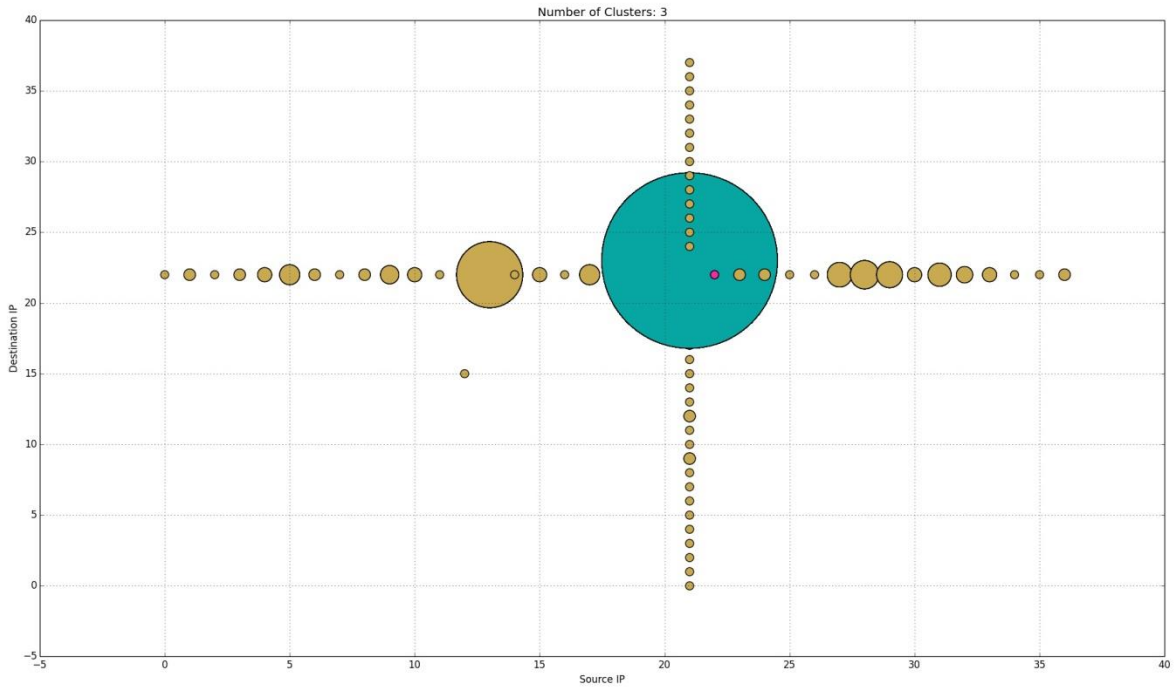


Figure 24. Initial visualization

When approaching to big green circle with magnifier certain IP numbers appear (figure 25).

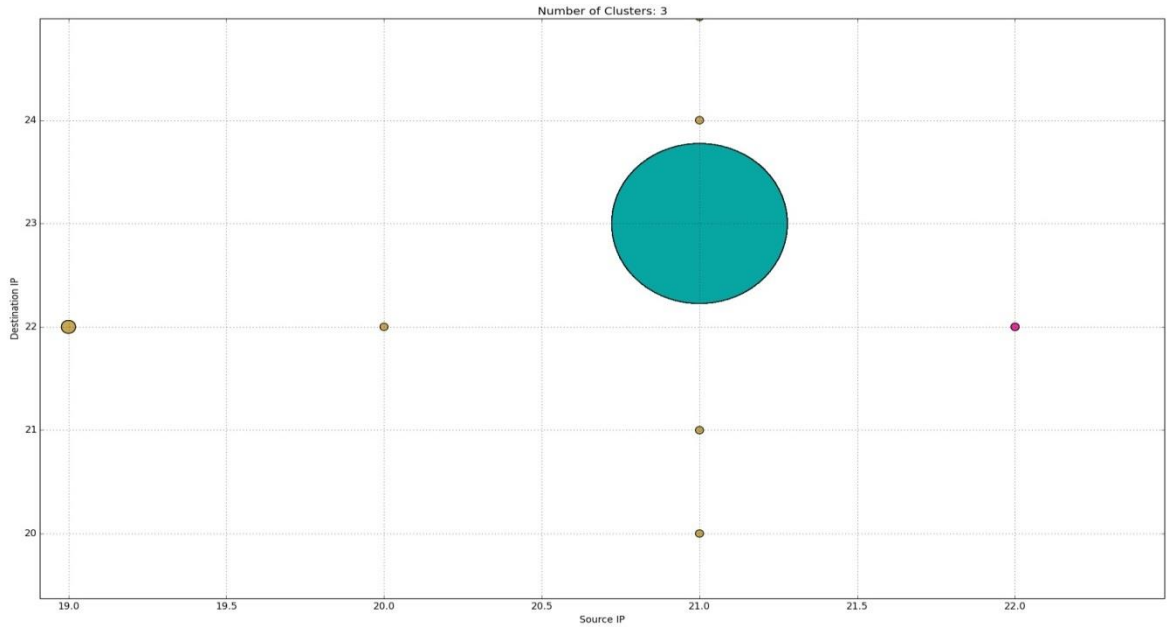


Figure 25. Zoomed View

When IP numbers are entered to system information window appears (figure 26). Most remarkable thing in this information is that 69.181.135.56 has 1755 connection and all of them with 69.181.135.46.

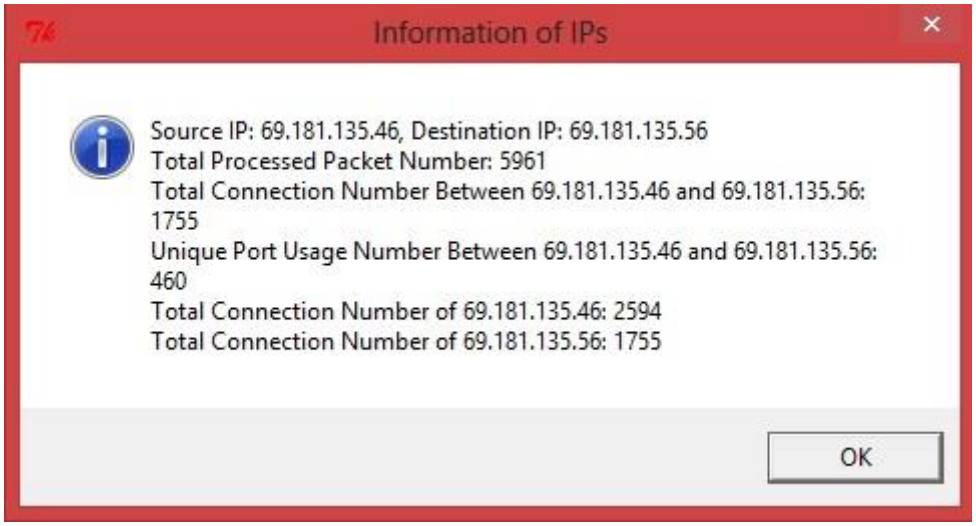


Figure 26. Detailed information

When connections between 69.181.135.56 and 69.181.135.46 are investigated in Wireshark a password attack which is conducted by 69.181.135.56 can be seen (figure 27).

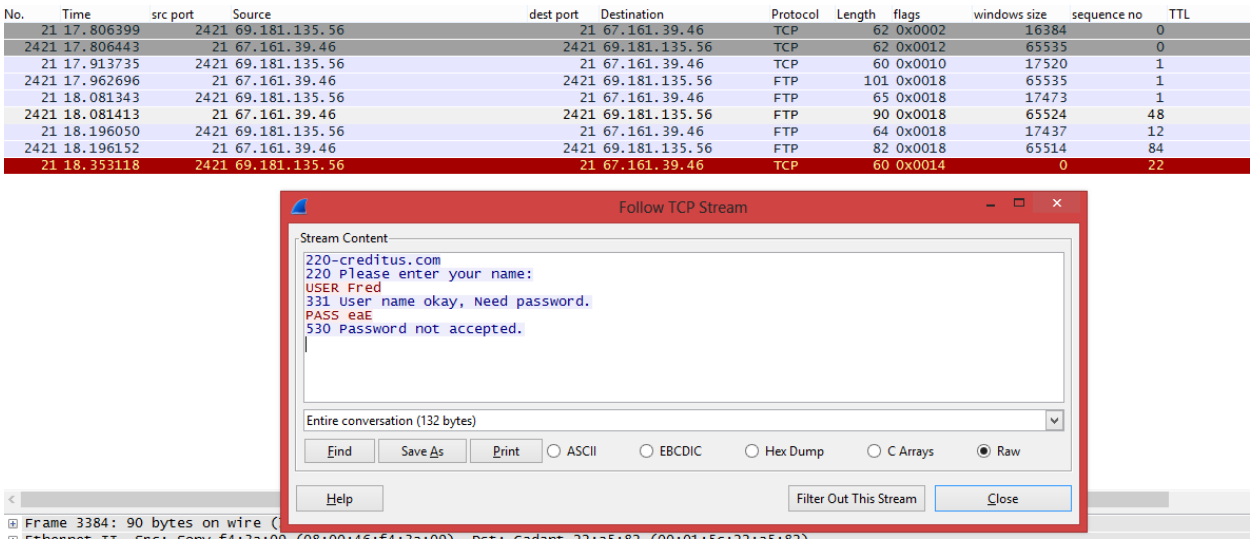


Figure 27. Wireshark Proof

8.4 Chapter Summary

In this chapter firstly it is explained why datasets are generated instead of using datasets on the internet. Then the tools' performance is evaluated over three scenarios after explaining why these scenarios are preferred.

9 Project Management

9.1 Project Schedule

Project management plan is built on two main phases. One phase's aim is gaining enough background for the project and writing first three chapters of the project. Second phase's aim is development of the software and writing rest of the report. Gantt chart for the project schedule is shown in figure 28.

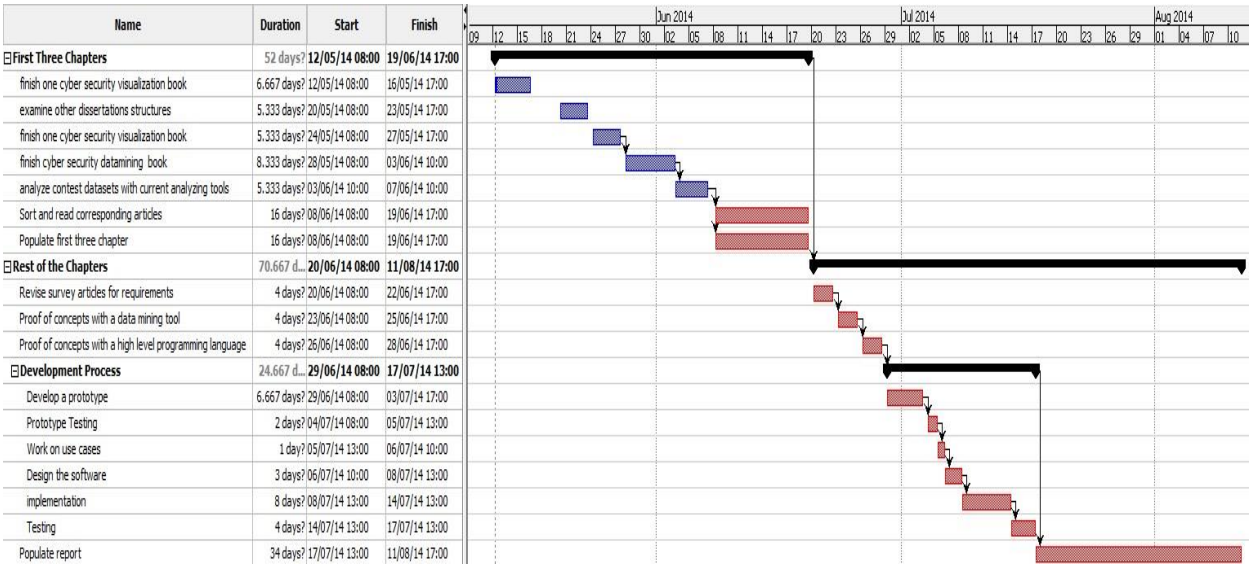


Figure 28 – Project Gantt chart

At the beginning of the project, assigned more time then needed to last task (populate report task) to have time to fix unpredicted problems and find area to change project plan easily. Since first main phase took much more time than predicted (3 more weeks) first main phase was adjusted and "Populate report" task is reduced to more reasonable time.

Also another problem was finding appropriate datasets for the project. Dataset searching process took more time than expected and determined to generate datasets eventually. Dataset generating process caused to adjust the time table also.

9.2 Risk Management

Risk management involves four phases. They are; Risk identification, risk analysis, risk planning and risk monitoring (Sommerville 2010). After finishing of risk monitoring risk analysis phase is started again. This cycle continues during software development process.

Risk analysis phase is carried out after risk identification phase. Purpose in this phase is considering risks. It is a good method to prepare a table in risk analysis phase. Probability and seriousness of the risks must be decided in this phase. Risks, probabilities and effects of this project are shown in figure 29. Impact values are low, moderate, serious and catastrophic.

Risk	Probability	Effects
Underestimated development time	High	Low
Underestimated hardware requirements	Moderate	Moderate
Wrong development environment choosing	Moderate	Serious
The approach may not work	Low	Serious
Project scope can be changed	High	Low
Developer of the project can be ill in critical stages	Low	Moderate
Difficulty to find appropriate datasets	Moderate	Moderate

Figure 29. Risk analysis table

In risk planning phase, project manager makes plan to take action for all individual risk, for both future and present. These strategies can aim to avoiding the risk, reducing the affects or coping with the risk if it is occurred. Figure 30 shows risks and strategies

Risk	Strategy
Underestimated development time	Prepare project plan to have time for this purpose
Underestimated hardware requirements	Ask university to provide hardware
Wrong development environment choosing	Prepare project plan to have time to change development environment
The main approach may not work	Prepare a detailed documentation to tell what is the problem with the approach
Project scope can be change	Evaluate studies in the areas in the beginning of the project to have time to make changes
Developer of the project can be ill in critical stages	Prepare project plan to have time to tolerate it
Difficulty to find appropriate datasets	Produce datasets

Figure 30. Risks and strategies

10 Critical Appraisal

In this chapter a critical appraisal of the project objectives will be conducted.

- Research cyber security visualization theory and tools

In the beginning of the project aim was developing a cyber security visualization tool which uses data mining. It meant that four different major study areas were unified in this project: cyber security, data mining, information visualization and software engineering. Thus the project was started with a research into both information visualization theory and cyber security visualizations due to lack of knowledge in the areas.

First of all two cyber security visualization books were read to have broad knowledge in the area. After this a deeper literature review in academic documents was carried out. Then actual tools were examined. After this study it was understood that what cyber security visualization tools are, what their usage reasons and how an effective tool must be.

- Research network security analysing techniques

To develop a software for cyber security analysts, analysing methods must be known. Therefore behaviours of analysts and their requirements were researched. Also a research is carried out to learn patterns of some well-known attack types. It was curious to integrate data mining into information visualization because important attributes of data and their forms in attack cases must be known to determine data mining method.

- Learn to use a data mining tool and research and compare data mining algorithms to find appropriate one for the project

Third work was a research into data mining. It was started with reading a book which covers data mining and cyber security areas together. Then an academic literature review study was carried out to examine which data mining algorithm was used for which purposes in cyber security area and what their pros and cons were. After this work it was determined to use a clustering method because of its simplicity for users and reliability against changing environment. After this point, a data mining tool (RapidMiner) was learnt and clustering algorithms were examined in this tool which caused to determine to use DBSCAN algorithm.

- Determine most appropriate visual elements to present information in this project

After research to understand whole picture which are mentioned above, one of the most critical phase was started: which plotting technique would be used with which visualization attributes and which attributes of the information would be chose for it. For this work, visualization features of RapidMiner were used besides paper based design. Despite in the beginning of this phase because of the literature review “parallel coordinates” seemed an effective way to reflect information, after experiment it was determined that “parallel coordinates” were using too much system resources during work time. Also it was making hard to interact with visualization display due to slow response time. Therefore it was determined to use scatter plot which is very simple, easy to understand and fast but there was another question: An IP packet header includes many data fields, so which of them will be used?

A 2D scatter plot normally reflects two different data columns with axes. Since source and destination IP addresses are quite important at analysis, axes were used to hold these

attributes. Other useful information in investigating attacks is port number. Instead of writing port numbers directly which can cause a big label occlusion problem in scatter plot, unique port usage numbers were reflected with size attribute. This choice also decreased label occlusion problem in axes because every IP pairs were used only one time.

It was obviously best way to use colour attribute to reflect clusters. But again which IP packet header fields would be used. In this phased focused on two different main options. First one was using visualized fields (source IP, destination IP and port number) for data mining and second one using different fields. Both options could have some advantages but since those three fields were visualized anyway, it was determined to express different data fields with data mining. Thus while analyst carrying out conventional visual analysis, data mining algorithm could give advices based on completely different fields. With that important reason second option was chose.

After then length, windows size, sequence NO and TTL values were chose for data mining. First and the most important reason were that those fields have numerical values and it was an important factor to increase data mining success rate. Secondly in attack pattern research it was realized that most of the malicious traffic's length, windows size, sequence NO and TTL attributes had characteristic values which was another important factor to increase success rate.

- Develop a cyber security visualization tool

And final work was development of the software. Software was developed in estimated time and during implementing Python knowledge was increased. Appraisal of the final product must be conducted in light of requirement analysis. Requirement analysis items are not written again, instead of it only appraisal of them is covered to save words.

1) The tool can point out critical events with its novel clustering approach besides showing an overall picture.
2) Only CSV data format can be used by the tool. It is a negative point in terms of interoperation with other tools.
3) The tool can only use data extracted from packet captures. Thus it does not have ability to conduct correlation between different data types.
4) Data on the screen can be manipulated without changing real data.
5) Reaching real data is not possible with the system.
6) Information visualization mantra is used in the system successfully.
7) DBSCAN algorithm's robust outliner and noise detection features are useful to reduce noise.
8) A 2D scatter plot is used instead of 3D one to reduce occlusion. Also every IP pair is used one time. Magnifier feature is another measure to reduce occlusion problem.
9) Labels are only used at axes. Because of dynamic labelling in axes, the system has no label occlusion problem.
10) Data scaling process is conducted successfully thanks to dynamic labelling in axes.
11) Because of the nature of the clustering algorithms visualization process can take long time. But after finishing this process visualization tool responses quite quickly during analysing process.
12) Since the system has no pre-processing functions importing data is not easy enough. Data exporting way is saving image of the current graphic on the screen which is straight forward.

11 Conclusions

11.1 Achievements

The research and the tool showed that data mining works well with information visualization. Information visualization is a crucial approach at cyber security area but it suffers with several problems like occlusion and data noise. Because of these problems analysing data on a visual display can be hard sometimes. Since information visualization works based on pre-attentive processing, issues that ruin display can break down this process also. As seen in the project before showing data, processing it with data mining can reduce effect of these problems. Because grouping different traffic characteristics with different colours, enhances their readability.

Also shown that, information visualization can be seen as an assistant for automatize cyber security tools. Since actual cyber security tools especially that use data mining are doing everything themselves and just giving results to analysts, they do not seem reliable by analysts. However collaborating of data mining and information visualization forms a new approach. With this approach data mining doing its own job but leaving analysing job to analysts.

Thus this work is a proof of a method which both increases performance of cyber security visualization and increases reliability of data mining based cyber security tools.

11.2 Future Work

Since time was limited and most of the time was spent with research, software development phase was completed in a short time. If there was more time, obviously there would be more prototyping phases for this software. Below are future works to make the tool more professional.

- The system is using only CSV files. CSV format has broad usage but the system also needs capability to use other file formats. Especially using pcap files would be very good.
- A flexible pre-processing module would be very useful for this tool. Currently, data must be processed before importing it to the system.
- The system is using DBSCAN algorithm. But also other algorithms can be useful at different cases. Thus other data mining algorithms can be integrated to the system.
- A design that lets coders to write their algorithms and integrate them to the system easily would make the tool very flexible.
- This tool is not enough for a complete analysing process. After spotting problem another analysing tool like Wireshark must be used. However reaching real data from a visualization system would be very useful which is not provided by most of the visualization tools also.
- A command line version of the tool would increase interoperation.
- In actual system whole data is imported to ram which is not suitable for processing big amount of data. If a data base is used in the system more data can be processed.
- Main idea of this tool is using data mining with visualization which causes a high level of system usage. Since the tool is worked in clients it is limiting data size because of lack of system sources of clients. A new architecture which is worked in a server can increase performance.
- In this tool all clusters are showed in same display with different colours. It was a choice. Also each cluster could be showed in a different display.

12 Student Reflections

Before beginning of the project data mining with cyber security visualization sounds me quite sensible but after my research I did not find any article about exactly what I thought but there were many articles that unifying information visualization and cyber security, cyber security and data mining and information visualization and data mining. Therefore I understood that this project could work but because of the novelty I could encounter with unexpected problems and it could be hard to find solutions to them.

My biggest problem was that this project was involving four big major areas: information visualization, data mining, cyber security and software engineering. Thus I knew research phase would take the biggest part of the whole time. But it took much more time than even my plans. My research phase exceeded about 3 weeks despite my hard working. It was too much time for a three months project. For that reason I worried that if I encountered with a big problem in development phase or if my approach did not work I might be in a big trouble. But works went as almost I expected after this phase.

One of the problems I encountered after research phase was the programming language since I was not familiar enough with Python. But software was developed in estimated time though and I increased my Python knowledge during implementation.

At the beginning of the project I knew almost nothing about information visualization. I can say that especially my knowledge on this area is improved very much. Also this improvement reflected to my objectives. One of my objectives in my proposal was “In visualization module different colours should be used for different kind of data to make them more understandable”. After I conducted a more comprehensive research for the report this objective seemed inappropriate to me. It was being mentioned about only colour. However colour is only one of the visual attributes. For that reason I replaced this objective with more generic one: “Determine most appropriate visual elements to present information in this project”

Also finding a suitable dataset for this study was another problem. I was seeking datasets that biggest parts of them include normal traffic and small parts of them include malicious traffic. But what I found were old and quite big datasets that have many attacks in it or too small datasets that have only malicious traffic. I waste too much time to be sure about that there were not any suitable dataset. I should have given up early for saving my time. Finally I generate my own dataset which took extra time.

Bibliography and References

Ball, R., Fink, G. A., & North, C. (2004) Home-Centric Visualization of Network Traffic for Security Administration. *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, (pp. 55-64)

Barbará, D., Couto, J., Jajodia, S., & Wu, N. (2001) ADAM: a testbed for exploring the use of data mining in intrusion detection. *ACM SIGMOD Record, 30*(4), 15 - 24

Bell, D. (2005) *Software Engineering for Students A Programming Approach.* Addison-Wesley.

Berthold, M., & Hall, L. (2003) Visualizing fuzzy points in parallel coordinates. *Fuzzy Systems, IEEE Transactions on, 11*(3), 369 - 374

Ciampa, M. (2011) *Security+ Guide to Network Security Fundamentals.* CENGAGE Learning Custom Publishing

Conti, G. (2007) *Security Data Visualization: Graphical Techniques for Network Analysis: Graphical Techniques for Rapid Network and Security Analysis* (1 ed.). No Starch Press

Conti, G., Ahamad, M., & Stasko, J. (2005) Attacking information visualization system usability overloading and deceiving the human. *Proceedings of the 2005 symposium on Usable privacy and security - SOUPS '05*, 89-100

Conti, G., Grizzard, J., Ahamad, M., & Owen, H. (2005) Visual exploration of malicious network objects using semantic zoom, interactive encoding and dynamic queries. *Visualization for Computer Security*, (pp. 83-90)

Du, X. (2011) *Data Mining and Machine Learning in Cybersecurity.* Auerbach Publications

Ertoz, L., Eilertson, E., Lazarevic, A., Tan, P., Srivastava, J., Kumar, V., et al. (2004). MINDS - Minnesota Intrusion Detection System. *Data Mining: Next Generation Challenges and Future Directions*

Fayyad, U. M., Piatetsky-Shapiro, G., & Smyth, P. (1996) From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, pp. 37-54

Fink, G. A., L. North, C., Endert, A., & Rose, S. (2009) Visualizing cyber security: Usable workspaces. *2009 6th International Workshop on Visualization for Cyber Security*, (pp. 45-56)

Gershon, N., Stephen, E. G., & Card, S. (1998, March/April) Information visualization. *interactions* , pp. 9-15

Goodall, J., Lutters, W., Rheingans, P., & Komlodi, A. (2005) Preserving the big picture: visual network traffic analysis with TNV. IEEE Workshop on Visualization for Computer Security, (pp. 47 - 54)

Gudadhe, M., Prasad, P., & Kapil, W. L. (2010) A new data mining based network Intrusion Detection model. *2010 International Conference on Computer and Communication Technology (ICCCT)* (pp. 731 - 735). IEEE

Gyanchandani, M., J.L.Rana, & R.N.Yadav. (2012) Taxonomy of Anomaly Based Intrusion Detection System: A Review. *Neural Networks, 2*(12), 1-13

Healey, C. G. (2009) Perception in Visualization [online] available from <http://www.csc.ncsu.edu/faculty/healey/PP/> [16 August 2014]

Healey, C. G., & Enns, J. T. (2012) Attention and Visual Memory in Visualization and Computer Graphics. *IEEE Transactions on Visualization and Computer Graphics, 18*(7), 1170-1188

Horng, S.-J., Su, M.-Y., Chen, Y.-H., Kao, T.-W., Chen, R.-J., Lai, J.-L., et al. (2011) A novel intrusion detection system based on hierarchical clustering and support vector machines. *Expert Systems with Applications, 38*(1), 306 - 313

Jiong Zhang, M. Z. (2006) Anomaly Based Network Intrusion Detection with Unsupervised Outlier Detection. *Communications, 2006. ICC '06. IEEE International Conference on* (pp. 2388 - 2393). Istanbul: IEEE

Jones, A., & Sielken, R. (2000) Computer System Intrusion Detection: A Survey. *Computer Science Technical Report*, 1-25

Karthikeyan .K. R, A. I. (2010) Intrusion Detection Tools and Techniques –A Survey. *International Journal of Computer Theory and Engineering, 2*, 901 - 906

Keim, D. A. (2002) Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics, 8*(1), 1-8

Kmeťová, M. (2010) Mária Kmeťová. *DIVAI 2010 - Distance Learning in Applied Informatics*, (pp. 129-134)

Lakkaraju, K., Bearavolu, R., Slagell, A., Yurcik, W., & North, S. (2005) Closing-the-Loop in NVisionIP: Integrating Discovery and Search in Security Visualizations. Visualization for Computer Security, 2005. (VizSEC 05). IEEE Workshop on (pp. 75 - 82). IEEE

larman, c. (2004) *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development* (3 ed.). Prentice Hall

Lau, S. (2004) The Spinning Cube of Potential Doom. *Communications of the ACM, 47*(6), 25-26

Lee, W., & Stolfo, S. J. (2000) A framework for constructing features and models for intrusion detection systems. ACM Transactions on Information and System Security, 3(4), 227 - 261

Liao, Y., & Vemuri, V. (2002) Use of k-nearest neighbor classifier for intrusion detection. *Computers & Security, 21* (5), 439 - 448

Lippmann, R., & Fried, D. (2000) Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation. *DARPA Information, 2*, 12 - 26

Livnat, Y., Agutter, J., Moon, S., Erbacher, R. F., & Foresti, S. (2005) A visualization paradigm for network intrusion detection. Proceedings from the Sixth Annual IEEE Systems, Man and Cybernetics (SMC) Information Assurance Workshop (pp. 92 - 99). IEEE

Mackinlay, J. (1986) Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics (TOG), 5*(2), 110-141

Maheswari, S., & Jain, D. C. (May 2012) A Comparative Analysis of Different types of Models in Software Development Life Cycle. International Journal of Advanced Research in Computer Science and Software Engineering, 285-290

Marty, R. (2008) *Applied Security Visualization* (1 ed.). Addison-Wesley Professional

McPherson, J., Ma, K.-L., Krystosk, P., Bartoletti, T., & Christensen, M. (2004). PortVis: A Tool for Port-Based Detection of Security Events. Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security, (pp. 73-81).

Microsoft (2014) Common Types of Network Attacks [online] available from <http://technet.microsoft.com/en-us/library/cc959354.aspx> [16 August 2014]

Munassar, N. M., & Govardhan, A. (2010) Comparison Between Five Models Of Software Engineering. International Journal of Computer Science Issues, Vol 7, Issue 5, 94-101

Nesbitt, K. V., & Friedrich, C. (2002) Applying Gestalt principles to animated visualizations of network data. *Proceedings Sixth International Conference on Information Visualisation*, (pp. 737-743)

Patcha, A., & Park, J.-M. (2007) An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer Networks, 51*(12), 3448-3470

Rusu, A., Fabian, A. J., Jianu, R., & Rusu, A. (2011) Using the Gestalt Principle of Closure to Alleviate the Edge Crossing Problem in Graph Drawings. *2011 15th International Conference on Information Visualisation*, (pp. 488-493)

Schriger, D. L., & Cooper, R. J. (2001) Achieving graphical excellence: Suggestions and methods for creating high-quality visual displays of experimental data. *Annals of Emergency Medicine , 37*(1), 75-87

Shiravi, H., Shiravi, A., & Ghorbani, A. A. (2012) A survey of visualization systems for network security. *IEEE transactions on visualization and computer graphics, 18*(8), 1313-1329

Shneiderman, B. (1996) The eyes have it: a task by data type taxonomy for information visualizations. *Proceedings 1996 IEEE Symposium on Visual Languages*, (pp. 336-343)

Simon, G. J., Xiong, H., Eilertson, E., & Kumar, V. (2006) Scan Detection: A Data Mining Approach. *Sixth SIAM International Conference on Data Mining*, (pp. 118-129)

Sommerville, I. (2010) *Software Engineering.* Addison-Wesley

Song, J., Takakura, H., Okabe, Y., & Kwo, Y. (2011) Correlation Analysis Between Honeypot Data and IDS Alerts Using One-class SVM. In P. Skrobanek (Ed.), *Intrusion Detection Systems.* InTech

Stevens, S. S. (1946) On the Theory of Scales of Measurement. *Science,, 103*, 677-680

Ware, C. (2004) *Information Visualization, Second Edition: Perception for Design.* Morgan Kaufmann

Wikipedia (2014) Parallel coordinates [online] available from <http://en.wikipedia.org/wiki/Parallel_coordinates> [16 August 2014]