

Visual Data Exploration Techniques for System Administration

Tam Weng Seng

Abstract

The objective of this paper is to study terminology used in visual data exploration and to apply them to projects in Computer Security. To achieve this objective, four specific examples are examined and the terms are applied to them.

1 Introduction

According to Robert F. Erbacher and Deborah Frincke, “A 1996 CSI-FBI survey found that \$4.5 Billion was lost to business due to compromises in information security” [3]. The number of mutated or new computer viruses, Trojans, worms, and denial of service attacks continues to increase at an alarming rate. New vulnerabilities are constantly being found in widely used commercial products and for various reasons many systems are not fixed with workarounds or updated with the latest patches. In light of this, it is important for system administrators to have a good overall situational awareness of the environment to be maintained.

Unfortunately, the main source of information available to a system administrator, through log files, is subject to at least three problems. Firstly, information in log files are mainly recorded or viewed as plain text scattered across several different files [8]. In addition, “The use of text is limiting since reading textual information is inherently a perceptually serial process. Interpretation of graphical images, on the other hand, is perceptually a parallel process.” [4] The next problem is the size of the log files. A third problem is the difficulty in analyzing these log files. Although some log files have different categories for the different types of messages, they are not usually very flexible and it may not be possible to create new categories [8]. Ultimately, all these factors contribute to information overload. Fortunately, information overload is not a problem unique to the field of System and Network Administration. Visual data mining or exploration techniques can be helpful in addressing this problem. Thus, the purpose of this paper is to explore some of the currently available literature on visual data exploration along with proposals that use these techniques for the purpose of System or Network Administration.

2 Taxonomy of Terms

To begin, a paper by Ben Shneiderman proposes that the “Visual Information Seeking Mantra” is “Overview first, zoom and filter, then details on demand” [7]. In this paper, the problem of visual data exploration is broken down into the

following seven tasks – overview, zoom, filter, details-on-demand, relate, history, and extract [7].

The first task is to gain an overview of the available information. This can be followed by filtering out unwanted information or focusing in on by specific information by zooming in on an item of interest. When specific items of interests have been located, it might be useful to be able to select items or groups of items to obtain additional information. While having details of specific items are useful, showing relationships between items in a data set or similar items in different data sets are also useful. For example, a user might want all items with similar properties to be highlighted. With the above tasks a history of steps taken during exploration process could allow users to undo, refine, replay or combine the steps taken. Last but not least, the ability to extract specific subsets of the data to allow for other uses such as printing, presentation in other tools or programs. The ability to save, send or print the settings used in the visualization process might also be useful [7].

Data Types

Subsequently, both Shneiderman and Keim propose classifying data into different data types. The following is an attempt to combine the two classifications:

- **Multi-dimensional:** A source of data with more than three attributes [5]. An example could be geographic maps with the depth and grade of crude oil deposits. Multidimensional data is also defined as items with n-attributes which become points in an n-dimensional space [7].
- **Temporal Data:** Data sets with time as an attribute. Additionally, temporal data should have a start and an end time. Items may also overlap [7]. Although temporal data could be considered a special case of multi-dimensional data with time as an attribute, the ability to directly represent time using animation as a visualization technique creates an argument for a separate category.
- **Three-Dimensional:** A source of data with three distinct variables, such as buildings and the human body [7]. While this could be considered a special case of multi-dimensional data when n equals three, three dimensional data can be mapped directly into width, length and height. This means that additional visualization techniques could be available that might not apply to data of higher dimensionality without the loss of information.
- **Two-Dimensional:** A source of data with 2 distinct variables, such as geographic maps and newspaper layouts (Shneiderman, 2) [5]. As with Three-dimensional data, a similar argument applies here.
- **One-Dimensional:** Shneiderman defines this as a linear source of data including plain text files, such as source code, alphabetical ordered lists of things [7]. Meanwhile, Keim defines one-dimensional data as “temporal data” and “text and hypertext” is placed under a separate category [5]. Unfortunately, this is not consistent with the definition of multidimensional data used by Shneiderman, which is said to contain n-attributes [7].

Temporal data, by definition, usually contains at least one attribute over time, which makes a second attribute. This can be seen in the example, given by Keim, of several stock prices over time which consists of three attributes - The name of the stock, the price of the stock and time.

- Networks or graphs: Items with relationships linked to an arbitrary number of other items. As items in a network are related to each other, it might be useful to examine the shortest or least costly path between two nodes or to travel the entire network [7].
- Tree: Hierarchies or trees are a special case of networks, or graphs, that do not have loops. In addition, each item only has one parent, except the root. As trees, unlike networks or graphs, cannot have loops and only one parent per item, it might make sense to examine how many layers a tree has from the root, how many children are related to each item [7].

To conclude the discussion on possible data types, it should be noted that the types might not be mutually exclusive and data exploration techniques might use a combination of these data types [7].

Visualization Techniques

With categories for the different types of data, Keim also proposes categories for the different visualization techniques along with interaction and distortion techniques to manipulate the visualizations. Firstly, the categories of visualization techniques could be - standard 2D/3D displays, geometrically transformed displays, icon-based displays, dense pixel displays, and stacked displays [5].

- The standard 2D/3D displays include x-y or x-y-z plots, bar charts, line graphs and pie charts.[5]
- Geometrically transformed displays are aimed at transforming multidimensional data and include exploratory statistics, such as scatterplot matrices. Other geometric projection technique include Prosecution Views, Hyper-slice, and Parallel Coordinates [5].
- Iconic displays are the use of symbols, or icons, to map an attribute of a multidimensional data set to an attribute of the icon. These icons might include faces, sticks, colour icons and geometric shapes [5].
- Dense pixel displays map each dimension to a coloured pixel and group the pixels from each dimension into adjacent areas. In general, dense pixel displays use one pixel per data value allowing large amounts of data to be displayed [5].
- Stacked displays use the concept of embedding one coordinate system within another coordinate system [5]. For example, two attributes from the data set form an outer coordinate system; two other attributes are embedded within in the outer coordinate system, and so on. An example of this is dimensional stacking [6].

The following are categories of interaction and distortion techniques proposed by Keim:

- Dynamic projections are a sequence of projections, which can be driven randomly, manually, precomputed or data driven, to explore a multidimensional data set. An example could be a system that tries to show all interesting two-dimensional projections of a multidimensional data set as a series of scatterplots [5].
- Interactive filtering is similar to the task of “filter” as proposed by Shneiderman. It should allow users partition and focus on interesting subsets of the data. To achieve this, interactive filtering should allow users to restrict the data displayed, or highlight specific subsets of data [5].
- Interactive zooming is similar to the task of “zoom” as proposed by Shneiderman. It should allow users to view the same data at different resolutions. Moreover, interactive zooming is not restricted to enlarging the display of data objects, but also could mean that additional details would be presented at higher levels zoom levels. For example, data might be represented as single pixels on a low zoom level, icons on another level, and as labelled objects at the highest resolution [5].
- Interactive distortion is based on the idea of showing portions of the data with more detail while others are shown with less. Some examples are hyperbolic and spherical distortions [5].
- Interactive linking and brushing is the combination of different visualization methods to overcome the failing of a single technique. For example, different projections of scatterplots built on the same data might be combined by colouring and linking subsets of points in the different projections. By a similar method, linking and brushing can be used by all the visualization techniques described. More specifically, selected points in a display could cause related points in other displays to become highlighted, making it possible to see relationships between the visualizations. Interactive changes made to a visualization should automatically be reflected in the other visualizations [5].

To conclude the discussion on visualization techniques, it should be noted that, like the data types, the categories of visualizations and interactive and distortion techniques are not mutually exclusive. A prototype could combine multiple techniques in a display [5].

3 Specific Examples

With the requirement in mind, and an understanding of some terminology from visual data exploration, the following prototypes can be examined with the idea of applying terminology from visual data exploration.

An Example of Dimension Stacking

To begin, in figure 1, we can see a combination of multiple visualization techniques. The “floor” of the display, is called the “host grid” and is a combination of a stacked and iconic display. With the hosts under monitoring, the outer dimensions of the stacked display are the target host’s location and

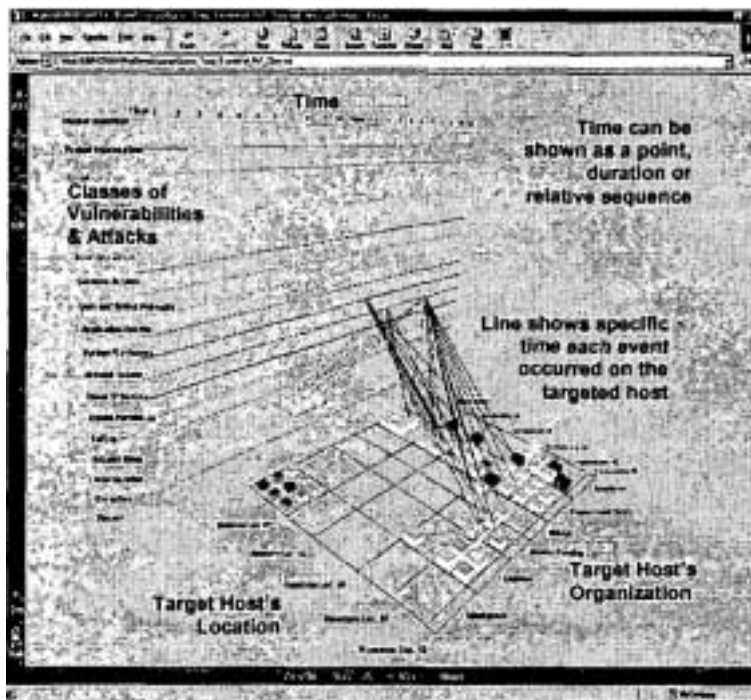


Figure 1: A screen shot of the program using VRML. [1]

organization. The inner coordinates, appear to be an iconic display showing the number of hosts and a null dimension. The different icons, although not explicitly stated in the paper, is most likely the Operating System. If desired, the icons could be used to encode additional information. Subsequently, targeted hosts, that are being monitored, are placed on one side of a dividing “vertical wall” and hosts detected to be attacking a monitored system are placed on the other side. This “vertical wall” is basically a 2-dimensional grid stood on its side. In this example figure 1, the two-dimensional grid has time on the x-axis and class of vulnerabilities and types of attacks on the y-axis. The attackers are linked to the target in time through links in the vertical wall. These lines could be argued to be a form of interactive linking and brushing between the different displays. In doing this, it should be noted that a graph or network data structure has been formed. It is not a tree as each target or attacker could be active at more than one time, this would create multiple connections over time making it impossible to identify a root node [1].

Subsequently, they chose a three-dimensional view so that could rotate the display and zoom in on particular portions of the display. When rotating the composite display to show the “rear floor,” the attackers are brought to the front. This “rear floor” shows is supposed to show information about the attacker, such as internet protocol (IP) address and number of hops. Although the authors of the paper were not very specific, the design of the “rear plane” appears to be the similar as the “front,” except that the outer coordinates are probably number of hops and something related the IP address of the attacker [1].

Another feature of this program is the ability to add another dimension of information in the “vertical” wall. By using the concept of dimension stacking, the frequency of a type of attack over at a particular point in time can be displayed. This frequency is displayed in the form of bar, where the size of the bar

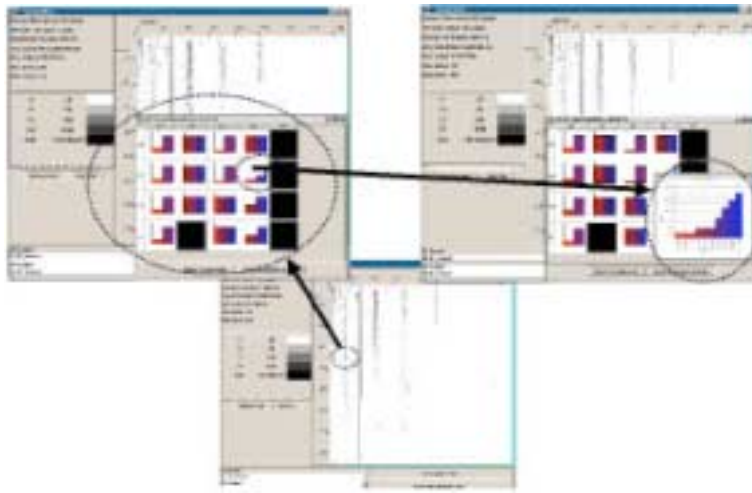


Figure 2: A screen shot of NVisionIP. [9]

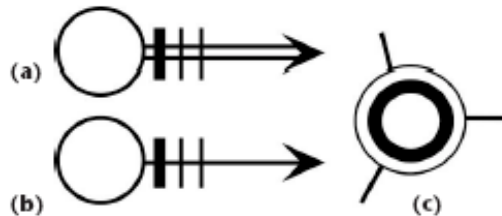


Figure 3: AN iconic display using a glyph metaphor. [4]

indicates the number of attacks. With this bar, it is possible to interactively filter the available data to obtain a clearer view of which target machines are being attacked at a particular time, with the “front view.” Conversely, the “rear view” would show information about the attackers [1].

An Example of a Dense Pixel Display

The next project to be examined is called NVisionIP. The source of information for the displays are “Netflows” from multiple network devices. figure 2 contains three snapshots of the project, emphasizing three different levels of details. From these displays, it is obvious that multiple visualizations techniques are used. At the highest level of detail, a dense pixel display is used. A more detailed, “Small Multiple View,” uses a stacked display is used in conjunction with a standard 2-dimensional chart to provide more details about a range of machines. An even more detailed, “Machine View” is similar to the “Small Multiple View,” except that the information is filtered down to a single IP address [9]. Another difference is that each 2-D chart displays different information, which could be considered a form of dynamic projection [5].

Subsequently, like many other visual exploration tools, this program includes the ability to interactively filter what is displayed. To make it easier for a user, a separate dialogue box with filter options is available at the click of the button. Meanwhile, at the highest level of detail, the dense pixel display can be customized by changing the number of categories or colors used to represent each data point. A user is also able to swap the axis on the dense pixel display [9].

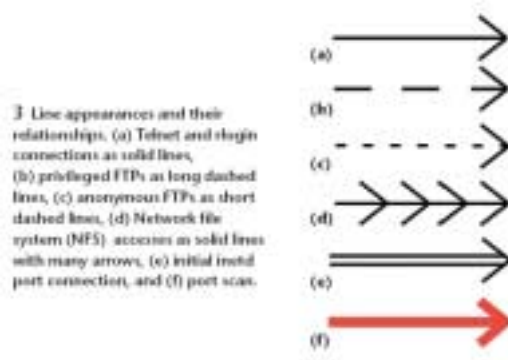


Figure 4: Types of lines used to connect the glyphs. [4]

An example of an Iconic Display in 2-D

This next example is based on glyphs, or icons, to represent data obtained mostly from the Hummer Intrusion detection system, which is stored in a Postgres Database, from servers used in the Computer Science department at the University at Albany-SUNY. Lines connecting the glyphs, encode most of the information about connections to a computer, by the line style, the direction of the arrow, hashes on the lines and the colour of the line. The colour of a node is also used to encode information [4],[3].

With that in mind, FTP, telnet and rlogin connections that have not been authenticated are represented as parallel lines as shown in figure 3a. Once successfully connected, and if necessary authenticated, the parallel lines are replaced with a single line, as shown in figure 3b, where the line style represents of the type of connection (See figure 4). In both cases, the direction of the arrow indicates the direction of the connection. In addition, to the direction of the connection, a single hash mark on the line represents a single user and a thicker hash represents users with multiple connections. Last but not least, red is used to represent unusual or unexpected activity. Yellow is used to represent questionable activity that is non-critical. For example, a node is highlighted as red when a user executes a su or sudo command. A link will turn red when an authentication challenge times out. Red links and nodes are displayed when port-sentry, an intrusion detection tool, detects inappropriate attempts to access system resources; attacks; or a host address that cannot be resolved by a domain name server (DNS). A node will turn yellow, if a NSF mount to the system does not respond. Subsequently, the nodes and lines remain on display until the connection is terminated. At that point, the nodes and lines gradually fade away completely. In doing so, short lived events such as port scan are left on display long enough to be noticed [4].

Meanwhile, the node under investigation is represented by the glyph shown in figure 3c. If the information on the system under investigation is available, spokes sticking out from the perimeter of the glyph represent 10 users on that system. The thickness of the inner circle represents the system load, and the intensity of the glyph represents the time since the node was last accessed. When all access to or from the node have faded off the display, the node will begin to fade until it is removed, unless it is accessed [4].

With this knowledge of the symbols used in this proposal, figure 4a. is an

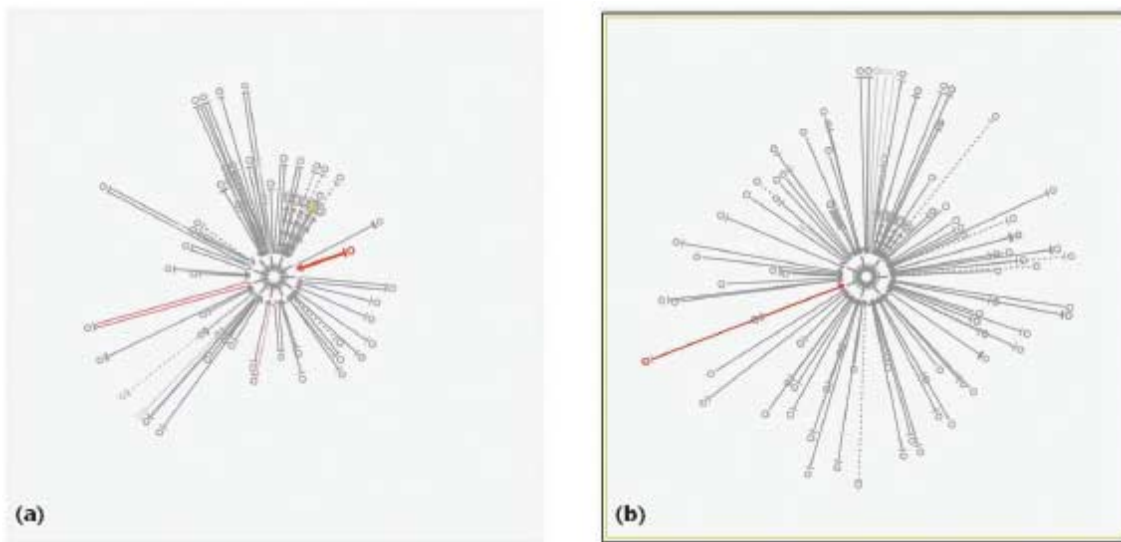


Figure 5: A Snapshot of the glyphs in action. [4]



Figure 6: A Snapshot of the control panel. [4]

example that shows “two connections that failed to authenticate, a port-sentry identified attack, a lost NFS mount, several initiated inetd connections, ftp and telnet connections during the morning” as indicated by the border around the display. Figure 4b is another example where the authors point out “an anomaly at the top where many remote systems are connecting to the server in sequence for short periods” [4]. This snapshot of activity is from the late evening, as indicated by the border [4].

To understand how time is encoded in the border, the border is white at noon and black at midnight. A yellow border is added to indicate that it is afternoon. Therefore, in Figure 5a, the border is white, without the additional yellow border indicating that the snapshot was taken close to noon. In Figure 5b, the black border with the additional yellow border indicates that the snapshot was taken as midnight was approaching [4].

Subsequently, the nodes representing remote systems are slated to be placed

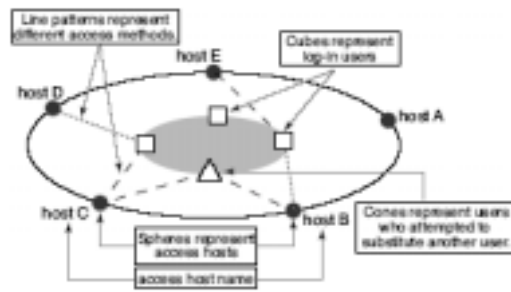


Figure 7: A Layers in Tudumi. [8]

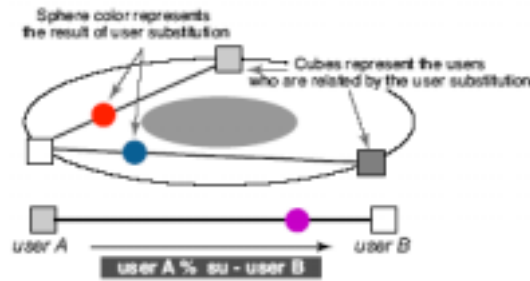


Figure 8: The Bottom layer in Tudumi. [8]

in one of five different rings expanding out from around the system under investigation. The criteria for choosing which ring to place a node is determined by the difference between the IP address of the remote and the local system under investigation. If it is a host in the same local subnet, meaning that the network portion of the IP address is the same as the monitored system, it is placed in the first ring. Remote hosts which, for whatever reason, cannot be resolved by the DNS are placed in the fifth ring. In doing so, the distance of the glyphs from each other gives some idea of the distance of the remote host from the host under investigation. In order to gain some information about activity over time from a particular host, surrounding the monitored system, are made to appear in the same position. As the positions in a particular ring of position fill up with new remote hosts and positions taken by hosts which may have faded away each ring of nodes may expand to include additional layers [4].

Having explained the content of each individual snapshot, Figure 6 is the control panel that allows a user to watch connections to the system under investigation over time. From this control panel, a user can take snapshots at a particular moment. VCR like controls allow a user to control the rate of the animation, increasing, decreasing, stopping and going backwards as needed [4].

An example of an Iconic Display in 3-D

Tudumi is another example of an iconic display, except that it is also a stacked display. The purpose of this program is to keep track of user access to a server and when users substitute themselves as another user. In this way, the two examples of iconic displays visualize similar information. While the type of information visualized maybe similar, this example is a combination of a stacked and iconic display, making it significantly different. By stacking the iconic displays vertically, it becomes a three dimensional display instead of two. In

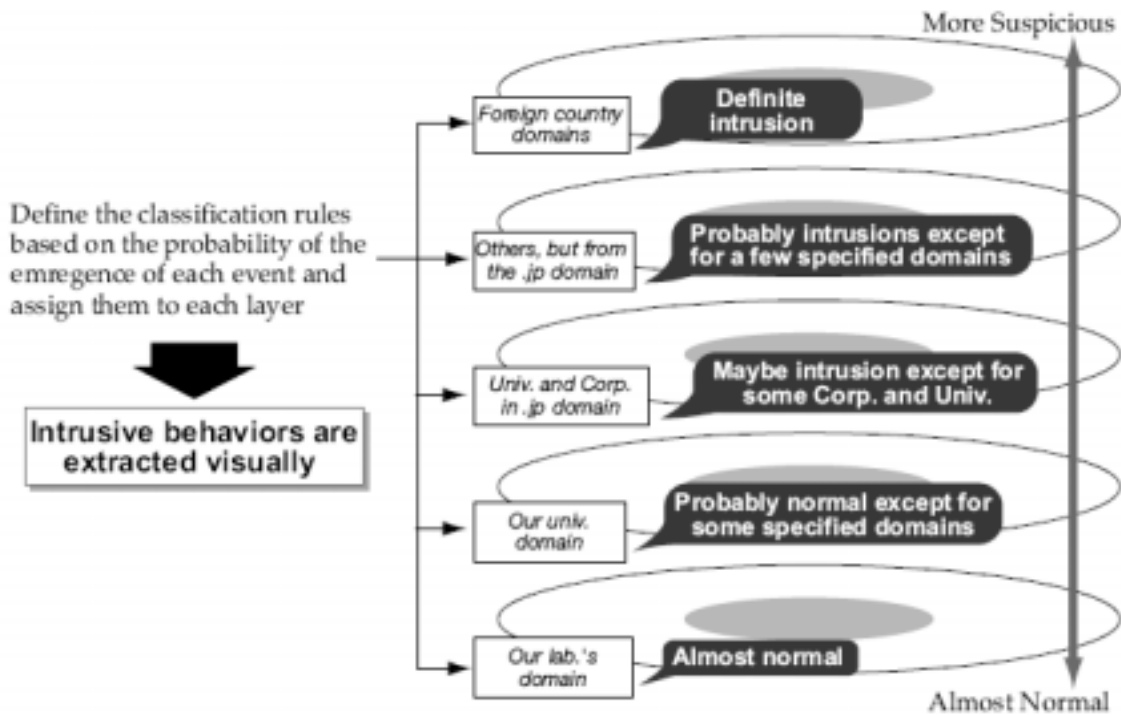


Figure 9: An example of a 3D structure of Tudumi. [8]

addition, it does not explicitly display the monitored host. Figure 7 is an example of a layer or disk in the display, except for the bottom layer which is shown in Figure 8. As this program is partly a stacked display, this program includes a method to select how many layers are to be displayed. It also includes a method to create rules telling the program which layer to place a remote host in the display. An example of this is shown in Figure 9. Another feature of this program is the ability to select a cube icon to interactively filter out all the other icons not connected to that cube. This reduces the clutter that might hide information. Unfortunately, unlike the previous example, it is not clear how this project will cope if the number of hosts or users exceeds what can be displayed in the single ring where they are placed [8].

4 Conclusion

This paper is an example of using terminology from visual data exploration and applying them to projects in computer security to show a relationship between them. The examples also show how these ideas can be applied in a practical manner to computer security and the field of System and Network Administration.

In conclusion, as Shneiderman points out, one purpose of a taxonomy is to encourage discussion and practical discoveries [7]. However, before discussion is possible, knowledge of its existence is necessary. Therefore, another potential advantage of a taxonomy is the ability to improve the efficiency in finding similar projects which use the same approach through commonly used terms. While taxonomies might not be perfect or may become outdated, further papers can renew their usefulness [2].

References

- [1] Anita D'Amico and Mark Larkin. Methods of visualizing temporal patterns in and mission impact of computer security breaches. In *DARPA Information Survivability Conference and Exposition (DISCEX II'01) Volume I-Volume 1*, pages 343–351. IEEE Computer Society, 2001.
- [2] Maria Cristina Ferreira de Oliveira and Haim Levkowitz. From visual data exploration to visual data mining:a survey. *IEEE Transactions on Visualization and Computer Graphics*, 9(3):378–394, jul/sep 2003.
- [3] Robert F. Erbacher and Deborah Frincke. Visualization in detection of intrusions and misuse in large scale networks. In *Proceedings of the International Conference on Information Visualisation*, pages 294–299. IEEE Computer Society, 2000.
- [4] Robert F. Erbacher, Kenneth L. Walker, and Deborah A. Frincke. Intrusion and misuse detection in large-scale systems. *IEEE Computer Graphics and Applications*, 22(1):38–48, jan/feb 2002.
- [5] Daniel A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002.
- [6] Jeffrey LeBlanc, Matthew O. Ward, and Norman Wittels. Exploring n-dimensional databases. In *Proceedings of the 1st conference on Visualization '90*, pages 230–237. IEEE Computer Society Press, 1990.
- [7] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the 1996 IEEE Symposium on Visual Languages*, pages 336–343. IEEE Computer Society, 1996.
- [8] Hideki KOIKE Tetsuji TAKADA. Tudumi: Information visualization system for monitoring and auditing computer logs. In *Proceedings of 6th International Conference on Information Visualization*, pages 570–576. IEEE CS Press, July 2002.
- [9] William Yurcik, James Barlow, Kiran Lakkaraju, and Mike Haberman. Two visual computer network security monitoring tools incorporating operator interface requirements. In *Workshop on Human-Computer Interaction and Security Systems*. ACM, apr 2003.