Intrusion Detection Visualization and Software Architecture For the Detection of Competent Attacks

Dr. Robert F. Erbacher
Assistant Professor
Department of Computer Science

Utah State University
UMC 4205
Logan, UT 84322

Final Report for VFRP

Air Force Research Lab
Information Assurance (IFGB)
Joe Giordano
Rome , NY

August 2004

Intrusion Detection Visualization and Software Architecture for the Detection of Competent Attacks

Dr. Robert F. Erbacher
Assistant Professor
Department of Computer Science
Utah State University

Abstract

The need for visualization-based intrusion detection and analysis techniques has been shown due to the complexity of the underlying data and the inability of purely algorithm techniques, such as data mining to effectively analyze intrusion data. We have developed several visualization mockups which have great potential for aiding in the identification and analysis of intrusions. Our goal with these visualization techniques was to focus on solutions for competent attacks. The goal was to ignore or filter out script kiddies and the like as they can be easily blocked by firewalls and should not consume the analyst's time. These techniques rely on glyph-based techniques which have proven effective for the representation of multi-parametric data. These visualization techniques in conjunction with effective interaction techniques will create a complete environment.

We also provide an initial architecture to guide the future implementation of the environment. As a first step in this implementation a java native interface based wrapper has been created for Simpcap, an engine for accessing large-scale libpcap files. The architecture incorporates support for the tap+bridge interface as well as other database formats. This will provide support for many of the interfaces in use by AFRL and other organizations. The architecture also incorporates an advanced user interface and support for multiple visualization displays.

Multiple visualization displays are critical as no single visualization can show all aspects of a database. Thus, these difference visualization techniques allow for different views of the data, either examining the same data parameters are completely separate ones. This provides for a more complete representation and analysis of the database than is otherwise possible.

Finally, we incorporate a discussion of several other related topics to information awareness. This includes cyber command and control, computer forensics, trending, and intrusion detection. While we believe the developed techniques will prove immensely effective we have begun to identify future directions of research to continue the process of refinement and improvement and engender future systems which provide complete solutions to the problems at hand.

Intrusion Detection Visualization and Software Architecture For the Detection of Competent Attacks

Dr. Robert F. Erbacher

1. Introduction

Intrusion detection has become a critical aspect of computer security. With the value and sensitivity of information maintained on many computer systems, it is critical that this information be protected and that infiltrations be identified quickly before substantial damage is incurred. Additionally, it is just as important to identify competent directed attacks before they can become infiltrations. The idea with such attack detections is to enable preventative measures to prevent such attacks from becoming full blown intrusions. This provides multiple lines of defense, a critical aspect to computer security [15], and ensures that particularly vital systems are protected from compromise. If we consider the sensitivity of mission critical systems, such as power plant controls, command and control systems, missile control systems, and military personnel databases, any intrusion would be unacceptable and pose potentially enormous consequences.

When considering intrusion detection we must consider several modus operandi. First, the traditional external threat in which a potential intruder attacks the network attempting to compromise systems remotely by bypassing the firewall or through other identified weaknesses either in the network itself or within individual computer systems connected to the network. We must also consider the traditional social engineering approaches in which an attacker gains insider knowledge or direct access through manipulation of individuals with valid access to the systems and network. A simplified example of such an attack is merely phoning an employee, pretending to be technical support, and asking the employee for their username and password.

The final category of intrusion detection is that of insider threat. Insider threat itself consists of multiple threats, generally consisting of attacks originating from within the perimeter network. The social engineering approach can in fact be considered one such insider threat, after access is gained to a computer system from the application of the acquired knowledge. Such insider threats can consist of:

1. A valid user going rogue
2. Access within the network through the deployment of Trojan horses. Potentially through email viruses, fake program downloads, worms, application of software vulnerabilities, web bugs, etc.
3. Social engineering to gain computer access codes
4. Access gained to the local network through the application of access codes gained from another remote system. This is a typical problem with individuals maintaining identical access codes across systems and is a critical weakness of the ever popular username/password protocol.

Numerous techniques have been applied to the intrusion and attack detection tasks. Typical techniques include:

1. Rule-based systems [9]
2. Signature-based systems [11]
3. Anomaly-based systems [14]
4. Data mining [10]
5. Visualization [1, 2, 3, 12, 13, 19]

Each of these techniques has advantages and disadvantages, however, the most critical aspect of these techniques is that there is currently no techniques that effectively solves and single problem, much less all of the problems inherent to intrusion detection. The techniques in general have particular difficulty dealing with attacks which have not previously been seen. Visualization provides a unique advantage in that it allows the analyst or administrator examining the network to apply their own knowledge, intuition, and perceptual abilities, to the analysis of the data in a visual form, without needing to rely on perusal of raw textual data. Such perusal of raw textual data is incredibly time consuming and ineffective.

2. Goals

The goal of this work was to develop visualization techniques for the visualization of host and network data to aid in intrusion detection and monitoring. These techniques needed to build on previous intrusion visualization techniques, such as that by Erbacher [1, 2, 3], Yurcik [19], and Ma [12, 13]. The techniques must apply perceptual metaphors to ensure the users focus and attention are drawn to areas of interest and not meaningless artifacts. This required careful consideration of human perception and pre-attentive vision [5] during the initial evaluation and refinement of the techniques. For example, we considered the following characteristics during evaluation and refinement:

1. Line intersections draw the attention of the user and thus need to have meaning and importance. Ineffective line crossings must be removed or reduced. This differs from the work by Ma et al. [13] which generates large numbers of line crossings ineffectively.
2. Screen real-estate must be used effectively with minimal occlusion. Again, this differs from previous work my many researchers which can generate enormous amounts of occlusion or use screen real-estate ineffectively.
3. The techniques must be scalable to extremely large networks. We have in fact generated a set of visualization techniques designed to provide different levels of abstraction and subsequently provide different levels of representation of the network such that the entirety of the network can be represented at one view while enormous detail can be represented at another.
4. Glyph based techniques are incorporated throughout. This will ensure the large numbers of parameters intrinsic to the data set can be incorporated within the visual representation.
5. The visualizations are geared towards including historical information. This is critical for providing context as to the meaning of each event. Again, some visualization techniques are designed to incorporate more historical information than others. In general the amount of context represented is configurable.

In addition to considering the design of the visualization techniques we had to consider the application of the visualizations to specific attacks. How will the developed attacks apply to known and future attacks? Intrinsically, the techniques are designed to show the basics of the activity occurring on the network. At a higher level, the techniques are designed to provide context, essentially showing the behavior of the connection over time. It is this concept of behavior that goes to the heart of many current intrusion detection techniques. As for specifics, the techniques are design to specifically and effectively represent the following types of attacks:

1. Certain aspects of insider threats are represented. For example, a typical metaphor of an attack once they have gained access to a system within a network is to immediately attempt to gain access to other systems within the network. This type of connect through will immediately be shown by several of the techniques
2. Attempts to steal information will be represented. By highlighting the bandwidth and number of connections between two systems, an unusual volume of activity will be represented effectively.
3. Port scans will be represented. By showing history information in conjunction with port and IP addresses, such attacks can be readily identified. Even more valuable, however, is that the type of port scan can be identified. This allows differentiation of a dispersed, “low and slow”, scan as opposed to a brute force scan typified by script kiddies. Such differentiation is critical as we do not wish to waste time with script kiddies but rather focus on the more competent, sophisticated scans.
4. Large numbers of connections, or attempted connections, to multiple remote hosts in rapid succession will be clearly visible. This will be represented in context with successful connections, highlighting successful attacks, worm distributions, and provide an indication of threatening systems far more rapidly than traditional techniques such that defenses can be initiated before its too late.
5. Statistical activity is represented, indicative of user specifiable parameters such that deviations from the norm can be rapidly identified and investigated. These statistics can include:

    a. System load
    b. Fragmentation
    c. Bandwidth usage
    d. Number of connections
    e. Connection rate
    f. Connection types
    g. Packet size
    h. Number of packets
    i. Number of ports
    j. Number of alerts
    k. Alert frequency
    l. Alert severity
    m. Alert Source (Snort, trending, data mining, etc.)
    n. Flags (Syn, Ack, Fin, Reset, No Frag)
    o. Connection duration

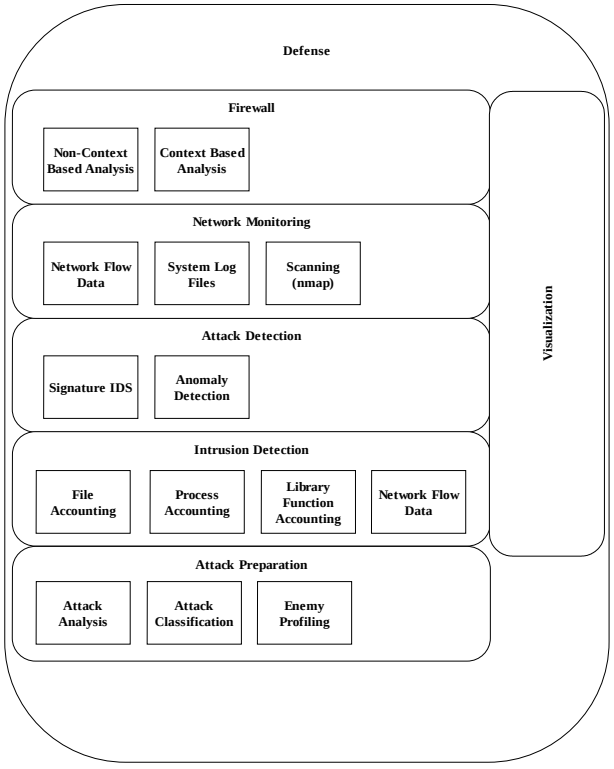
Many more attributes representative of attacks will be effectively represented, too many too enumerate or validate here. While we believe these goals have essentially been met we must perform additional evaluation and testing to confirm the effectiveness of the techniques. This will require complete implementation of the designed techniques.

3. Visualization Techniques

The developed visualization techniques are geared around a glyph-based metaphor [4] in which each visual object represents one data element. However, numerous visual attributes related to the visual object are dynamically mapped to different data parameters related to the data element. Thus, large numbers of parameters can be visually represented simultaneously. Often it is relationships between data parameters that identify artifacts or anomalies within the data needing to be examined and thus the glyph-based paradigm which excels at visually representing these relationships can be crucial to the analysis of multi-parametric data sets.

Ultimately, eight mockups were developed during the course of this project. Initial evaluation and feedback with respect to these mockups has been very positive. Response essentially indicated that the individuals who were shown these mockups would like to see them implemented. These responses were from expert network analysts as well as other researchers within the intrusion detection field. The mockups incorporated both 2D and 3D techniques, line-based and space filling techniques, and both host-based and network based representations.

The visualization techniques are adaptable to the needs of the environment and the monitoring needed. The diagram below shows typical lines of defense in a network.



Having multiple lines of defense is critical in critical networks. The visualization techniques are geared towards a majority of the defense perimeter levels. This is another advantage of the glyph-based paradigm. For example, the visualization can represent pure raw data collected unmodified from the network, snort alert data, the results of other intrusion detection tools such as data mining techniques, or any combination therein.

3.1 New Mockups

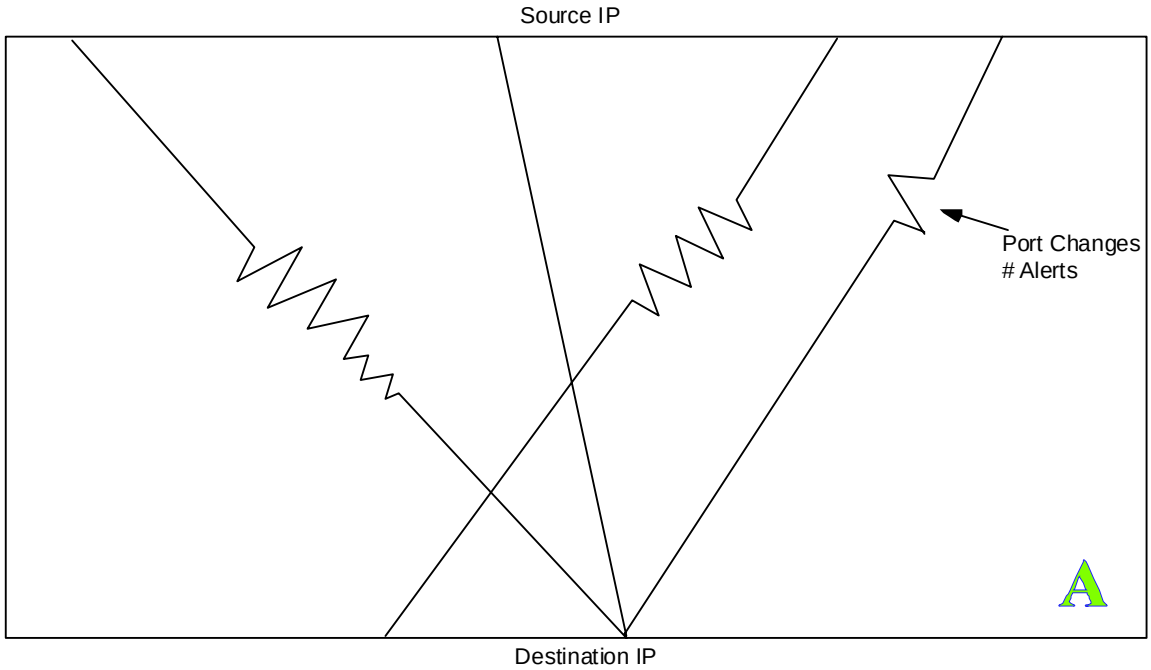
The following examples provide eight mockups developed during the course of Dr. Erbacher's term at AFRL. They each have advantages and disadvantages. When taken together they provide a set of novel and unique capabilities that provide functionality and power to the analyst not currently available.

3.1.1 Mockup "A": Line-Based Connections

The first mockup, labeled "A", is designed to show overall network activity as well as congestion points and volume of activity. The source IP is placed along the top edge of the display with the destination IP along the bottom edge of the display. This can be adjusted to place the local IP along the top edge and remote IP along the bottom edge, which may provide for more consistency. Large numbers of parameters can be integrated within this metaphor. The line may have an integrated histogram representing additional statistics, such as the port connected to, changes in port usage, number of alerts, etc. Each line may be considered a glyph with various visual attributes, including:

- Line thickness
- Line color
- Line style (solid, long dashes, short dashes, etc.)
- Histogram line angles
- Histogram line height
- Histogram line deviation

The goal with this technique is to incorporate sufficient information into a single display such that the analyst can identify anomalous activity without the need for additional sources of information.



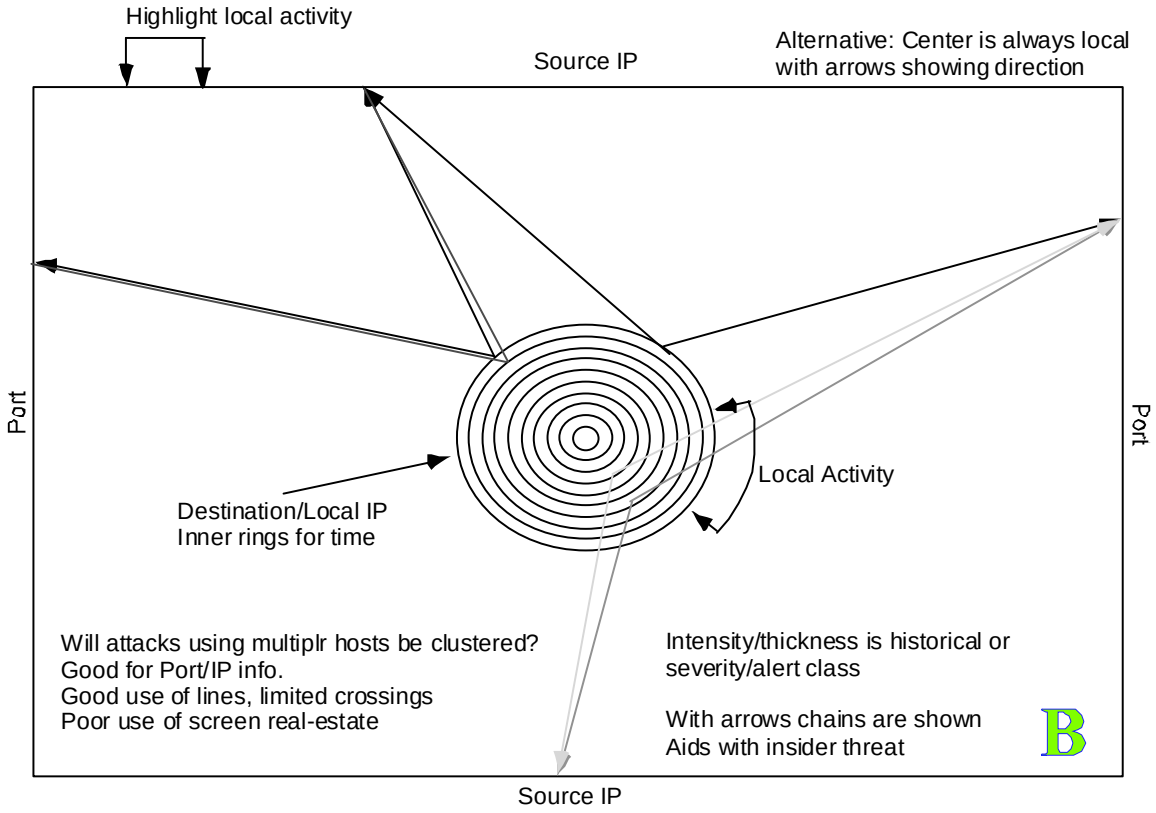
This technique is limited in that there are numerous line crossings which have no value to the analysis process. The technique overall contains many good ideas however. For example, large numbers of parameters are representable, as well as large numbers of hosts. The association of hosts with the edges of the screen aids differentiation of hosts.

3.1.2 Mockup "B": Improved Line-Based Connections

The second mockup, labeled "B", is based off of the first mockup to leverage its advantage and eliminate its disadvantages. The primary modification incorporated into the second mockup was using the center of the screen for IP address placement as well as the edges of the screen. The second modification was the placement of port numbers on both the left and right edges of the screen and IP addresses on both the top and bottom of the screen. The goal of these modifications was the removal of the unnecessary line intersections intrinsic to the previous mockup. While

the representation does differ, the nature of the glyphs remains the same. Consequently, the lines may have the same visual attributes as the previous technique. Additionally, unlike the previous technique, this representation allows for the incorporation of historical information. This can be critical for the identification of context and is another failing of the previous technique. Context is provided through the application of multiple rings within the center of the screen. Current connections will use the outer moist ring while older connections will use inner rings. The number of rings, duration represented by each rings, and the total history provided, will ultimately be configurable. The remaining key characteristics of this layout are as follows:

- The IP address represented at the top and bottom edges of the display may either be duplicated one for one, differentiated based on locality, or segregated numerically. A non-linear mapping will also likely prove most effective. These adjustments will need to be configurable.
- The IP address mappings may be set up for the source IP being at the top and bottom, with the destination IP within the center of the screen. Alternatively, the remote IP address may be mapped to the top and bottom of the screen with the local IP address within the center. The second alternative provides for more consistency and more screen real-estate for the remote IP addresses, where it is more greatly needed.
- The port number is duplicated one for one on both the left and right edges of the display. Again, a non-linear mapping may be appropriate. This ensures no line intersections will be required.



This technique appears to offer enormous promise and will benefit from a complete implementation. Its main disadvantages are its limited representation of history and limited number of available visual attributes for mapping data onto. It does, however, appear to promise enormous effectively, especially in conjunction with additional techniques.

3.1.3 Mockup "C": Radial Space-Filling Statistics

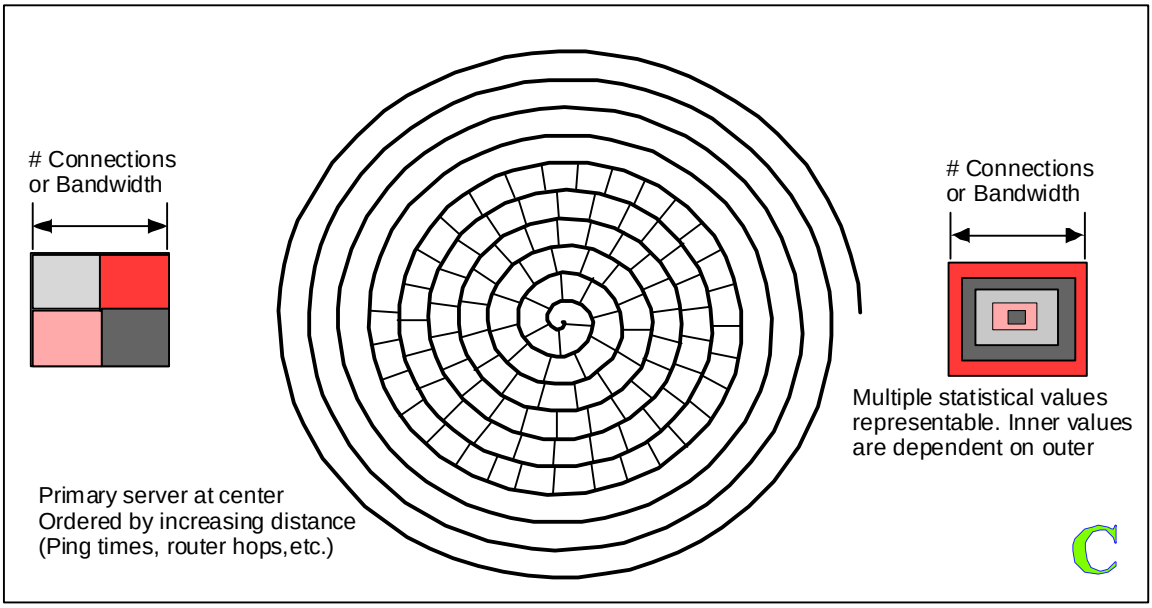
The next mockup was geared specifically for the representation of large numbers of statistics. This technique would work effectively with mockup "B", as it can represent the large numbers of statistics that the second technique cannot. It is lacking in that it can not provide the connection information that is the key to the second technique. Together they provide a complete set of capabilities. The technique shown in mockup "B" is considered a space-

filling technique; as opposed to a line-based technique. It is geared towards maximizing screen real-estate by using all available pixels, if at all possible.

This technique essentially provides for the monitoring of network hosts. This is another glyph-based technique in which the most critical system will be placed in the center of the display, where the analyst's focus of attention will most readily be. Less severe hosts are placed in a spiral going out from the center, with the least critical hosts towards the outer edge. Hosts of similar criticality can be placed in IP address order.

The glyph is designed around a rectangular metaphor. In essence any organization will be effective, two examples are provided here. The most critical aspect of the glyph is its width, which should be related to the most critical parameter of the data currently under analysis. This will draw attention to those glyphs of highest severity. Examples of critical attributes include: number of alerts, bandwidth usage, number of connections, system load, etc. Any number of parameters can be mapped to the glyph merely by incorporating additional visual attributes (e.g., additional rings). Each ring in and of itself may represent multiple parameters, through its intensity, color, thickness, pattern, etc.

History information may be incorporated through additional rings or other visual attributes within the data set. When incorporating history information it is critical that it be differentiated from the normal or current parameter values.



This technique is limited in that it does not provide connection information and positioning is somewhat arbitrary. The relationships between elements is not shown or represented in any way. Mockup "D" improves the presentation of relationships by using a similar glyph-based metaphor to represent statistical information but in a different layout; one that incorporates relational information.

3.1.4 Mockup "D": Treemap-Based Statistics

More specifically, mockup "D" the layout is based on treemaps [6]. Treemaps are a space filling visualization techniques originally designed for the representation of hard drives and its associated hierarchical information. While treemaps have best served representing hierarchical information they are effective at showing any types of representational information.

Treemaps work be associating corresponding amounts of screen real-estate with the corresponding value of an associated variable. For example, continuing with the hard-drive example we would have the following steps in the development of the treemap:

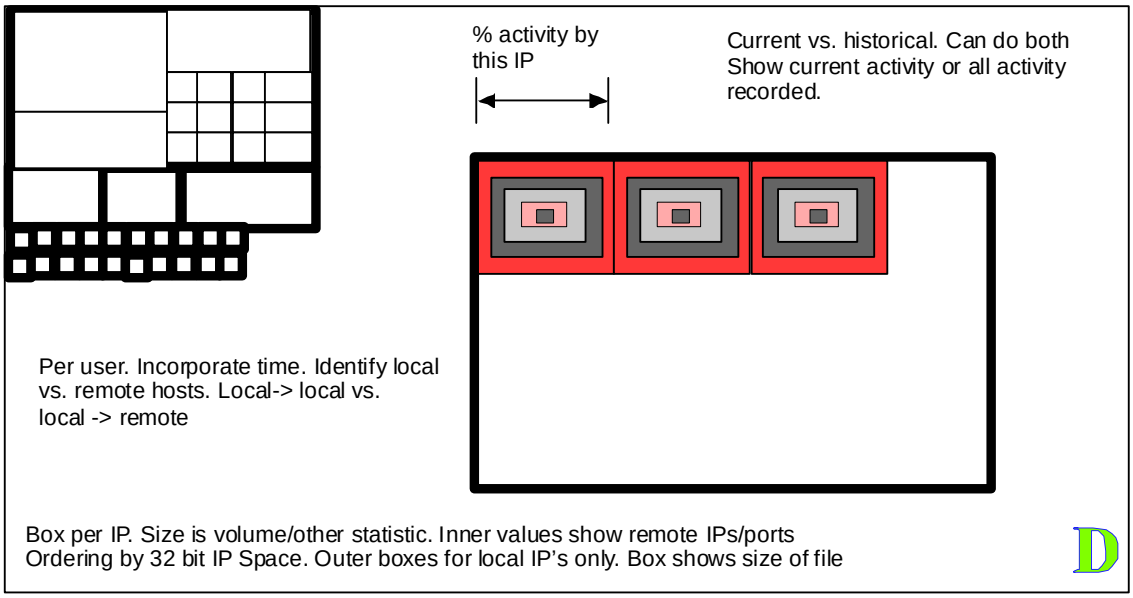
1. The screen real-estate is divided based on the amount of disk space used by each of the top level directories; for example performing a "du –sk" on each of these directories.

2. A thick border is placed around each of these top level regions.
3. The algorithm proceeds into each subdirectory, repeating the algorithm and further subdividing the screen real-estate using slightly less thick lines.
4. The process continues until one of the following conditions is reached: a single pixel, the bottom of the directory hierarchy, or a user specified maximum depth.

Applying this algorithm to a network will require adjustments to the representation. In fact, there can be multiple representations or organizations that could prove effective. For example, we envision organizations based on history and network activity as being potentially valuable.

In terms of history, the top level subdivision will be based on local IP address. Subsequent subdivisions will be based on a chosen statistics (bandwidth, number of connections, number of alerts, etc.) over a specific period of time. Thus larger regions will be given to the period of time with the greatest value for the chosen statistic. Further subdivisions of the time period will further create the hierarchy.

In terms of network activity, the top level will again be local IP addresses. Subsequent subdivisions will be based on connections made to/from this IP address. This hierarchy has the advantage of showing connection information, even if somewhat repetitive. Additionally, the two techniques could be integrated, particularly if # of connections is used as a primary statistic.



The main limitation of this technique is its complexity and the possible difficulty analysts will have in comprehending its activity. Its adaptability and ability to show multiple representations through different mappings is a great advantage as much of an analysts work could potentially be performed in this single display.

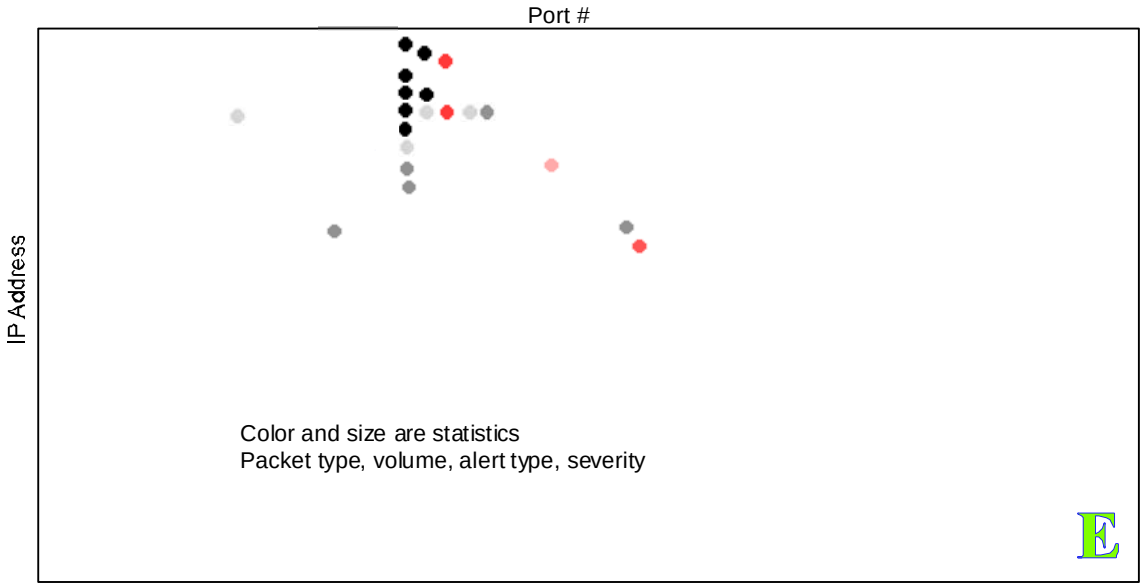
3.1.5 Mockup "E": Scatter Plot-Based Statistics and Connections

Mockup "E" presents a more traditional representation, namely that of a scatter plot. Clearly, there will be advantages to allowing control of the variables mapped to the X and Y axes. Example mappings that could prove valuable include:

- IP address versus port number
- IP address versus IP address
- IP address versus number of alerts
- IP address versus number of connections

Each of these mappings will show critical activity relevant to the identification and analysis of intrusion data. In conjunction with the axes mappings each visual object can be a glyph in its own right, though only simple examples

are presented in this mockup. While a simple technique, the technique may have great potential if to assist the other techniques if the data parameters are mapped effectively.



3.1.6 Mockup "F": Gravity-Based Statistics and Connections

The goal with mockup "F" is to incorporate a sense of automatic clustering, organization, and relative statistics. This is done through the application of force directed graphs [17]. More specifically, we will incorporate the concept of global gravity and local forces between connected nodes. In this way, nodes will be drawn to towards the center of the display based on the value of a nodes chosen statistic. Thus, if number of alerts is the chosen statistic then the nodes with the highest number of alerts will be closest to the center while those with the least number of alerts will be furthest away.

A second statistic will be used to control the distance connected nodes will be from one another. Nodes are connected when there is or has been active network communication between the two associated hosts. This force is applied similarly to the gravity described in the previous paragraph. It will need to be determined the exact formulation that will integrate the two competing equations in an effective way.

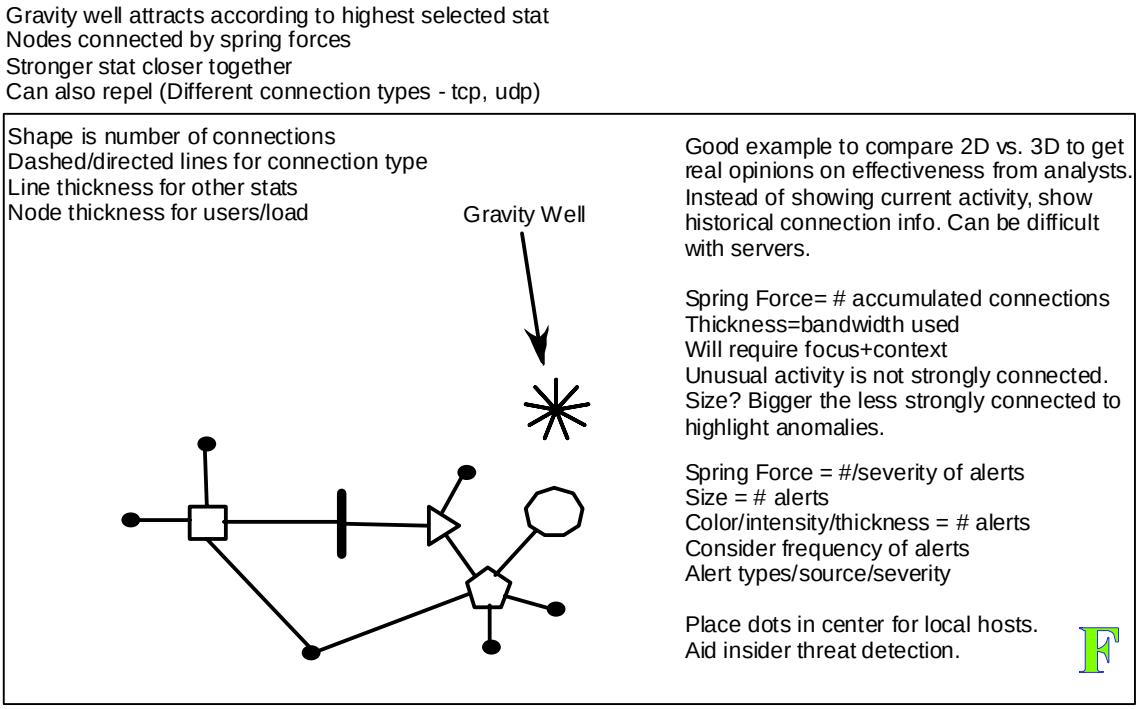
In addition to attractive forces there can be repulsion forces. This could be applied to represent diametrically opposite activity, e.g., different connection types (tcp versus udp). Again, the statistic applied to these forces must be user selectable.

Additional glyph-based characteristics can be applied to the line width, line intensity, line color, line style, node shape, node visual attributes. In the representation below the node shape is a basic representation of its number of connections. Thus, a single connection is represented as a dot. This will represent most remote hosts and do so in a space efficient manner that matches well with our need to use screen real-estate effectively, especially considering we will not have additional characteristics to map to any type of visual attribute that might be associated with remote hosts due to our inability to monitor such systems. Two connections is represented as a single line, three connections as a triangle, four connections as a square, etc.

Shapes can also be used to represent the type of node or be used as glyphs similar to what has been exemplified in earlier visualizations. This technique will also be interesting to examine in three dimensions. Most techniques will not benefit from such a 3D approach but the clustering and complexity of this technique could make a three dimensional representation effective.

This technique has enormous potential for representing anomalous and typical behavior. This is particularly true of context can be maintained over a long period of time. This will result in anomalous activity in not being strongly

connected. We can also represent rate of change as an indicator of anomalous activity. When replayed rapidly, a sudden long sequence of connections will result in large, rapid changes in node positions which should not occur.



The main disadvantage of this technique is its complexity of implementation and the enormous impact the selection of control variables can have on the visualization. Selection of appropriate control variables, especially for the gravity and spring forces, will result in extremely effective visualizations and animations for intrusion detection and analysis. Poor selections will result in ineffective representations.

3.1.7 Mockup "G": Inter-Network Connection Summarization

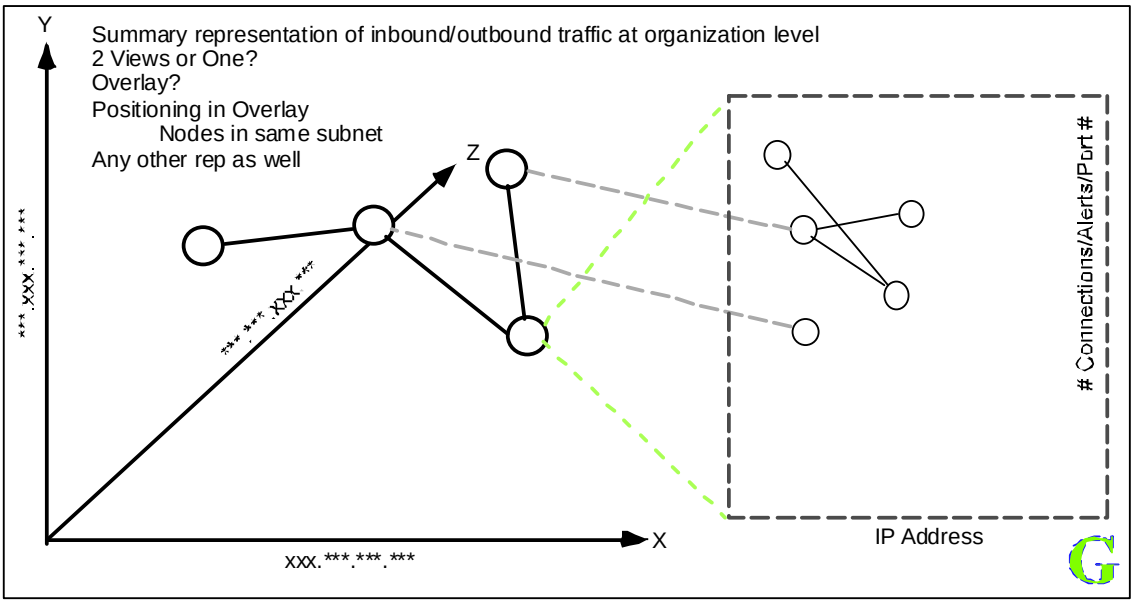
As mentioned in the previous mockup, 3D techniques tend to be less effective than 2D representations, due to the need to interact with the display to see all viewpoints and the potential for elements to be occluded. However, we do wish to consider 3D representations to provide a complete repertoire of techniques and to provide #D techniques which we can use for comparison and analysis in conjunction with the 2D techniques. This analysis will provide feedback as to the true effectiveness of 3D techniques in comparison to 2D techniques and aid determination as to whether work with 3D techniques is viable for continued exploration.

Mockup "G" provides a simple 3D representation in which IP addresses are mapped onto the 3 axes of a typical 3D space. The IP addresses are used in 111.222.333.444 notation. The first number '111' is mapped to the x axis. The second number '222' is mapped to the y axis. The third number '333' is mapped to the z axis. The fourth number is ignored at this level of analysis. Thus each node within the visualization represents a subnet. The node or glyph can provide visual attributes as with the previous glyph examples to represent the number of active connections and nodes contained within the cluster.

Connections between nodes within this visualization show connections into and out of the subnet. This allows for monitoring and analysis of the external activity of the subnet. Since most attacks originate from external sources, this type of monitoring will provide a summary of the pertinent information without the clutter inherent to traffic within the subnet which can distract the analyst unnecessarily. Additionally, it can be assumed that other administrators will be responsible for the traffic contained within each subnet.

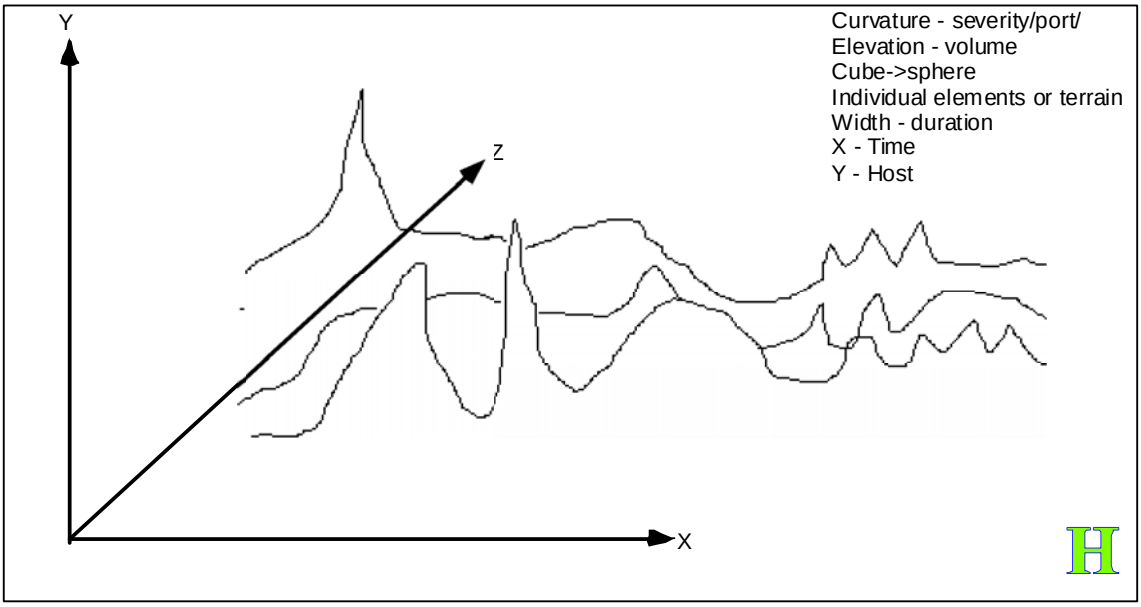
Of critical importance to this visualization technique is the ability to zoom into the each node and examine the subnet. The zoomed in representation can either use the same visualization or it can be a completely different visualization. The goal is to provide more detail as to the activity of the hosts contained within this subnet.

This technique is enormously advantageous due to its ability to summarize external activity. It does have limitations in that it is a 3D technique and will require interaction in order to completely analyze. It is also critically dependent on the ability to zoom into each node to examine the sub-networks. This abstracting out of activity within a subnet could cause critical activity to be missed.



3.1.8 Mockup "H": Surface-Based Statistics

The final mockup, "H", is shown below. The idea with this technique is to provide a 3D statistical summary. It is similar to the other techniques that attempt to provide statistical summaries in its goal with the added hope that the 3D representation will allow more detail to be represented and more effectively. This technique attempts to generate a statistical landscape in which unusual features of the terrain will indicate anomalies within the network activity. Thus, we map time to the x axis, host to the z axis, and a statistical variable to the y axis. This will most likely be number of alerts, number of connections, bandwidth in use, or some other critical variable. Additional variables can be mapped to the color, intensity, and transparency of each point. This representation allows for an enormous duration and number of hosts to be represented simultaneously.

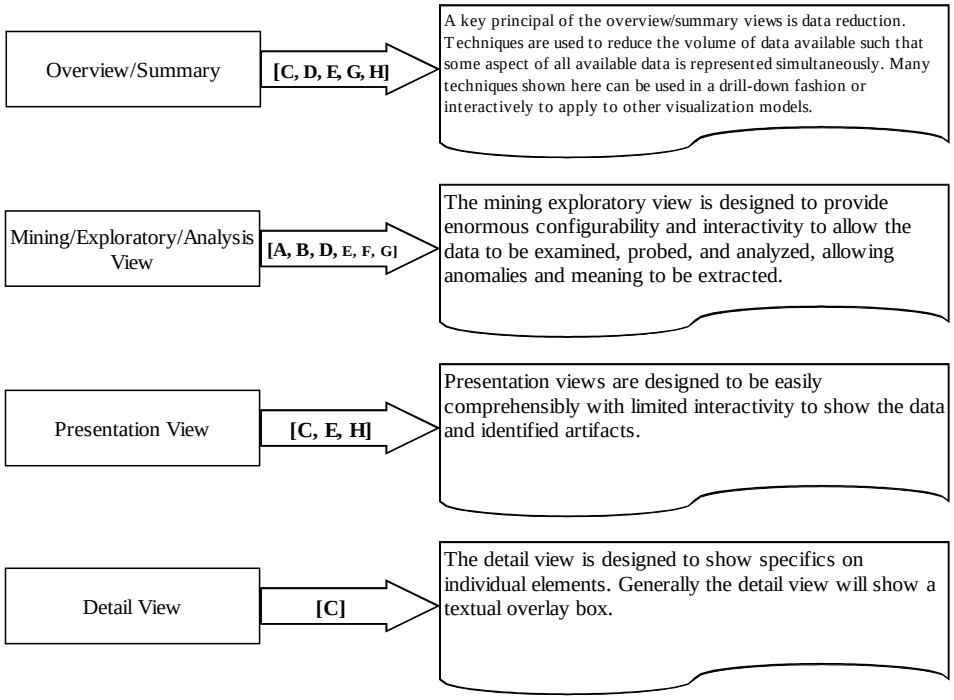


The disadvantage in this technique is that it does not show connection information and is limited in the number of data parameters that can be represented simultaneously. Much of the additional information can be provided through probing. Thus when an element or region is selected additional representational techniques are displayed showing the selected elements in new techniques that will provide far greater detail.

3.2 Visualization Hierarchy

The designed visualization techniques must be used together in a single hierarchy of visualizations, as exemplified in the below diagram. The visualizations get more detailed the farther down the list they are used, however, many of the visualizations may be used at multiple levels, providing varying degrees of detail. This hierarchy is critical due to the inability of a single visualization technique to show the analyst all aspects of a data set, as well as the inability for all data elements to be represented on the display simultaneously. For example, the analyst will examine the overview display to get a sense of the data in its entirety though with limited detail. The exploratory view is then examined with respect to a subset of the data to garner more comprehension of the data. Finally, the detail view is used to extract specifics as to IP addresses, etc. The presentation view is generally used for offline presentations, e.g., to management.

Thus, in a typical environment all of the representations will be needed to ensure all the capabilities are provided. In actuality, multiple representations of each level will be provided.



3.3 Interaction techniques

The effectiveness of the developed visualization techniques will be highly dependent on the incorporated interaction techniques. Without interaction the visual displays will be of limited value, particularly since the user will need some way to relate the visual representations to actual IP addresses, port numbers, etc., such that the administrator or analyst knows what systems to take action against. Typical metaphors that will need to be incorporated include:

- Probing. This provides feedback as to the actual raw data for a selected element. This can be the numerical IP address or port number or the raw packet data related to an identified event.
- Selection/Highlighting. By selecting and highlighting nodes, linkages, or events these events can be more easily monitored, examined, and followed during the course of an investigation.
- Coordinated views. No single visualization techniques can always represent exactly what an analyst needs to see. Thus multiple visualization techniques must be incorporated into the environment. These represent

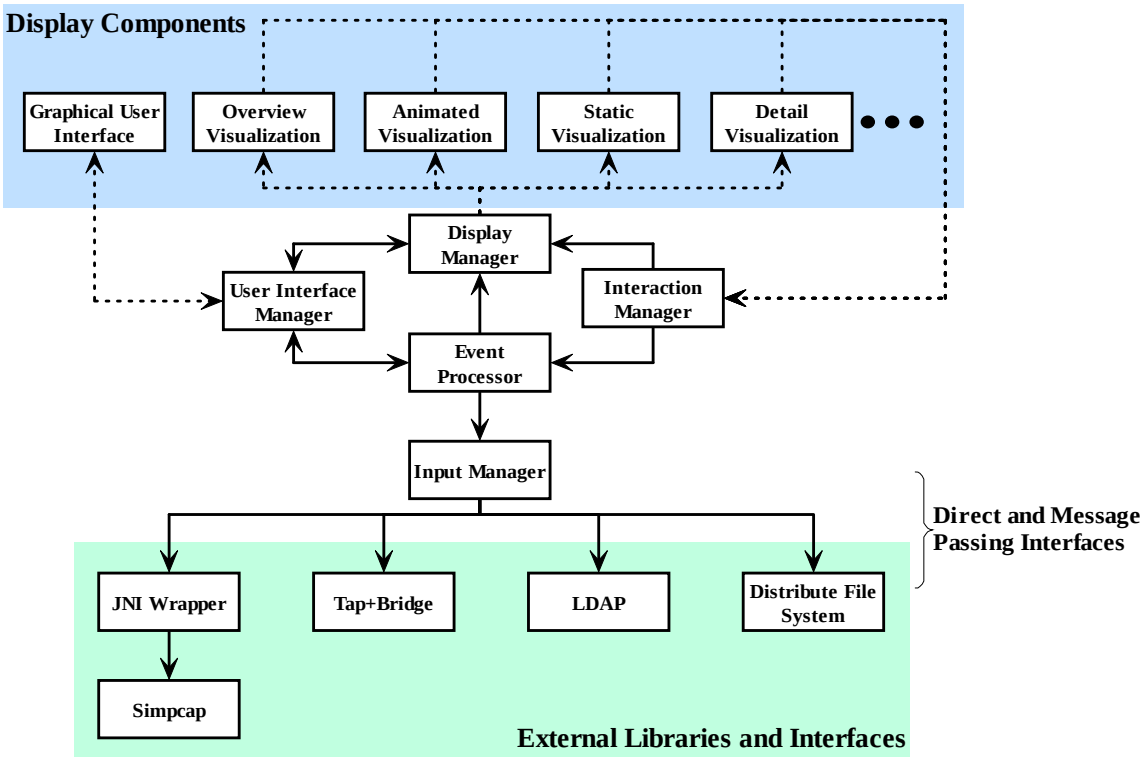
multiple views of the data. Through coordinated views, each interaction in one view is transferred to all other views. Thus a selection in display results in a selection in another display. This eases context and focus within multiple views, allowing an identified host to be identified and followed in multiple views.

Other types of interaction will need to be incorporated to improve the effectiveness and capability of the designed techniques. These include:

- Data parameter to visual attribute mappings. All of the visualization techniques identified during this project incorporate glyph-based techniques, with each glyph incorporating multiple visual attributes. We have identified numerous statistics that may be applied to these visual attributes. To acquire the most effectiveness from these visualizations we must provide an interface to control which data parameters or statistics are mapped to each visual attribute.
- Enable/Disable visual attributes. We must provide the ability to select attributes, glyphs, or regions of the display to either filter out or focus on. This will aid in reducing the clutter within the display and improve the ability of the analyst to focus in on elements of particular interest or concern.
- Zooming. The sheer number of visual elements to be displayed, i.e., the number of hosts to be represented, will require that an enormous number of elements be display simultaneously. This reduces the visual acuity of each element. Through zooming, a small number of elements can be examined in greater detail, thus increasing the visual acuity of those elements.

4. Visualization Architecture

An initial software architecture has been designed to be followed in the development of the software implementation. This architecture is shown below. The goal behind the architecture is to support multiple input data formats, multiple visualization techniques, and a complete user interface. While this architecture only provides a high level overview it does provide the foundation.



The visualization techniques will obviously incorporate the mockups described previously. When considering the input metaphors we must consider techniques currently in use by AFRL; thus the incorporation of Simpcap and Tap+Bridge. These are basic input tools developed and in use by AFRL. By providing a general, adaptable input mechanism additional metaphors can easily be incorporated. Possibilities include LDAP, distributed file systems,

postgres, snort, etc. To provide the most robustness a message passing interface, indicated by dotted lines, will be incorporated for communication of events. This will ensure that multiple display windows can easily be incorporated without any impact on the effectiveness of the interaction.

The environment will be based on a java implementation using jogl (java bindings for OpenGL) for the graphics. Of course Swing will be used for the majority of the user interface components. Java was chosen as it provides for easily collaboration and distribution. Java looks to be the direction for future implementations. Additionally, the Tap+Bridge tool is implemented in java and as such a java implementation of the graphics will ease integration of the two. OpenGL is used for the graphical components due to its popularity, ease of use, and efficiency.

5. Simpcap JNI Wrapper

As a first step in implementing the above architecture a Java Native Interface wrapper has been implemented around Simpcap to provide access to its functionality from within Java. This is a necessity since Simpcap is intrinsically a C library while we have chosen to provide a java implementation for this environment. The Java implementation intrinsically provides an identical interface as to the original C version, with modifications for the class-based interface and other fundamental differences in the languages. The functionality is accessed through the Simpcap class as follows: simpcap.*function*. An example java implementation that uses Simpcap is shown below. When comparing this example to the original C version the similarity is notable. This will allow for easy comprehension and adaptability by developers.

```
import java.awt.*;
import java.awt.event.*;
class example1 {
    public static void main(String[] args) {
        Simpcap simpcap=new Simpcap();
        byte[] data=new byte[500];

        byte[] error=new byte[500];
        byte[] dummy=null;

        Packet mypacket=new Packet();
        String filename=new String("res.txt");
        int s1;
        int rawfid;
        Pcap_File_Header hdr=new Pcap_File_Header();

        int ack=0;

        if(args.length < 1) {
                System.out.println"Could not open simpcap cature session\n");
                Return;
            }

        if((s1 = simpcap.Open_Session_Offline(args[0], dummy, error)) == -1) {
                System.out.println("Could not open simpcap cature session 1\n");
                return;
            }

        rawfid=simpcap.Open_File_Ptr(filename);

        while(simpcap.GetPacket(s1, mypacket) > 0) {
                if((mypacket.Net.emb_p_type == simpcap.TCP) &&
                   ((mypacket.Transport.tcp_hdr.flags & 0x10)>0) )
                    ack++;

                simpcap.Print_Packet(rawfid, mypacket);
                mypacket=new Packet();
            }

        simpcap.Close_Session(s1);
        System.out.println("Total ACK packets = " + ACK);
        return;
    }
}
```

Clearly, the java implementation will incur a performance penalty, especially when considering the multiple layers of implementation that must be executed. Essentially the java environment must pass calls through the JNI interface to the simpcap C code, which then calls native java code, again through the JNI, to set the result values. This results in the following call paradigm: Java -> JNI -> C -> JNI ->Java. However, this does not mean that the implementation is too slow and with the power of today's computers, there ever growing capability, and the improvement in performance of java, the performance impact of this implementation should not be of concern.

6. Trending

The goal with trending is to enhance intrusion detection analysis by providing trending data indicating the severity and comprehensiveness of an attack. Trending will provide an indication as to the extent to which an attack will escalate or is escalating. In this aspect of Dr. Erbacher's work he performed an initial investigation into techniques and algorithms which may be applied to this form of trending. The four principal techniques identified include:

- Linear regression [8]. Linear regression is a fairly simplistic statistical analysis tool. It can be applied to the intrusion detection scenario but only with limited success. The main difficulty is the need to identify the window, data period, onto which the linear regression should be applied. The linear regression is also not designed for the level of noise encountered with intrusion data.
- Fuzzy logic [18]. The idea with fuzzy logic is to associate true and false values with a range of data. This would allow determination as to whether a series of events have exceeded a pre-specified threat level and notify the analyst. The effectiveness of fuzzy logic as it applies to intrusion data has not yet been proven.
- Wavelets [16]. Wavelet theory is a statistical technique that has found wider application to a variety of tasks. Like linear regression it is designed to identify a best fit. However, wavelet theory originally derived from radio signals in which there can be an enormous amount of noise. Thus wavelets can handle the noise intrinsic to intrusion data. Wavelets have the added advantage in that the analyst does not need to manually specify the window onto which the algorithm should be applied, as is required with linear regression.
- Visualization. Visualization can be applied directly to trending. The most notable such technique is exhibited through the environment called ThemeRiver [7]. ThemeRiver implements a concept put forth by Tufte and uses a typical XY graph mapping. Time is mapped onto the X axis while the Y axis is used to represent corresponding amounts of the variables under investigation. More specifically, the variable's volume along the Y axis represents its value and each variable is mapped on top of each other.

7. Other Tasks

While the majority of Dr. Erbacher's activity focused on the above intrusion detection visualization work, additional activity was performed in several other areas. More specifically, activity was performed with respect to computer forensics and cyber command and control. These activities were at a much lower load than the above but accomplishments were still made.

7.1 Computer Forensics

The goal with computer forensics is to analyze computer hard drives found at crime scenes, or after an intrusion to identify the full impact on the hard drive. With such hard drives the criminal could have hidden incriminating evidence anywhere on the hard drive. For example, authorities may need to locate a single file containing true accounting record, rather than fake records, evidence of conspiracy, bribes, threats (e.g., Microsoft Word files, Microsoft Excel files, etc.). With the size of today's hard drives, in the hundreds of gigabytes, identifying a single file that may be hidden anywhere can be extremely time consuming. Current techniques rely on textual-based techniques and console commands, principally searching metaphors.

Dr. Erbacher's goal was to propose novel visualization techniques to aid in the visual analysis of hard drives. The technique that was proposed and received much positive response from the DEVA group was to apply the treemap concept, as described in relation to mockup "B". As the original treemap is geared primarily towards the representation of hierarchical information we will need to modify the concept to be more applicable for the representation of individual files as is required by the forensic analysis process. The hierarchical information still remains critical as it will identify anomalous entries, e.g., a file located in directory with no similar files. Thus, we are exploring the concept of *filtered treemaps* to improve the visual presence of individual files. This will also incorporate the use of additional statistics as opposed to file size to identify files of particular relevance. Finally,

interaction techniques to allow analysts to use the commands they have become familiar with, such as searching and comparing, will be integrated with the visual interface. Results in this scenario would rely on highlighting identified elements. This is far more effective than the typical textual results as the analyst can gain a better determination as to whether the file is anomalous and requires additional analysis and do so far more rapidly.

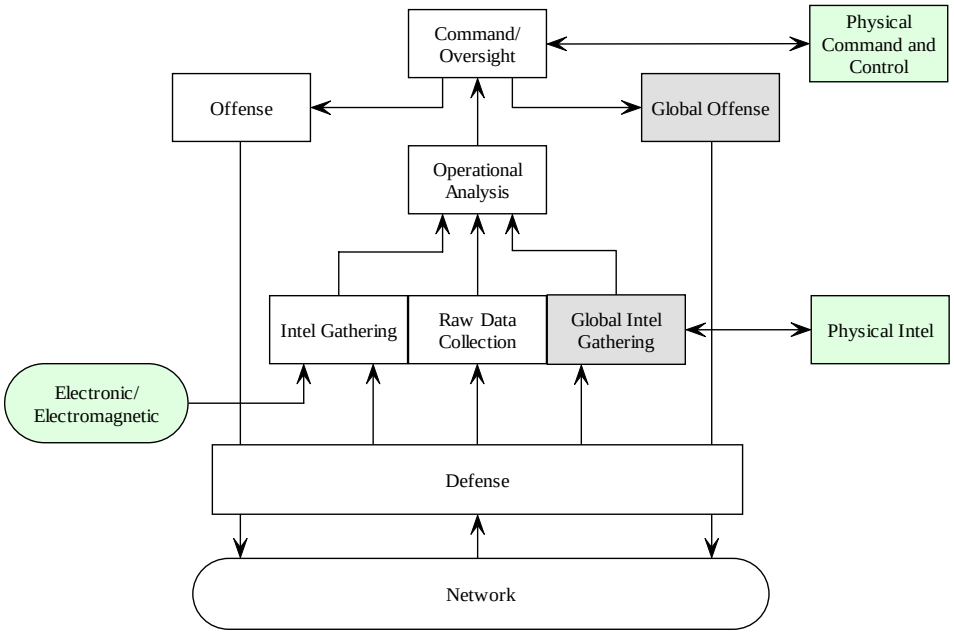
A second task within this project was to assist in advising an undergraduate student as he performed an initial series of research to identify the requirements legal and process requirements for computer forensics. This research was performed as part of his internship for the AFRL. The result of this research was a paper documenting an analysis of the literature. These results will be critical for the future development of forensic tools as it identifies the needs of such future tools in order for their results to be applicable to criminal prosecutions and subsequently fulfill needed legal requirements.

A critical result of this is the need to incorporate process monitoring and analysis capabilities into any developed visualization environment. This will allow all activities performed by an analyst to be recorded and replayed, ultimately within the visualization environment. Thus, the methodology applied by the analyst to be reviewed for appropriateness.

7.2 Cyber Command and Control

Dr. Erbacher provided input into the development of a cyber command and control infrastructure. Dr. Erbacher took the point of view of the traditional network, its requirements in a cyber command and control environment, and how it needs to integrate with the traditional physical military. This portion of the paradigm does not examine how cyber infrastructures can assist in physical command and control, i.e., the collection, analysis, and presentation of physical Intel.

The concept is exemplified in the below diagram. In this scenario, we are considering principally cyber attacks and defenses. However, a cyber reaction may require physical Intel or a physical response; thus the association with the physical command and control infrastructure. More specifically, all network activity must operate through the network-based defenses to reduce the possibility of exploitation and attack from opposition forces. Even disengaging network defenses temporarily for a single computer can lead to that computer being compromised and providing an access point on th inside of the perimeter defenses when the defenses are re-enabled.



Intel gathering is critically important for all higher level actions. Such Intel will incorporate raw network data, global cyber-based Intel, and physical electronic and electromagnetic data. Global Intel will consist of data collected from web sites, email and other communications and data storage. The physical data consists of positions of friendly

and hostile unit positions and network-based activity. For example, with the wired nature of today's military it is critical that the cyber command and control be aware of the location of all units, both friendly and hostile. Hostile units may consist of jammers, enemy units, etc. In the case of jammers deployed against friendly units it will be necessary to implement multiple response mechanisms. First, the location of such activity needs to be identified, likely through the monitoring of error rates. Second, a physical response may be needed to remove the offending activity. Thus analysis of available Intel can result in one of three responses: localized cyber response, physical response, or global cyber response. An example of a global cyber response is the removal of web sites instantiated throughout the world for propaganda, fund acquisition, and force direction.

8. Conclusions

This project made significant progress in terms of examining the needs of intrusion analysts, identifying visualization paradigms effective for such analysis, developing an architecture for the implementation of such and environment, and providing the groundwork for the implementation. In particular, the data access paradigm was prepared for development within a java framework, through the implementation of a JNI wrapper.

The project also provided opportunity for interaction with numerous groups within AFRL. This allowed discussions with respect to other aspects of information assurance, including cyber command and control, computer forensics, and intrusion detection. Many of the discussed techniques have applicability to these techniques. For cyber command and control the visualization techniques and aid as a fundamental operational framework of the command infrastructure. With respect to computer forensics we discussed the applicability of the treemap visualization towards aiding in the analysis of computer hard drives.

9. Future Work

The principal task to be done is provide an implementation of each of the described mockups. This must be followed by user studies and associated data analysis for verification of user effectiveness and intrusion analysis effectiveness. We must also examine the effectiveness of the environment when dealing with extremely large-scale environments and different types of environments, e.g., academic versus military versus commercial. We must also examine the applicability of the techniques to other domains, including wireless, next generation internet, and network forensics. Finally, we must examine the generalizability of the techniques to broader domains, incorporating abstractions and generalizations of the techniques.

A subsequent analysis must be applied to the interaction techniques. The interaction techniques currently have taken second fiddle to the visualization techniques. Our proposed visualization techniques will incorporate basic interaction techniques to perform the basic capabilities described. However, these interactions must be enhanced and refined. This particularly related to interfaces that must be developed, such as those to specify the data parameter to visual attribute mappings.

The user interface for mapping data parameters to visual attributes is in and of itself a major topic within the field. Effective interaction paradigms have yet to be developed. Additionally, we must examine interfaces for more effectively interfacing with the underlying database, providing the power and capabilities provided by such databases to the user.

10. References

1. Robert F. Erbacher and Menashe Garber, "Fusion and Summarization of Behavior for Intrusion Detection Visualization," *Proceedings of the IASTED International Conference On Visualization, Imaging, and Image Processing*, Marbella, Spain, September 6 - 8, 2004, (*To Appear*).

2. Robert F. Erbacher, "Intrusion Behavior Detection Through Visualization," *Proceedings of the IEEE Systems, Man & Cybernetics Conference*, Crystal City, Virginia, October, 2003, pp. 2507-2513.

3. Robert F. Erbacher, Kenneth L. Walker, and Deborah A. Frincke, "Intrusion and Misuse Detection in Large-Scale Systems," *Computer Graphics and Applications*, Vol. 22, No. 1, January/February 2002, pp. 38-48.

4. Robert F. Erbacher and Georges G. Grinstein, "Issues in the Development of 3D Icons," *Visualization in Scientific Computing*, Springer-Verlag, 1995, pp. 109-123.

5. Markus Gross, Visual Computing, The Integration of Computer Graphics, Visual Perception and Imaging, Springer-Verlag, 1994.

6. Brian Johnson and Ben Shneiderman, "Treemaps: a space-filling approach to the visualization of hierarchical information structures," *Proceedings of IEEE Visualization '91*, pp. 284-291.

7. S. Havre, E. Hetzler, P. Whitney, and L. Nowell, ThemeRiver: Visualizing Thematic Changes in Large Document Collections. *IEEE TVCG*, Vol. 8, No. 1, Jan- Mar 2002.

8. Kivikunnas S., "Overview of process trend analysis method and applications." *Proceedings of the Workshop on Applications in Chemical and Biochemical Industry*, Aachen, Germany, 1999.

9. Koral Ilgun, Richard A. Kemmerer, and Phillip A. Porras, "State Transition Analysis: A Rule-Based Intrusion Detection Approach," *Journal of Software Engineering*, Vol. 21, No. 3, 1995, pp. 181-199.

10. Lee W., Stolfo S, and Mok K., "Adaptive Intrusion Detection: a Data Mining Approach," *Artificial Intelligence Review*, Vol. 14, No. 6, December 2000, pp. 533-567.

11. M. Roesch, "Snort- lightweight intrusion detection for networks," In *Proceedings of the 1999 USENIX LISA conference*, November 1999.

12. S. Teoh, Kwan-Liu Ma, and F. Wu, "Visual-based Anomaly Detection for BGP Origin AS Change Events," in Proceedings *of the 14th IFIP/IEEE Workshop on Distributed Systems: Operations and Management*, October 20-22.

13. Soon Tee Teoh, Kwan-Liu Ma, Shyhtsun Felix Wu, and Xiaoliang Zhao, "Case Study: Interactive Visualization for Internet Security," *Proceedings of IEEE Visualization*, 2002.

14. H. Teng, K. Chen, and S. C.-Y. Lu, "Adaptive real-time anomaly detection using inductively generated sequential patterns," In *IEEE Symposium on Security and Privacy*, 1999, pp. 278–284.

15. Vandenwauver, M. Claessens, J. Moreau, W., Vaduva, C., and Maier, R., "Why Enterprises Need More than Firewalls and Intrusion Detection Systems," *Proceedings of the Eigth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises* (WETICE'99), 1999, pp. 152-157.

16. H. Vedam and V. Venkatasubramanian, "A Wavelet Theory Based Adaptive Trend Analysis System for Process Monitoring and Fault Diagnosis", in the *Proceedings of the American Control Conference*, Albuquerque, NM, June 1997.

17. C. Walshaw, "A Multilevel Algorithm for Force-Directed Graph Drawing", Proceedings of the 8th International Symposium on Graph Drawing, 1984, pp. 171-182.

18. Wang X. Wei TYC. Reifman J. Tsoukalas LH., "Signal trend identification with fuzzy methods," *Proceedings 11th International Conference on Tools with Artificial Intelligence*. IEEE Computer Society Press, 1999, pp.332-5.

19. Xiaoxin Yin, William Yurcik, Yifan Li, Kiran Lakkaraju, Cristina Abad, " VisFlowConnect: Providing Security Situational Awareness by Visualizing Network Traffic Flows", *Workshop on Information Assurance (WIA04) held in conjunction with the 23rd IEEE International Performance Computing and Communications Conference(IPCCC 2004)* , 2004.