Information Visualization for an Intrusion Detection System

James Blustein
Daniel L. Silver
Ching-Lung Fu

Technical Report CS-2005-15

July 28, 2005

Faculty of Computer Science
6050 University Ave., Halifax, Nova Scotia, B3H 1W5, Canada

This technical report is an author's pre-print of an article to be presented at

Third Annual Conference on Privacy, Security and Trust
12–14 October 2005
at The Fairmont Algonquin
St. Andrews, New Brunswick, Canada

Ching-Lung Fu is the corresponding author.

Information Visualization for an Intrusion Detection System

James Blustein
Faculty of Computer Science
Dalhousie University
Halifax, NS, Canada
Email: jamie@cs.dal.ca

Daniel Silver
Jodrey School of Computer Science
Acadia University
Wolfville, NS, Canada
Email: danny.silver@acadiau.ca

Ching-Lung Fu
Faculty of Computer Science
Dalhousie University
Halifax, NS, Canada
Email: cfu@cs.dal.ca

***Abstract*— Spatial hypertext was developed from studies of how humans deal with information overflow particularly in situations where data needed to be interpreted quickly. Intrusion detection requires security managers of large networks to rapidly respond (often in real-time) to masses of information. Users of such systems need to recognize large developing patterns in masses of data, they prefer to work individually (although they must function in collaborative groups), and they rely on their intuitions more than deductive logic. Such users have particular personality characteristics and job needs which can be well served by interfaces which use a spatial hypertext model. Also, like most users, they prefer to be in charge of the process that they use the computer as a tool to assist with. The architecture proposed in this article is based on spatial hypertext and machine learning. That interface design allows for a great deal of interface flexibility and user control. The article discusses in detail how spatial hypertext, and the proposed architecture in particular, can well fulfill the needs of intrusion detection system users through personalized information filtering.**

I. Introduction

The perceived need for information security is demonstrated by laws in Europe and the USA [1], [2]. However many companies and other institutions do not follow good security practices [3]. Part of the reason for this distressing situation is the poor quality of tools that are available to security administrators [4]. Whitten and Tygar [5] showed that even for highly educated users, security systems with inappropriate user interfaces could undermine an enterprise's entire security apparatus. Although we are focusing on improving security through better interfaces, the work we report here is part of a larger project to improve security through the development of both user interfaces (UIs) *and* underlying functional technology.

The objective of this work is to develop user interfaces that help bridge the gap between monitoring software and users by developing interfaces that adapt to their users rather than systems that require the user to adapt their working styles. This article proposes an adaptive UI for an intrusion detection system (IDS) based on an uncommon interface model, namely spatial hypertext, that we feel is well suited to the specialized tasks of intrusion detection. Although response to a security alert is a necessary part of the context in which any security monitoring system operates, we do not consider it in detail in this article.

The article is organized as follows: first we present detailed background of the diverse areas that we are drawing upon (usability engineering, human-computer interaction, spatial hypertext, user adapted interaction and user modeling), before we discuss a framework for IDS using a combination of user modeling and spatial hypertext. We conclude with a summary that outlines the unique suitability of our approach to a serious and pressing problem.

II. Background

The motivation for our research is to improve security by developing better interfaces for system administrators to work with IDS. Intrusion detection systems are tools that monitor network traffic (at multiple levels, from low-level packets to higher level application-layer messages) to (a) reject traffic that is clearly dangerous, (b) alert system administrators to other potential attacks, and (c) to help system administrators to determine the dynamic state of their network and recognize patterns in traffic and performance [6]. These are difficult tasks in real life. The huge number of events in network traffics makes a high accuracy intrusion detection mechanism less effective. The result is the high number of detection reports. Unfortunately, there will be many false detections in these reports. IDS users filter out the false detections and deal with the real intrusions. It is time consuming to go through the detection reports, yet IDS is more effective if the response time is minimized. The effectiveness of IDS is difficult to be improved by detection technologies alone. Developing a user interface (UI) for IDS that helps the user to recognize the emerging intrusion patterns in network traffics should help improving the system effectiveness.

When developing UIs it is essential to characterize and understand the users, the tasks the software should help them to accomplish, and the context in which the software will be used [7]. With such complex and demanding cognitive tasks we also must consider the process, outcome, and users' satisfaction as major components of any measure of success [8].

A. Tasks and Users

1) Intrusion Detection Systems: Intrusion detection systems are usually used as security management tools for computer network administrators who monitor individual systems and

networks to detect inappropriate access [3]. The detection is best done in real-time to keep networks working at maximum efficiency. However some administrators choose to review traffic logs to detect potential attacks post facto [4]. Of course there are times when networks are under attack and the source of the attack must be determined before it can be stopped.

Administrators sometimes examine individual network data units (e.g., packets and messages) for signs of intrusion but more often they look for unusual patterns in the traffic, and rely on automated tools to prevent simple uncoordinated attacks. Therefore an important subtask for security administrators is to be able to characterize the usual state of the network and to identify unusual situations, and recognize potential problems when they are still developing.

2) Characteristics of an IDS User: Survey results published by Gates and Whalen [9] support the impression we formed from in-depth interviews with several computer security experts in industrial, military, and academic computing. Gates and Whalen found that ten times as many computer network security specialists have personality types corresponding to the Myers-Briggs Type Indicator (INJT) than in the general population. They further reported that the preferences stated by their survey population ($N = 76$) was in contrast to those of other types of security experts and computer specialists.

Although one must be careful not to infer meaning without sufficient basis, the tentative conclusions they reached match our own, specifically that "INJTs are perfectionists who value personal competence and their own original ideas; they also tend to not invite others to assist with their projects, and may not see practical weaknesses in their plans.... This result implies that security professionals are more focused on 'the big picture,' and that we have few practitioners who are focused on 'the here and now.' [9]"

Indeed one of the common characteristics found in intrusion detection is that security specialists look for attacks patterns that are not represented by single network transmissions (messages at the top-most layer, or packets at lower levels) [10], [4].

Security managers generally prefer to work alone even when they are members of a collaborative team in which any member can substitute for another [9], [3]. Due to the nature of the work, security managers often work in semi-secrecy within their organizations and can be alerted to network attacks at any time of the day or night. The need for their work is most often acknowledged when they fail to completely protect their systems from an attack.

B. Spatial Hypertext

The concepts of spatial hypertext (SH) evolved from hypertext in the process of looking for alternative representations for navigational and semantic links [11], [12]. Over the last decade, SH has emerged as a vigorous area of research emphasizing how people collect, organize, annotate, and interpret information. Unlike the most common type of hypertext as often experienced on the World Wide Web (WWW), the nodes in SH are not connected by explicit links. In SH, nodes are more abstract entities [11]. Relationships that might otherwise be unexpressed or explicitly represented by links in hypertext are instead represented by visual cues such as shape, colour, proximity, and alignment in SH. Some relationships that are too subtle to be clearly and succinctly represented by links may also be depicted visually. In the spatial-object model employed in SH, nodes are 'objects' which act as information placeholders, and spatial cues imply the relationships among the objects. Figure 1 is an example of a SH workspace.

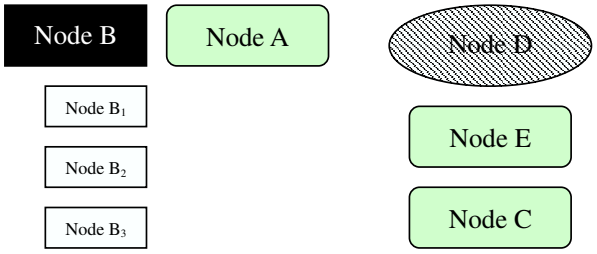

Fig. 1. Graphical representation of a spatial hypertext workspace

In the case of the Visual Knowledge Builder (VKB), the objects are visual elements which can contain text, (still and animated) images, and other media[1]. The relationships amongst the objects can be represented by spatial cues such as: colour, shape, proximity, alignment, containment, and overlap. For example, relationships between nodes can be inferred by similarity (in shape, colour, border styling and colour), and physical arrangement (proximity, overlap, and larger-scale patterns such as shared vertical alignment). Containment is a relationship indicator that has no direct analogy with notes-stuck-on-a-wall but it does correspond to nested coloured boxes arranged on the tabletop which can contain groups of cards, and other boxes.

SH allows users to manipulate the appearances of the objects and leave the relation representations implicit and ambiguous. This important property of the SH system gives the user the freedom to express and develop new insights from the materials [13], [14]. There is no limitation on where the users are allowed to place the objects in the workspace. The users express the the relationships among objects by changing visual (colour, shape, and size) and spatial (proximity and alignment) cues. In short, one can say that SH is a spatial-object model where the object holds the content of a document and the relationships among objects are expressed by changing the appearance and location of the objects [13].

SH is an excellent medium for information intensive work and knowledge structuring tasks [15]. Many management tasks such as network management and IDS tasks are information rich and complex. We believe SH will be suitable for IDS task management because the users will be able to directly

[1]It may be most helpful to readers unfamiliar with spatial hypertext to imagine VKB objects as familiar physical entities: note cards of different colours and shapes that are arranged on a tabletop, or Post-It™ notes of various colours and shapes on a wall.

manipulate all the objects in IDS and to have freedom to arrange the tasks and objects to suit their needs.

1) How Is It Used? — Flexibility in Process: Spatial hypertext allows flexible data presentation. SH supports direct manipulation of the objects and relationships presented in a workspace. The data represented objects can be altered, and the presentation (colour, position, etc.) of objects themselves can be altered by the user. By interacting with the objects in the workspace, users actively create meaning for themselves. By enabling users to manipulate the appearances of the objects and leave relational representations implicit and ambiguous, SH systems allows users to freely express and develop new insights from the materials represented by objects [13], [14].

SH systems provide their users with what could be an overwhelming number of options for arranging the display of objects to represent meaningful relations. How then do SH users avoid being overwhelmed when they have so many more choices to make than authors of relatively straightforward web pages written in XHTML? There are three main answers to that question:

- SH is mainly used much as physical notes being arranged on a surface during a brainstorming session, that is, the purpose is not to present some preconceived structure of information to a second party but rather for the author (or authors) to identify or develop the information. The ambiguity in the information structure allows new insights and relationships emerge as the users work through the materials [11]. Humans' exceptional spatial intelligence helps users to recognize relationships instantly from the ambiguity in the information [13].
- Some SH systems (notably VKB) include tools (known as spatial parsers) that identify apparently related structures and suggest automatic changes to users ("It looks as though you are constructing a vertical list. Would you like me to align those objects for you?"), and identify potential meaning in arrangements of objects [11].
- If SH gains wide acceptance within an organization, it seems extremely likely that standard schemata and genres will appear much as they have for the WWW.

2) Beyond The Author-Reader Dichotomy: On the WWW there has been for some time a clear distinction between the author of a website and the reader (or user) of that site [2] SH does not employ that dichotomy — the data represented objects can be altered, and the presentation (colour, position, etc.) of objects themselves can be altered by the reader. By allowing users to manipulate the appearances of the objects and leave relational representations implicit and ambiguous, SH system allows users freedom to express and develops new insights from the materials represented by objects [13], [14].

Although SH may have many applications, one of its first uses was for 'information triage' — successfully managing an overflow of information. SH systems equipped with spatial parsers are designed to help relieve the burden of classification and organization from the user without removing their control over the process of information management. More specifically, machine learning components of SH systems can use inferred relationships between objects (based on characteristics and relationships between groups of objects) to present information in an appropriate format that identifies important potential relationships to the user. One could say that SH characterizes the position of an object in a potentially high-dimensional cognitive space.

[2]Astute readers will realize that this distinction has been dramatically changed by the prominence of Wikis but, for the sake of clarity, we ask readers to disregard Wikis for the time being.

C. User Adapted Interaction and Use Modeling

User Adapted Interaction (UAI) is the study of how computer systems can be tailored in terms of function and interface to individual users. A user interface is said to be adaptable if its content or format can be manually tailored by the user [16]. Adaptive User Interfaces, or Adaptive UI, focuses on the automated tailoring of an adaptable UI based on a user model.

A user model describes a user's behavioral characteristics [17], [18]. User Modeling (UM), is the process of acquiring such information either through overt methods such as a questionnaire or covert methods such as recording frequently used commands. The simplest and oldest form of UM is recording explicit user preferences. This method is commonly available and provides accurate information; however users tend to avoid customizing software [19], [20] because of the time it requires to manually set and update the parameters. More modern techniques incorporate machine learning subsystems that infer implicit user characteristics by constructing a model from covert data [21].

Adaptive UI is a promising user-centered approach designed to tailor a system's interface behaviour to the idiosyncrasies of the user and the changing environment of the application. Research into Adaptive UI brings together concepts from Human-Computer Interaction (HCI) and UM to improve the usability and performance of software systems. Controllability is one of the major usability issues for Adaptive UI technology. Some researchers advocate maximum user control over all aspects of system adaptation, others suggest that maximum control is not always be the best approach as it can lead to distraction and inefficiency [22]. There has been much discussion among researchers about controllability trade-offs. However, as Jameson [23] argues, there is a deficiency of systematically gathered evidence about what users themselves think about adaptation and controllability.

Peng and Silver [24] propose a theory of user interaction expectation that as long as the state of system interaction is within the current region of user expectation, the user will be satisfied with adaptation. If the system's interaction falls outside of this region of expectation then user satisfaction will degrade. A more conservative user will have a smaller region of expectation and therefore less tolerance to adaptation. A more accepting user will have a larger region and greater tolerance to adaption. To prevent dissatisfaction the user must be given control over aspects of adaptation that limit changes in interaction state.

III. Human Factors of IDS

One of the most important human factors in security is that humans often select options that are the least cognitively demanding and provide maximal expected benefit [25]. In line with the definition of usability introduced in Section II, we consider user and task characteristics that are specifically related to IDS. In the following section we discuss how SH-based IDS could be suitable in light of these considerations.

A. User Characteristic

1) Memory: Current pattern-recognition technologies can only mimic a small fraction of what humans are capable of with our visual systems. Humans have exceptional visual intelligence that can recognize objects and patterns more easily than recalling them, without such prompting, from memory [26]. In the proposed framework, we can take the advantage of this human strength by reducing requirements for users to remember unnecessary details. Patterns can be represented or manipulated in objects or large semantic chunks.

2) Visibility: IDS must present several types of information about the current and past state of the network being monitored. Such information-rich displays have great potential to overwhelm users with data to be interpreted [27]. A good user interface design must have a proper visualization of the network or system status for humans to pick up visual cues of representations of the situations quickly while avoiding inundating them with details.

3) Confidence and Locus of Control: The target IDS users are experts in what they do. They will need to feel in control of the system at all times. The controls and responses of the IDS system has to be clearly understood by the users. Any unclear actions performed by the system may lead to the system being considered untrustworthy (and therefore unsuitable) because the users do not feel in control of those actions.

4) Individual Differences and Task Type: What might otherwise be small differences between users are magnified in situations with open-ended (i.e., not straightforward) tasks, stress, or both. Recognizing and responding to unauthorized network use in real-time is both open-ended and potentially stressful. Therefore tools which can suit many working styles are necessary.

B. Task Characteristics

Intrusion Detection is part of a dynamic human adversarial system: Some humans attempt to gain unathorized access to networks, while other humans try to detect and prevent such access. The techniques used by both groups changes over time in response to the actions of the other. The dynamic properties of IDS are one of the main reasons why purely computational approaches are unlikely to be wholly successful.

1) Dynamics: The process of IDS is dynamic in at least two ways: new data constantly arriving, and emerging patterns arise over time. Those patterns cannot be recognized solely by software because of their complexity and rate of arrival. Intrusion detection systems that depend only on software pattern matching are liable to slow attacks, which occur over a longer time span than the software's monitoring window. Purely human-based systems cannot cope with the deluge of data either. All of the IDS professionals we interviewed described pattern recognition as a key feature of their job. A major part of IDS is recognizing deviation from normal patterns. But normal patterns shift over time. A simple example: software can classify overall network traffic patterns as normal for a weekday, but they cannot easily recognize when a weekday is a holiday. Aside from the obvious user-based reasons, humans are needed in IDS because patterns change over time, and the patterns are not obvious to software.

2) Pattern Recognition and Representation: IDS is, in part, about a complex synthesis of raw data into knowledge in the mind of the user. As more knowledge is created, it becomes easier to classify incoming data because that data will be identified with data that was already converted into knowledge, i.e classified. However the synthesis of knowledge will also require that existing knowledge be reconsidered and reorganized. Intrusion Detection systems manifest that knowledge outside of the users mind.

To be useful, an IDS must support users by accurately manifesting their knowledge, and making reorganization of that manifestation straightforward.

The following analogy may help to illustrate the concept. Data about incoming network traffic are like notices on a public bulletin board for announcements. When there are few announcements, then there is no discernable pattern. After some time however, some notices are removed (because they are for events that have already occurred) and new notices are placed in the unused spaces on the board. But some notices are so important that they remain fixed for long periods, and are only moved when the entire board is rearranged. In terms of understanding, those notices that are rarely moved represent major concepts that serve to delineate the structure of the knowledge. Because the patterns of data are neither obvious nor uniquely classifiable, the users' representation must be much more flexible than the bulletin board analogy.

IV. Theory: An IDS Framework based on SH & UAI

A. Applicability of SH and UAI for IDS

Current IDS have unacceptably high false alarm rates [28], [10]. Many suspicious events are logged as it is better to error on the side of caution. Consequently, the systems tend to overwhelm users with data. To be effective, network security managers must be able to interpret and incorporate data from the IDS interface to form knowledge of the current and long-term state of the network. An interface that allows a user to quickly interpret suspicious network events, *in the context* of large set of such events, will facilitate the appraisal of the true threat level and build a long-term view of the state of the network.

We believe SH is a suitable interface model for an IDS because SH systems have been developed for similar applications. We discus the particular advantages of SH for such a task below, but first we introduce our vision of how SH could be used as an interface for IDS.

B. Proposed Framework

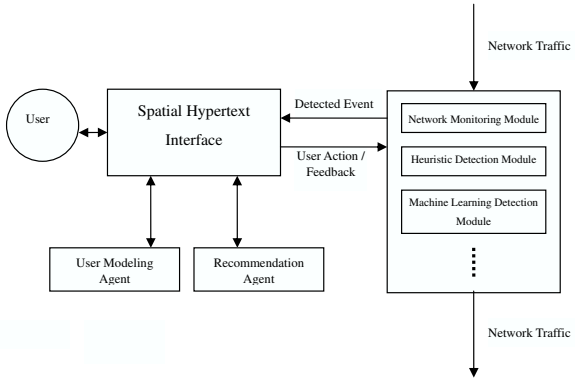

Fig. 2. The overall system structure

We see an adaptive SH interface being used in combination with a Machine Learning (ML) based networking monitoring system [29] that is part of a network firewall. The ML component will partition traffic events into three groups: safe, dangerous, or suspicious. The SH interface must keep the user aware of the current state of the network and assist the user in distinguishing which of the suspicious traffic events poses a true threat and which should be allowed to pass through the firewall.

The architecture of a potential system is displayed in Figure 2. On the left is the SH interface and supporting agents, on the right is the automated ML-based network monitoring system. The SH has three components (shown in the diagram as discs labeled as "agents").

1) SH Interface: Any realistic ML-based monitoring system will always incorrectly identify some of the dangerous and suspicious events incorrectly. This is due to the changing nature of network traffic and the types of attacks that can occur. The goal of the SH Interface is lessen the impact of this deficiency by providing: (a) a UI that can tolerate false detections, and (b) a UI that enables users to see the 'big picture' of network activity.

The SH interface will be used to visualize the dynamic state of network traffic. Objects (network events) will be clustered by spatial cues. Safe, dangerous and suspicious events passed from the monitoring system will be displayed as per the the directions of the User Modeling Agent. For example, the User Model may direct the display of Trojan-like events to a particular area of the screen for the current user.

Because of the characteristics of our users and their tasks, the interface will need to be interactive: Users must be able to change any of the presentation features of the display to help interrogate the data, to see patterns, and to focus on particular aspects. We expect that some form of multi-focus fisheye or focus-in-context display will be appropriate [30].

2) The User Modeling Agent: The task of the UM Agent is to develop and manage user models. A user model is developed from training data that comes in the form of event of currently displayed events (as provided by the network monitoring system) and their spatial cues as manually established by the user. In this way the user model will reflect the working habits and cognitive map of a particular user. Initially for a new user a user model will not be known.

The initial spatial cues for events could be based on a standard user model or the user's interaction with training set of simulated network traffic events. The UM Agent must also be capable of saving and restoring a user model for a particular user on demand.

Human interaction with a system will change over time. For example, a novice user will become an experienced user after a few weeks. This is generally known in UM and ML as "concept drift." The UM Agent must take into account of the temporal nature of the user's experience. A time window can be used for capturing data for constructing a user model from the most recent data. We suggest that either an Inductive Decision Tree or kNN machine learning technique would produce a good user model that maps event features to partial cues. A kNN approach would work particularly well as each new event can be added incrementally to the user model.

As there will be an overwhelming amount of data at times, the SH interface must present information in a relatively consistent manner but in concert with changes in the users working style and environmental factors [16]. This is where the theory of user expectation and adaptation comes into play. The user must be able to manipulate the degree to which the UM Agent can dynamically manipulate the spatial cues. If the interaction state of the IDS varies outside the users region of interaction expectation, the system will not be consider usable. There are methods of controlling spatial cuing within the spatial parsing literature on advanced SH systems. Appropriate control over adaptation based on a user model will require further research.

3) The Recommendation Agent: The Recommendation Agent will generate suggestions to help the user arrange the spatial cues of event objects and clusters of objects. It will also ensure that specific events catch the user's attention so there are no unattended warnings or alarms.

In the SH workspace of our IDS, the detected network events will be presented in a layout according to the user model. As the user moves the objects and adjusts other spatial cues, it may be difficult or inefficient to group or align objects manually. The Recommendation Agent can detect such actions and ask the user if he or she wants the objects to be grouped or aligned automatically.

In many cases, the objects displayed in the SH workspace have special meaning, but the user may not sense them. If a potential threat has been detected, but not dealt within a specified period, then the agent could prompt the user to analyze the situation and take appropriate action. Low priority recommendations can be prompted to the user at less central screen locations; for example in the tool bar areas of many GUI applications. We expect that this positioning would avoid unnecessary obstruction on the part of the agent. Assistance on the arrangement of the objects would be an example of

low priority recommendation. Higher priority recommendations, due to something like one minute passing without user response to an alarm, could trigger a more apparent signal in an attempt to attract the user's attention.

V. Initial Assessment

A basic tenant of user-centered design is that at every stage of development there should be evaluation with respect to user needs if not actual testing with users [31]. At this early stage we evaluate the ability of our proposed design to support only the specified users (administrators).

First we consider how SH, and its particular application to IDS, can be expected to suit users-in-general before we discuss particular advantages for the specific user group we identified earlier.

A. General Characteristics

Because it can be used to represent many types of relationships between many different objects, SH can be a powerful medium for expression. Furthermore because the mode of interaction with SH is direct manipulation [32] of objects and relations, there are several advantages:

- the display and methods of operation are flexible,
- the operations are familiar since most human are used to manipulating objects in space [33]
- information is shown in context,
- users the power of recognition rather than recall which reduces the cognitive load needed to operate the system [26],
- users maintain a sense of control, and
- the UI harnesses humans' powerful visual-spatial cognitive ability particularly in recognizing high-level patterns [13], [34].

B. Specific Characteristics

Our target users are at domain experts (in computer network system administration).

1) Reduction of False Alarms, Increase in IDS Trust: An IDS receives and must display a large amount of information on detected events. False alarms triggered by suspicious events can reduce a users trust of the system. SH can overcome this problem by representing the differences between events as differences in spatial cues for each object in a spatial hypertext workspace. In addition to identifying high probability alarms, the system will help the user identify suspicious events by using SH objects. The user will be able to detect the subtle differences and relations among the objects and thereby better demarcate the true threats from false alarms. In summary, a SH interface will utilize the users visual recognition ability to find the subtle differences among event objects.

2) Flexibility and Control: Such users will need to have controls of all the functions and controls when they need them. In a SH workspace, the flexibility and high level of control are offered at the same time. The user have the freedom to organize and manipulate the objects in the workspace to fit the user's own preferences. Furthermore, there are no limitations on how the user can manipulate the objects in the workspace.

SH workspaces can present the global context and preserve the local details at the same time. Providing the global context means that the administrator can have a better sense of the overall system status visually, and at the same time, the administrator can drill into the details of each group or object showing on the workspace. A more traditional approach might also have an overall system status displayed graphically, but it probably would not provide flexibility and extra functionality.

3) Inter-personal Communication: One of the most important characters of SH is the "Constructive Ambiguity" that leaves much room for users to form different interpretations of the same interface. We expect that each user will manipulate the interface in their own way to suit their own conceptual models, which will also lead to a more customized and usable user interface.

As we discussed in Section II-A.2, our target user population operate mostly as introverts but also work in groups. A specific advantage of SH for this population is that by being able to represent ambiguous and implicit relationships, SH has been found to aid inter-personal communication without impeding understanding or flexibility [11].

4) Seeing "The Big Picture": A major advantage of SH over many other interface styles is its ability to represent ambiguous and implicit relationships. This fact alone results in efficient communication on a group of collaborators and allows the users to form new interpretation, which makes SH suitable for information-intensive work environment [15]. The ambiguity in the information structure allows new insights and relationships emerge as the users work through the materials [11].

VI. Conclusion

Intrusion detection systems must handle rapid (often real-time) masses of information so as to report the abnormal use of networks and computer systems. IDS users have particular personality characteristics and job needs. In particular, they must recognize developing patterns in large quantities of data, they prefer to work individually (although they must function in collaborative groups) and remain in control of the system, and they rely on their intuitions more than deductive logic. Most significantly, an automated monitoring system that accurately detects intrusions will error on the side of false alarms. Therefore a user interface for an IDS must mitigate the impact of false alarms by leaving the final decision on suspicious events up to the user.

The IDS architecture proposed in this article is based on spatial hypertext, adaptive user interfaces and user modeling. Spatial hypertext was developed to handle information overflow particularly in situations where the data must be interpreted quickly. SH has proven to be effective for dynamic information analysis tasks and intrusion detection is an information intensive and deeply analytic process that cannot be undertaken without the assistance of a computer. Adaptive UI improves the usability and performance of the IDS by

automatically marking up inbound events with relevant spatial cues. A user model is developed by using machine learning technology to map the features of the detected event to the most likely spatial cues desired by the IDS user. The beauty of this approach is that the user model is idiosyncratic to the current user, will adapt over time and its impact on adaptation can be controlled so as maintain user expectations.

The SH and Adaptive UI design allows for a nice mix of interface flexibility and user control. The article discussed why spatial hypertext, and the proposed architecture in particular, can well fulfill the needs of intrusion detection system users. Using the criteria we established in Section II regarding the usability of a system, and applying knowledge of the specific user population, studies of the use of spatial hypertext (SH) and intrusion detection systems (IDS) we conclude that SH has many of the necessary characteristics of an ideal user interface (UI) for network security administration. Our evaluation was based on studies of work habits and personality profiles of such users as well as an analysis of general tasks.

The proposed IDS is at an early stage of development, however, we feel that a system based on a combination of SH and UM has promise. The architecture seems likely to provide more than acceptable levels of efficiency, efficacy, satisfaction, learn-ability and memorability. Our next steps are to further investigate the system's requirements through joint analysis and prototyping sessions with IDS users. After the system requirements are refined, a full IDS will be designed, implemented and tested.

REFERENCES

[1] "Health insurance portability and accountability Act (HIPAA), public law," 104-191 104th [US] Congress, 21 Aug. 1996. [Online]. Available: \url{http://aspe.hhs.gov/admnsimp/pl104191.htm}

[2] "Personal information: Privacy SB 1386, (California Disclosure Law)," California State Legislature, 12 Feb. 2002. [Online]. Available: \url{http://info.sen.ca.gov/pub/01-02/bill/sen/sb_13511400/sb_1386_bill_20020926_chaptered.html}

[3] A. T. Zhou, J. Blustein, and N. Zincir-Heywood, "Improving intrusion detection systems through heuristic evaluation," in *IEEE Canadian Conf. on Electrical and Computer Engineering (CCECE)*, 2004, pp. 1641 – 1644.

[4] A. Zhou, J. Blustein, and N. Zincir-Heywood, "The state of network security management: Issues and directions," Dalhousie University Faculty of Computer Science, Technical Report CS-2003-06, May 2003. [Online]. Available: http://www.cs.dal.ca/research/techreports/2003/CS-2003-06.shtml

[5] A. Whitten and J. D. Tygar, "Why Johnny can't encrypt: A usability case study of PGP 5.0," in *USENIX Security*, 1999.

[6] J. McHugh, A. Christie, and J. Allen, "Defending yourself: The role of intrusion detection systems," *IEEE Software*, pp. 42 – 51, Sept./Oct. 2000.

[7] *Part 11: Guidance on usability*, International Organization for Standardization Std. 9241, 1998.

[8] A. Dillon, "Beyond usability: Process, outcome, and affect in human computer interactions," *Canadian Journal of Information Science*, vol. 26, no. 4, pp. 57–69, Dec. 2001.

[9] C. Gates and T. Whalen, "Profiling the defenders," in *New Security Paradigms Workshop*, 2004.

[10] D. J. Marchette, *Computer Intrusion Detection and Network Monitoring: A Statistical Viewpoint*, ser. Statistics for Engineering and Information Science. New York: Springer-Verlag, 2001.

[11] F. M. Shipman and C. C. Marshall, "Spatial Hypertext: An Alternative to Navigational and Semantic Links," *ACM Comput. Surv.*, vol. 31, no. 4es, p. 14, 1999.

[12] C. C. Marshall, F. G. Halasz, R. A. Rogers, and J. William C. Janssen, "Aquanet: a Hypertext Tool to Hold Your Knowledge in Place," in *ACM Hypertext*. ACM Press, 1991, pp. 261–275.

[13] C. C. Marshall and F. M. Shipman, "Spatial Hypertext: Designing for Change," *Commun. ACM*, vol. 38, no. 8, pp. 88–97, 1995.

[14] C. C. Marshall, F. M. Shipman, and J. H. Coombs, "VIKI: Spatial Hypertext Supporting Emergent Structure," in *ACM European Conf. on Hypermedia Tech.* ACM Press, 1994, pp. 13–23.

[15] C. C. Marshall and F. M. Shipman, "Spatial Hypertext and the Practice of Information Triage," in *ACM Hypertext*. ACM Press, 1997, pp. 124–133.

[16] J. McGrenere, R. M. Baecker, and K. S. Booth, "An evaluation of a multiple interface design solution for bloated software," in *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, 2002, pp. 164 – 170. [Online]. Available: http://doi.acm.org/10.1145/503376.503406

[17] A. Kobsa, *Generic User Modeling Systems*. Kluwer Academic, 2001, pp. 9–63.

[18] G. I. Webb, M. J. Pazzani, and D. Billsus, *Machine learning for user modeling*. Kluwer Academic, 2001, pp. 19–29.

[19] N. Ducheneaut and V. Bellotti, "E-mail as habitat: An exploration of embedded personal information management," *interactions*, vol. 8, no. 5, pp. 30–38, 2001.

[20] W. E. Mackay, "Triggers and barriers to customizing software," in *ACM CHI*, 1991, pp. 153–160.

[21] G. Fischer, *User Modeling in Human-Computer Interaction*. The Netherlands: Kluwer Academic, 2001, pp. 65–86.

[22] J. Kay, "Learner control," in *roc. User Modelling and User-Adapted Inter.*, 2001.

[23] A. Jameson and E. Schwarzkopf, "Pros and cons of controllability: An empirical study," in *Proc. of Adaptive Hypermedia*, 2002.

[24] X. Peng and D. L. Silver, "User control user adaptation: A case study," in *Tenth Int'l. Conf. on User Modelling (UM 2005)*, Edinbugh, Scotland, 24 – 30 July 2005, (in press).

[25] D. Besnard and B. Arief, "Computer Security Impaired by Legitimate Users," *Computer and Security*, vol. 23, no. 3, pp. 253–264, May 2004.

[26] J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland, and T. Carey, *Human-Computer Interaction: Concepts And Design*, 1st ed. New York: Addison-Wesley, 1994.

[27] T. S. Tullis, "An evaluation of alphanumeric, graphics, and color information displays," *Human Factors*, vol. 23, no. 5, pp. 541 – 550, 1981.

[28] G. H. Kayacik and A. N. Zincir-Heywood, "Case study of three open source security management tools," in *IFIP/IEEE Symp. Integrated Network Management*, 2003.

[29] D. Song, M. I. Heywood, and A. N. Zincir-Heywood, "Linear genetic programming approach to intrusion detection," in *Genetic and Evolutionary Computation*, 2003.

[30] Y. K. Leung and M. D. Apperley, "A review and taxonomy of distortion-oriented presentation techniques," *ACM Trans. on CHI*, vol. 1, no. 2, pp. 126–160, 1994.

[31] D. Hix and H. R. Hartson, *Developing User Interface Ensuring Usability Through Product & Process*, ser. Wiley Professional Computing. John Wiley & Sons, Inc., 1993.

[32] B. Shneiderman and C. Plaisant, *Direct Manipulation and Virtual Environment*, 4th ed. Addison Wesley, 2004, ch. 6, pp. 214 – 216.

[33] A. Cockburn, "Revisiting 2D vs 3D Implications on Spatial Memory," in *Australasian User Interface*. Australian Computer Society, Inc., 2004, pp. 25–31.

[34] F. M. Shipman, C. C. Marshall, and T. P. Moran, "Finding and Using Implicit Structure in Human-Organized Spatial Layouts of Information," in *ACM CHI*. ACM Press/Addison-Wesley Publishing Co., 1995, pp. 346–353.

[35] G. Marchionini, "Information-seeking strategies of novices using a full-text electronic encyclopedia," *Journal of the American Society for Information Science*, vol. 40, pp. 54 – 66, Jan. 1989. [Online]. Available: http://dx.doi.org/10.1002/(SICI)1097-4571(198901)40:1<54::AID-ASI6>3.0.CO;2-R