The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness

A Review of the Published Literature and Discussion of Future Research Plans

Nicklaus A. Giacobe
Ph.D. Student
College of Information Sciences and Technology
The Pennsylvania State University
University Park, PA 16802
814-863-8555
nxg13@ist.psu.edu
This page left blank intentionally
# Contents

Table of Figures .................................................................................................................. ii

Introduction .......................................................................................................................... 1

Section 1 – The Classical Roots of Intrusion Detection ......................................................... 3
  - Host-Based IDS ............................................................................................................... 3
  - Network-Based IDS ....................................................................................................... 4
  - Anomaly Detection ........................................................................................................ 5
  - Misuse-Based IDS and Active DoS Prevention ............................................................... 7
  - Testing of IDSs ............................................................................................................... 8

Section 2 - Data Fusion in Intrusion Detection Systems ....................................................... 10
  - Alert Correlation ........................................................................................................... 10
  - A Review of the JDL Fusion Model ............................................................................... 12
  - Bayesian Inference ....................................................................................................... 14
  - Dempster-Shafer (D-S) Theory of Evidence as a Fusion Engine .................................. 14
    - D-S Theory in Intrusion Detection .............................................................................. 17
    - D-S Theory in Practice in Intrusion Detection .......................................................... 18
    - Evaluation of D-S Theory Based Systems ................................................................. 20
  - Neural Networks and SVMs ......................................................................................... 20
  - Summary of Data Fusion Techniques in Intrusion Detection ........................................ 21

Section 3 – Situational Awareness in Cyber Security ........................................................... 23
  - What is Situational Awareness? .................................................................................... 24
  - Cognitive Load Issues in System Design ...................................................................... 28
  - High Level Fusion ......................................................................................................... 29
  - CTA of Computer Network Security Analysts ............................................................. 31
  - Visualization in Support of SA ...................................................................................... 34
    - Network Topology-Based Displays ........................................................................... 35
    - Network Traffic-Based Representations .................................................................... 39
    - IDS-based and Track-Based Displays ...................................................................... 41
  - Current State SA in Cyber Security .............................................................................. 42

Section 4 – Future Research Plans ...................................................................................... 44
Table of Figures
Figure 1 - DIDS to JDL Model Comparison - Adapted from (Snapp et al. 1991) and (Hall et al. 2004) .......................... 5
Figure 2 - Alert Aggregation Methods - adapted from Debar and Wespi (2001) ................................................................. 11
Figure 3 - The Joint Director of Laboratories (JDL) data fusion process model adapted from Hall et al. (2004) ................................................. 12
Figure 5 - Dempster Rule combining Sensor Si and Sj’s observations (m) from Wu et al. (2002)................................. 16
Figure 6 - Weighted D-S Combination with Wi and Wj from (Wu et al. 2002) ................................................................. 16
Figure 4 - Confidence Interval, Belief and Plausibility Functions from Wu et al. (2002) ......................................................... 16
Figure 7 - Sample Data from a D-S Theory-Based IDS - Tables 1-3 from (Tian et al. 2005b) ........................................ 19
Figure 8 - Levels of Situation Awareness adapted from Endsley (1995) ................................................................. 25
Figure 9 - Mapping of IDS Fusion tasks between JDL Model and Endsley SA Model. From Yang et al. (2009) .............................................. 30
Figure 10 - Data Reduction Processes through the Work of an Analyst - from D’Amico et al. (2008) .................. 33
Figure 11 - Representation of BGP data in BGPeep from (Shearer et al. 2008) ................................................................. 35
Figure 12 - Examples of Root Polar Plot (left) and IP addresses represented on this style of plot (right) from (Fink et al. 2005) ............................................................................................................. 36
Figure 13 - Hierarchical Network Map from Mansmann and Vinnik (2006)................................................................. 38
Figure 14 - Representation of Threats and Actors on a Geopolitical Map from (Pike et al. 2008) .................. 38
Figure 15 - Representation of host to port to remote port to remote host of network traffic from (Fink et al. 2004) ............................................................................................................. 39
Figure 16 - Connection Map (left) and Treemap (right) from (Blue et al. 2008) ......................................................... 40
Figure 17 - Panel Displaying Network Connections from a Single Host from (Fischer et al. 2008) .................. 41
Figure 18 - Representing the Three Ws from (Foresti et al. 2008) ................................................................. 42
Introduction

Intrusion Detection (ID) plays a significant role in network security situational awareness. Intrusion Detection Systems (IDSs) hold the promise of being able to monitor the activity of users, applications, viruses, trojan horses, worms and other malware. Whether the IDS is host-based or network based, whether it contains anomaly detection or signature-based pattern matching technologies, it will provide alert data which can be a component of situational awareness.

Section 1 of this review begins with a discussion of the historical basis of ID. A number of IDS systems will be reviewed from the published literature to orient the reader to how IDSs were intended to operate. This section will conclude with an assessment of current technologies in ID.

In 2000, Tim Bass introduced the concept that multi-sensor data fusion could be used to combine information from IDSs to create cyberspace situational awareness. His assessment of data fusion identified its utility in other fields including military battlefield situational awareness, manufacturing and remote sensing (Bass 2000). For the past nine years, a number of researchers have followed Bass’s call and attempted to implement data fusion techniques with IDSs. Section 2 will begin with a review of alert correlation mechanisms and will continue with a review the Joint Directors of Laboratories (JDL) model of data fusion. A variety of data fusion algorithm research contributions follows, chronicling their successes in the implementation of data fusion technologies in the application domain of intrusion detection.

Section 3 will address the issue of situational awareness in network security. Beginning with an understanding of theoretical discussion of what security is and what awareness is, the section continues with a comparison of situational awareness in more traditional security-related fields. What is obvious is that practitioners of security management are tasked with data and information from a variety of sources. A discussion of cognitive load orients the reader toward understanding that cyber security
support systems need to reduce and organize data and information so that the security analyst can easily understand it and take appropriate action. Higher levels of fusion and awareness provide not only an understanding of the events that are currently taking place on the network, but also a projection of what is likely to happen in the future. However, few automated systems have been proposed to create these higher levels of awareness. The current status of fusion and awareness development is represented in the daily work of network security analysts. How analysts do their job can be elucidated through performing cognitive task analysis (CTA). However, very little observation, interview and general inquiry has been performed and published in the research. It is no wonder that there has been limited success and adoption of fusion and awareness technologies, even the potentially promising ones. The section concludes with a discussion of the recent developments in visualization techniques for cyber security.

This literature review will wrap up with Section 4, a proposed plan of future research topics. First presented is a model for understanding the contributions of researchers who are developing data fusion algorithms. Next is a suggestion for the development of a more rich set of features and needs based on CTA of current network security analysts at a variety of levels in an organization. Finally, visualization tools are discussed and a suggestion is made to match the tools to the data fusion and awareness levels.

Cyber and network security is a complex task. The current networked computing environment is based on insecure operating systems, complex software from multiple vendors, quick and dirty system deployment and ubiquitous high-bandwidth network connections. It is threatened by blended attacks, monetization of hacking and exploitation, and the easy access to exploitation tools. Needless to say, an organization’s network security mission is difficult to staff, develop and maintain. Tools must be developed from both the data/algorithm (bottom up) and the human analyst (top down) perspectives to achieve better situational awareness.
Section 1 – The Classical Roots of Intrusion Detection

Intrusion detection (ID) is the process of identifying the activities of unauthorized users on a computer system or network. There are two different modes of intrusion detection – one is host-based and the other is network based. Host-based ID assumes that the computer system (host) has the capability to securely log activity that has taken place on it. Network-based ID monitors network traffic from the host to note activities that would indicate that the system has been used without authorization and generally reports on activities such as frequent, incorrect attempts at authentication from across the network, signs of virus, trojan or worm activity and other similar signals. There are also two different methods that IDSs use to identify interesting behavior. Misuse detection attempts to compare activity to existing and known signatures of bad traffic. Anomaly detection attempts to identify the intrusion through the identification of traffic that does not match the known normal patterns. From Denning’s seminal description of IDS in 1987 until approximately 2000, IDSs could be classified using these descriptions. The height of IDS development along these lines can be marked by the 1998 – 2000 DARPA IDS data sets that were provided by the U.S. Government to the IDS industry for the development and testing of various IDS solutions.

Host-Based IDS

Denning reported a generic model of an intrusion detection system in 1987. Her work included the concept of abnormal use monitoring and documented a number of different user types, user behaviors and various malware technologies including virus and trojans. The language that is introduced by Denning includes a differentiation between the users, processes and objects in a system. Denning’s model was for an Intrusion Detection Expert System (IDES) and provided the industry with a first standard anomaly record, containing an event, time-stamp and profile (Denning 1987). A number of other standards for intrusion detection records are presented, but most importantly, this early research provides a nomenclature for the discussion of intrusion detection that is used throughout the literature.
The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness

One of Denning’s colleagues at SRI expanded on IDES and implemented it as a statistical anomaly detector. IDES maintained a base of known normal behaviors about the subject and compared potential intrusion data to that known behavior. The statistical behavior is measured each day and new behavior is compared to the statistical norm of the previous behaviors (Javitz et al. 1991). An actual implementation of this statistically based algorithm is presented in a full description of Next Generation Intrusion Detection Expert System (NIDES) (Javitz et al. 1994).

One should note that this early work in intrusion detection is primarily performed on UNIX systems and observing processes executed by users. While Denning reports that the processes could be actual users at terminals, the actors could actually be independent code running on the machine in terms of a virus or trojan horse. While there was an interconnection of computers in networks from 1987 to 1994, the primary focus of IDS at this stage is host- and process-based.

**Network-Based IDS**

The concept of monitoring the actual network traffic was introduced by researchers at UC Davis with their introduction of the Network Security Monitor (NSM). The NSM monitored a singled broadcast domain (single shared medium LAN) and tracked intrusion detection parameters from the network traffic. Because all traffic crossing the LAN is available to all hosts on the network, it was relatively simple to monitor all network traffic. NSM assessed the probability that specific network traffic was an intrusion based on a variety of parameters, including previous historical connections between hosts, previous reported abuses from a specific host, signatures of previous attack types and the abnormality of a connection between the given hosts (Mukherjee et al. 1994).

Snapp et al. developed a unique perspective on intrusion detection through the monitoring of user behavior from across the network using the network traffic itself. They combined the network audit data from C2 compliant hosts on the network in conjunction with the LAN monitor (based on NSM, above) – which keeps track of all data traversing a single network segment. This system aggregates the
alert data, and notable raw host-based log data in a single location for combination and analysis. This represented early fusion of network-based IDS and host-based IDS (C2 log monitoring) processes.

The expert system (NIDS) described in (Snapp et al. 1991) is based on varying layers of their intrusion detection model (IDM). Each level builds on the previous, adding additional inference capability. While there is not a direct correspondence to the JDL Data Fusion model, the concepts are very similar.

<table>
<thead>
<tr>
<th>DIDS Layer</th>
<th>DIDS Layer Description</th>
<th>JDL Model Level</th>
<th>JDL Model Level Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Audit records (data)</td>
<td>0</td>
<td>Source pre-processing (data alignment)</td>
</tr>
<tr>
<td>2</td>
<td>Event</td>
<td>1</td>
<td>Object refinement</td>
</tr>
<tr>
<td>3</td>
<td>Subject</td>
<td>2</td>
<td>Situation Refinement</td>
</tr>
<tr>
<td>4</td>
<td>Event in Context</td>
<td>3</td>
<td>Threat Refinement</td>
</tr>
<tr>
<td>5</td>
<td>Threats</td>
<td>4</td>
<td>Process Refinement</td>
</tr>
<tr>
<td>6</td>
<td>Security State Assessment</td>
<td>5</td>
<td>Cognitive Refinement</td>
</tr>
</tbody>
</table>

Figure 1 - DIDS to JDL Model Comparison - Adapted from (Snapp et al. 1991) and (Hall et al. 2004)

One will note that there is not a direct level to layer mapping between the two models. The DIDS “Event Layer” could be interpreted in the JDL model as either Source Pre-Processing or Object level refinement. While the JDL Object level refinement may address both the subject and event layers introduced in DIDS.

**Anomaly Detection**

Lee, Stolfo and Mok outlined a framework for building an IDS. Their framework was based on classification, link analysis and sequence analysis. Primarily, their work was an effort to classify data into either normal behavior or match it against known patterns of types of intrusion. They also introduced the concept of the need for multiple lightweight IDSs that work in a cooperative manner. However, they do not describe any processes that should be used to create inferences about the information in the
data, instead provide only for simple division of labor to reduced performance bottlenecks (Lee et al. 1999).

Evaluation of their system was performed using the DARPA Intrusion Detection Evaluation program from MIT Lincoln Labs. This evaluation included a set of training data as well as some data to perform the evaluation. The data was provided with four categories of simulated attacks including Denial of Service, Remote to Local, User to Root and Probing, each indicating a typical type of attack that was common at the time (Lee et al. 1999).

Portnoy, Eskin and Stolfo identified a problem with the two major types of intrusion detection systems. Anomaly detection requires baseline data that is free from intrusions to represent the normal behavior from which to report deviations. Misuse detection requires knowledge of every possible intrusion method from which to develop signatures to compare against. Creating these data sets manually is virtually impossible. They proposed an unsupervised anomaly detection method indentifying the intrusion data as qualitatively different from the normal data. The qualitatively similar data is clustered and small clusters of data are identified as abnormal. This initial method had about a 50% detection rate with a low false positive rate. They identified that this work showed promising results because of its lack of need for pre training and a priori knowledge (Portnoy et al. 2001).

More recent work includes automated detection using statistical methods on the payload of the packet rather than simply the headers. They identified that the contents of the packets of worms, especially, have relatively unique properties. Payloads of normal traffic are compared, paying attention to the frequency distribution of the payload size. This represents an effective mechanism for detecting worm traffic across a network, so that it can be blocked automatically by network devices before it gets to target computers such as web servers and other critical network nodes (Wang et al. 2004). The most recent work on this line of research is the automation of the development of signatures for signature-
based IDS that evaluates the payload (Wang et al. 2006a), even if the attacker has gone as far as to intentionally obfuscate their attack traffic by mimicking the normal traffic patterns of the victim’s network (Wang et al. 2006b). Building on that work, they also developed a hybrid IDS that includes automated detection and protection against zero-day worm attacks (Locasto et al. 2006).

This body of work has significant importance as it represents the ability of algorithms to independently evaluate network traffic for anomalies. Using statistical methods to analyze the content of network packets allows the IDS sensor to operate independently of human input and without a priori knowledge of the specific details of an attack method. Pattern matching for known exploits may be the most efficient way to determine if those exploits exist. However, the automated development of new signatures for unknown, zero-day and polymorphic exploits has great potential. Allowing for the automated creation of new signatures brings the best of signature-based and anomaly detection IDS technologies.

**Misuse-Based IDS and Active DoS Prevention**

Eugene Spafford’s work in misuse detection starts with matching network traffic to signatures of known misuse characteristics. However, in his early work, he identified that for specific exploits to be utilized, they must also satisfy certain pre- and post-conditions. This state transition analysis model was anticipated to be useful in both host-based (log-based) and network-based intrusion detection. Using colored Petri Nets, they implemented a prototype IDS and evaluated log files from Unix systems, but identify that there could be more applications in network-based IDS as well (Kumar et al. 1994; Kumar et al. 1995).

The use of Tripwire for intrusion detection and the associated issues are outlined in (Kim et al. 1994). This research outlined the use of Tripwire’s capability to use message digest (MD) hashing of the files on a UNIX file system to compare them in a secure manner to known previous good versions of the same files. Using MD hashes avoids some problems with file system integrity checks such as file size, file
name and date last saved or changed. These values can be changed and modified in such a way as to obscure the fact that the actual file may have changed. However, the message digest signature is calculated in a way that it is computationally improbable to create a useful (to a hacker) change to a file and also make the change reflect the same hash value (Kim et al. 1994).

Schuba et al. reported some possible methodologies in dealing with SYN floods. A SYN flood is a particular kind of denial of service attack that utilizes valid network traffic in an inappropriate sequence and from fictitious source IP addresses to intentionally flood the victim computer system. The SYN packets are never answered in a normal handshake process, forcing the host system to waste resources to wait for the remainder of these half-open connections. The solution was to put a firewall or firewall-like application as an intermediary between the protected host and the outside network. The firewall could act as a relay to handle the SYN/SYN+ACK part of the transaction and only forwarding on the completed, legitimate connections to the protected host. Handling this process is an active agent called “synkill”. This active process is based on observed network behavior and resolves the problem by injecting of RST packets to close half-open SYN connections from known or suspected malicious sources (Schuba et al. 1997).

**Testing of IDSs**

A number of other IDSs were developed and introduced in the mid 1990’s. These included IDES and NIDES (SRI International), MIDAS (NSA), USTAT (UCSB), NADIR (Los Alamos Labs), NSM and DIDS (UC Davis) (Puketza et al. 1996). This brought about the need for methodologies to test the effectiveness of the IDS algorithms and implementations. Puketza suggested that an IDS can be tested by simply creating batch processes of what intrusion behavior may look like, and then execute those scripts. In a UNIX environment, these methods make a lot of sense. However, they also indicate that the complexities of a user interfacing with a GUI are too complex for their solution to model (Puketza et al. 1996).
The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness

The pinnacle of classic IDS can be marked with the development of the DARPA datasets for IDS evaluation. From 1998 to 2000, The Defense Advanced Research Projects Agency (DARPA) through MIT Lincoln Labs released an annual data set that included background network traffic as well as traffic that represented actual network intrusions (2008; Lippmann et al. 2000a; Lippmann et al. 2000b). This raw traffic data was collected and anonymized, so that it could be used to design and evaluated network intrusion detection systems. Included in these sets were training data, intended to indentify known attacks, as well as the evaluation data. There is some concern that the use of these datasets will produce an IDS that is overfit for the intrusion types represented and will not perform effectively to identify novel attacks. However, these data sets do represent some capability to evaluate IDSs on a level playing field.
Section 2 - Data Fusion in Intrusion Detection Systems

To gain better situational awareness, the individual alerts from multiple IDSs must be aggregated, fused or correlated in some fashion. Methods of correlation can bring better situational awareness if the method provides some additional meaning to the data. For instance, alerts that identify a common source might provide information about the identity of the intruder, or at a minimum, provide the analyst with information crucial to the defense of the network, such as blocking all traffic from the source address. Alerts that have a common destination may provide information about the vulnerabilities on a given host in the defended network. Alerts with a common attack signature may provide information about the types of attacks in use and suggest methods for defense, or most certainly suggest where the analyst should pay attention. The process of alert correlation is non-trivial. It is not as simple to aggregate data based on these simple factors in a large scale network and even more difficult to do so in real-time. In this section, early alert correlation methods are described. Next, the JDL Model of Data Fusion is presented to develop common ground for understanding some of the more recent contributions. This section concludes with a description of recent contributions of modern data fusion algorithms suggested in the literature.

Alert Correlation

Cuppens identified that the use of multiple IDSs can be beneficial because different systems can interpret the data stream in different manners. The output from a behaviorally-based IDS can be fused with the alert data from a signature-based system, output from a scenario-based IDS and yet another IDS based on abnormality detection. These approaches can be complementary if their alert data can be fused to provide confirmation and additional supporting evidence, or reduce and eliminate false positives. Unfortunately, different IDSs report their findings in different formats with different data representing similar events. For example, they might report IP addresses or hostnames or even when those are in agreement, different systems might report either service names or TCP/UDP port numbers.
Using a temporal limit, Cuppens’ suggestion was to fuse the data based on source identifying information (IP or hostname), destination identifying information (IP or hostname) and service identifying information (service name or port number) (Cuppens 2001).

In the absence of standards, industry will develop its own mechanisms for the need to merge dissimilar formats. Debar et al. developed a system based on IBM Tivoli technology to aggregate alert data from dissimilar intrusion detection systems. Treating each individual IDS as a probe or sensor, a variety of different system types can be employed to provide a different perspective on the attack. Sensors that met their system’s standard interface could integrate directly, but other IDSs that used different formats needed pre-adapters to integrate their output and translate it to the system standard format. This process of increasing the number of systems increases the total volume of alerts that must be handled. The goal of this system was to generate one alert per attack, even if the attack has multiple individual alerts associated with it. Duplicate alerts were noted and removed. Alerts that are part of a multi-stage attack process can be related and linked if they occur within a specified time window. Finally, alerts that are otherwise aggregated by their common characteristics are represented in “situations” (Debar et al. 2001). These common characteristics are identified in the table below.

<table>
<thead>
<tr>
<th>Situation</th>
<th>Combination</th>
<th>Implication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Situation 1</td>
<td>Same Source, target and alert class</td>
<td>Single attacker against same host</td>
</tr>
<tr>
<td>Situation 2-1</td>
<td>Same source and destination</td>
<td>Single attacker on same host, possibly using varying attack methods</td>
</tr>
<tr>
<td>Situation 2-2</td>
<td>Same target and same alert class</td>
<td>Distributed attack on a single host</td>
</tr>
<tr>
<td>Situation 2-3</td>
<td>Same source and same alert class</td>
<td>Single attacker using the same attack and trying to find any host vulnerable to that attack</td>
</tr>
<tr>
<td>Situation 3-1</td>
<td>Same source only</td>
<td>Single attacker using a variety of attack methods on a variety of hosts</td>
</tr>
<tr>
<td>Situation 3-2</td>
<td>Same target only</td>
<td>Distributed attacks</td>
</tr>
<tr>
<td>Situation 3-3</td>
<td>Same attack class only</td>
<td>Common or novel attack method in use by many attackers</td>
</tr>
</tbody>
</table>

Klaus Julisch from IBM suggested that alert data could be aggregated by identifying the root cause of the alert, rather than paying specific attention to the details of the alert itself. This work identified that a significant number of alerts can be attributed to a much smaller number “root causes”
that persist until the analyst takes some action to remove the cause. A reduction of 87% of the alert volume is suggested by simply dealing with the underlying causes and eliminating the vulnerabilities that are implied by the alert data. A monthly root cause analysis and action of removal is recommended. The contribution of this work is in the concept of how to aggregate the alerts. Using a heuristic algorithm, alerts are aggregated based on their similarities (Julisch 2003).

Formalized logic and representation can help a combined alert aggregation system to overcome the limitations of the underlying IDS alert data formats. Morin and colleagues provided an abstraction layer to represent network architectures, software products, vulnerabilities, security tools and other actors, devices and systems that are in the network security environment (Morin et al. 2002). Later, Morin and Debar proposed the use of chronicles to reduce the number of alerts. Alerts are compared to patterns of multi-event signatures. Under the defined constraint of time, linked sequences of events are created from the alert data. These streams of events are compared to known signatures of attack patterns. If there is a match, an alert can be sent to the user (Morin et al. 2003).

**A Review of the JDL Fusion Model**

There are many models that describe different implementations of data fusion. This paper adopts the JDL Model as a reference model to understand the contributions of researchers.
Source pre-processing (or Level 0 fusion) automates the process of controlling the data that enters the fusion system. The rate of alerts from the various sources and sensors would overwhelm the processing power and capability of the fusion engine. Feature reduction and filtering is required to eliminate the bulk of the data from the sensor and relieve the fusion engine from having to sort and classify the information from the sensor itself (Hall et al. 2004). In the IDS MSDL, Level 0 fusion is a function of the individual IDS reporting mechanism. The raw packet data is digested and analyzed. If that data represents a possible intrusion, the IDS will generate an alert, reporting only the pertinent information. As alerts are generated, they are generally reported to the analyst or aggregated with other similar alerts. The format of the alert is a question for Level 0 fusion. If the various IDSs in the system use a standard reporting format, Level 0 is straightforward. However, time differences need to be addressed if the sensors are not synchronized.

The purpose of object refinement (Level 1 fusion) is to identify characteristics entities being sensed. It takes input from the multiple sensors and aligns, associates, tracks and identifies these entities from uncorrelated raw data (Hall et al. 2004). Primarily in ID, this is the correlation of alerts from multiple IDSs into “tracks” by combining alerts from the same source, to the same destination or using the same techniques in a defined period of time. This level of fusion represents the ability to identify a single of an attack even though the IDS may represent a multi-stage attack as multiple alerts.

Level 2 fusion is situational refinement. Using some sort of inference engine, situational awareness must be developed based on the data from the low level sensors. An interpretation of the data is made in respect to the environment, other entities and made under and assumption of a particular timeframe and location. Representation of multiple attacks from a single source, multiple attacks on a single destination, multiple attacks using the same methods can all be used to provide level 2 refinement.
Threat refinement is defined in Level 3 fusion. Estimations of danger, impending events, and other conditions are performed by an inference process to assess the threat of the adversary. Projection of what the attacker will do next is the goal of Level 3 fusion.

MSDF-based IDSs generally provide for Level 0-2 fusion processes. They refine the data, correlated it into tracks and make some inferences about whether these tracks are indeed attacks, where they come from, what their targets are and the identification of attack methods. Level 3 fusion, or the projection of the attacker’s likely next step, is the pinnacle of current intrusion detection. The higher levels of fusion are not well represented in the ID literature. According to Hall et al., Level 4 process controls the sensors, prioritizes tasks and provides for sensor health assessment. Level 5 addresses the human-in-the-loop and the contributions of the analyst in a cognitive process (Hall et al. 2004).

**Bayesian Inference**

In order to develop a system based on Bayesian inference, a complete list of all of the possible states of the system must be developed. These states must be mutually exclusive. Evidence that the system is in one of these states (it must be in one, because all of the possible states are included), is gathered and summed using Bayes theorem. The combination of evidence assumes that the evidence is statistically independent (Siaterlis et al. 2004). Traditional Bayesian inference requires the maintenance of all possible probabilities that a system is in a given state. Holsopple et al. identified that using Bayesian inference requires training using historical data. They noted that this historical data may not be representative of future attacker actions, and that the use of the Bayesian algorithm requires the high overhead of each individual probability analysis and summation (Holsopple et al. 2006).

**Dempster-Shafer (D-S) Theory of Evidence as a Fusion Engine**

D-S Theory of Evidence is used to solve complex problems where uncertainty is the norm and simple rule-based systems would be overwhelmed by the need to structure the uncertainty. The
The flexibility of D-S theory allows it to be used to interpret evidence based on degrees of belief. The Dempster rule allows for combining degrees of belief from independent evidence. Implementing D-S theory in a particular domain, according to Shafer, involves two related problems. The uncertainties of the problem have to be sorted into independent evidence and Dempster’s combination rule must be calculated (Shafer 1992).

Wu, et al. provides a relatively easy-to-understand description of D-S Theory. Their scenario uses an instrumented room (room with many microphones) and sensors that are tasked with discerning whether the current speaker is one person (A) or another (B). So, in deciding whether the person speaking is A or B, the possibilities exist that “it is either person A”, or “it is person B” or “it is either person A or B” or that “it is neither person a nor B and that it must be someone else.” This frame of discernment (FOD or $\Theta = A, B, [A,B], [\text{Somebody else}]$) is simply all of the possibilities that a sensor could produce (Wu et al. 2002).

Sensors are not perfect. They produce false positives. They miss reporting actual events (false negatives). To account for this, the observation that a sensor produces can be augmented with a confidence interval. The lower bound of the confidence interval is the Belief Function and the upper bound is the Plausibility Function of that sensor’s observation. The Belief Function is based on the sum of the evidence that supports the proposition. The Plausibility Function is the sum of all of the observations that do not rule out the proposition, or logically, one minus the sum of all observations that do rule out the proposition. Combining observations from multiple sensors follows Dempster’s mathematical rules (Farroukh et al. 2008; Tian et al. 2005b; Wu et al. 2002).
The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness

\[
[\text{Belief}(A), \text{Plausibility}(A)] \quad \text{Belief}(A) = \sum_{E_k \subseteq A} m_i(E_k) \quad \text{Plausibility}(A) = 1 - \sum_{E_k \cap A = \emptyset} m_i(E_k)
\]

Figure 4 - Confidence Interval, Belief and Plausibility Functions from Wu et al. (2002)

To clarify the Plausibility Function, Tian et al. identify that plausibility is the reliability of the proposition, otherwise defined as how much that the proposition is *not* in doubt (Tian et al. 2005a).

\[
(m_i \oplus m_j)(A) = \frac{\sum_{E_k \cap E_{k'} = A} m_i(E_k) m_j(E_{k'})}{1 - \sum_{E_k \cap E_{k'} = \emptyset} m_i(E_k) m_j(E_{k'})}
\]

Figure 5 - Dempster Rule combining Sensor Si and Sj's observations (m) from Wu et al. (2002)

Building of the FOD (Θ) for a particular IDS is based on what a human analyst may know about the current situation, attack methods and what the IDS will report under those attack types (Zhai et al. 2003a). This a priori knowledge must be built into the system so that it can compute the belief and plausibility functions to report with the sensor’s observation to be evaluated by the fusion engine.

The classic Dempster combination assumes that both sensors can be equally trusted. However, in complex systems, weights can be applied to the sensors’ outputs to account for differences in the reliability of the sensors, whether based on the sensor’s past performance or a human-entered heuristic value (Wu et al. 2002).

\[
(m_i \oplus m_j)(A) = \frac{\sum_{E_k \cap E_{k'} = A} [w_i m_i(E_k) \cdot w_j m_j(E_{k'})]}{1 - \sum_{E_k \cap E_{k'} = \emptyset} [w_i m_i(E_k) \cdot w_j m_j(E_{k'})]}
\]

Figure 6 - Weighted D-S Combination with Wi and Wj from (Wu et al. 2002)
D-S Theory in Intrusion Detection

Zhai et al. developed a model and prototype system to utilized D-S for intrusion detection. This early model surmised that D-S could be used to fuse alert data from multiple IDS’s, reduce false positives and generally create an early warning mechanism. The combination of evidence from multiple sensors was to provide better situational awareness. Basically, the concept is intuitive and follows logic similar to what a human analyst would do:

- List all possible results, and all possible propositions
- At the end of detection cycle – each IDS assigns basic probability assignment (bpa) to propositions in the cycle and build belief/plausibility propositions
- Using Dempster’s Rule – combine bpa’s into one
- Calculate the total bpa based on all cycles
- Reason, according to decision rule, estimate and make a decision (Zhai et al. 2003b)

However, the true complexity of the FOD (Θ) is not given sufficient attention in this work. While it may be more straightforward to describe the FOD as only one of four possibilities, this oversimplifies the true problem. In a real network defense situation, there are hundreds of possible vectors and thousands of possible methods to attack each of those vectors that should all be included in the FOD.

Siaterlis, Maglaris and Roris developed a distributed denial of service (DDoS) detection engine based on D-S. As opposed to other research, this work was specifically focused on being able to detect DDoS style attacks, especially those that flood communications links. From the perspective of a network provider, being able to detect DDoS attacks against its customers is an important customer service capability, even if that DDoS traffic does not completely saturate the service provider’s network, it may, in fact, effectively saturate the end-user customer’s link (Siaterlis et al. 2004; Siaterlis et al. 2002-2003).

In choosing D-S, the authors specifically rejected several other methods, including physical models such as the Kallman filter (normal network behavior model required), Neural Nets (training needed), Expert Systems (beliefs between true and false cannot be represented) and Fuzzy Set Theory. D-S was chosen because of its ability to perform without a known “good” network state or “normal”
state and because of its ability to combine data from multiple heterogeneous sources (Siaterlis et al. 2004; Siaterlis et al. 2002-2003).

**D-S Theory in Practice in Intrusion Detection**

To understand the application of D-S Theory in the actual practice of IDS sensor fusion, it is helpful to describe the process using examples. First, the FOD (Θ) must be determined. This is the list of all of the possibilities that the sensor output could possibly report and all of the propositions must be determined. Next, alerts are correlated into tracks. The IDS system works in cycles, each cycle calculating the probability assessments of each proposition, then fusing all of the sensor data for that cycle related to that proposition. Finally, there is a fusion of the probabilities from multiple cycles together.

From the correlated tracks as represented in Table 1, below, the bpa (basic probably assignment) is calculated for each alert. To get from Table 2 to Table 3, below, D-S theory using the Dempster Combination Rule is applied, combining the sensor data from all of the sensors (E-Trust and Snort in this example).
A final assessment is performed by the human analyst. In this very brief example, the Port-Scan, Scan UPNP and UDP flood propositions are much stronger than the others. This provides for situational awareness that those three methods are the ones in use by the attackers (Tian et al. 2005b).
The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness

**Evaluation of D-S Theory Based Systems**

Risk assessment methods using D-S Theory presented in (Mu et al. 2008) identified a reduction of false positive results. The IDSDFM fusion model originally presented in (Tian et al. 2005a) and then later refined in (Tian et al. 2005b) and (Tian et al. 2005c) was reported to have reduced both the false positive and false negative rates of alerts, but no evaluation method was described.

The distributed IDS system presented in (Farroukh et al. 2008) was evaluated subjectively by the researchers. They claim that the distributed system was comparable to a centralized system, but did not suffer from the possibility of a single point of failure.

Yu, et al. tested several fusion methods to include Voting, Bayesian and D-S Theory algorithms and compared them to a fusion system using a combination of system. They tested against the KDDCUP’99 full dataset for all systems. They found that their D-S Theory based algorithm had a 99.6768% accuracy of correct classification, and misclassified only about 1000 data points out of almost five million records (Yu et al. 2008).

**Neural Networks and SVMs**

Xiao et al. proposed the use of Neural Networks for data fusion in ID. They proposed that the IDS, antivirus, firewall and other security tools could be used as sensors in a data fusion application. However, other than identifying that a neural network could be used in this fashion and claiming to have developed such a system, no discussion of their actual implementation was provided (Xiao et al. 2006).

Liang et al. also proposed the use of neural networks, however, to analyze network data from the 1998 DARPA data set in order to extract features from the data and compare the fitness of their neural network-based approach to genetic algorithms. An overall security risk value is calculated, providing an overall assessment of the network security situation on a scale of [0,1] (Liang et al. 2007).
The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness

However, other than the assessed value, no information about the underlying reasons for the increase or decrease in the assessment is provided.

Wang et al. provided more description in their use of neural networks. In the combination of data from a Snort IDS and from a Cisco Netflow collector, they identified that some feature reduction was required as to not overload the neural network with unnecessary data fields from each of the IDSs. They proposed the use of a multi-layer feed-forward neural network (MLF-NN). Using the DARPA 1999 data set, they were able to train and evaluate their algorithm. Their output provided a rapid fusion and assessment of the intrusion behavior identified in the traffic. However, their results showed a higher than desired false positive rate, implying that more sensors are required to eliminate the false alerts (Wang et al. 2007). The same research team also developed a prototype of a multiclass support vector machine-based (SVM) fusion engine and reported their findings in (Liu et al. 2007a; Liu et al. 2007b; Liu et al. 2007c). Their assessment was that the SVM approach was just as reliable, but much faster, indicating that SVMs have promise in real-time IDS applications.

Thomas and Balakrishnan developed a data fusion system based on neural networks that combined the results of three different IDS systems that monitored the DARPA 1999 data set. The individual capabilities of PHAD, ALAD and Snort IDS systems had a success rate of 28%, 32% and 51% respectively. However by fusing the alert data from all of the systems together, they were able to increase to the total successful detection rate to 68% and significantly increase the detection rate of certain attack types even though those attacks were not detected well by each individual IDS (Thomas et al. 2008a; Thomas et al. 2008b).

**Summary of Data Fusion Techniques in Intrusion Detection**

The contributions of data fusion algorithms in the application domain of Intrusion Detection can be summarized by following their history. Early IDS fusion is formed with simple alert correlation based on source address, destination address and attack type. More recent applications of data mining
techniques such as SVMs and Neural Networks appear to be focused on developing real-time similarity correlations. These techniques can be applied to the JDL Model to understand where their contributions will be most effective. For instance, when combining data from different IDS sensors, a technique that uses high-speed correlation of low-level inferences may provide a fused output that reduces false positives by only reporting those alerts that have been confirmed by multiple IDS mechanisms. This type of mechanism might be further aided if there were some standardization of the output of the individual IDS sensors.

Use of D-S Evidence Theory might be more appropriate for a higher level of inference. For instance, combining dissimilar data sources that all point to different types of evidence of an intrusion might be possible using D-S, where data mining algorithms may not be possible. Evidence of an intrusion may be reported by an IDS, or by host-based sensors, central authentication services, anti-virus systems and other very dissimilar sources that all may show possibilities of the effects of malware or blended threat activity.

Level 3 fusion implies the ability to project into the future. Current IDS systems and the proposed fusion algorithms do not provide the capability for future projection. Level 4 of the fusion process indicates a feedback mechanism in terms of tuning of the sensors. To date, no IDS function addresses this level of model. JDL Level 5 implies cognitive refinements and a handoff to the human-computer interaction layer. This level is discussed in the next section.
Section 3 – Situational Awareness in Cyber Security

To understand the impact of the contributions identified in the previous section, it is important to understand the goals of the computer network security analyst. In a very broad sense, the objective is for the analyst to provide network security. However, “security” is vague and not well defined. One could certainly digress into identifying specific tasks that the analyst performs in order to provide security, but what is “security”?

According to Giovanni Manunta, security, in a general sense, has different meanings to different people in different contexts. In terms of a formal definition, security is “a function of the presence of an interaction of Asset (A), Protector (P) and Threat (T) in a given Situation (Si)”. The absence of one of these core elements (A, P and T) renders the concept of security meaningless. More importantly, the context of the situation frames the security environment giving it scope (Manunta 1999).

Matt Bishop takes a different, more practical approach when defining computer security. In terms of the information system itself, a secure system can be defined as a state of the system in which only authorized actions can be executed by authorized users. If the system allows for a user to execute unauthorized actions, it is deemed non-secure. This all-or-nothing approach is limited in its applicability. However, one can combine this with the mechanisms of enforcement of a defined security policy and measure the effectiveness of the mechanism to meet the objectives of the requirements that are documented in the policy as well as document the policy’s ability to provide security (Bishop 2003).

The CIA Triad identifies the core components of information security. Confidentiality (C) represents the system’s ability to limit access to the information contained only to authorized users. Integrity (I) represents the system’s ability to provide for the ability to prevent unauthorized modification of the information. Availability (A) represents the system’s ability to operate without disruption (Tipton et al. 2007). A system’s security can be measured with all three of these attributes.
Successful attacks and exploitations of vulnerabilities can be categorized as violations of one of these three attributes.

The role of the analyst can therefore be defined as the ability to take action to maintain the Confidentiality, Integrity and Availability of the information system. In terms of Manunta’s equation, the information system is the Asset (A) and is a combination of its hardware, software, data, network connections, administrators, users, policies and procedures. The analyst plays the role of Protector (P) as an agent of the information system owner and must be aware of the past, present and possible future threats. Hackers, malware, insiders, unwitting end users, system failure and natural disasters, in total, represent the Threat (T). The Situation (Si) is the combination of all of these elements at a specific moment in time. The CIA Triad provides a framework for the analysis of the impact of a specific threat and understanding the vector by which the threat will affect the information system. The development and maintenance of situational awareness about the Asset and Threat to the asset is crucial to the successful analyst.

**What is Situational Awareness?**

According the Endsley, situational awareness is more than simple awareness of multiple sources of data; it requires the synthesis of that data into an understanding of it in the context of the “operator’s” goals. In complex systems, the act of acquiring and maintaining good situational awareness takes a major part of the operator’s time. The classification of that information helps the operator to understand the situation and select appropriate actions to take. Depending on the application domain and individual cognitive characteristics of the operator, the development and maintenance of awareness can be relatively quick or can take significant effort. This awareness can take the form of pattern matching the current situation to known previous examples by matching features in similarity (Endsley 1995).
Endsley’s model of SA contains three levels. Level 1 is the understanding specific elements of the current situation. Level 2 is the marked by an understanding or comprehension of the meaning of elements of evidence and their relationship to the overall current situation. Level 3 is a predictive element of understanding what is likely to happen based on current system state and the consequences of actions that may be taken by adversaries and operators alike (Endsley 1995). It is the combination of these three levels that define situational awareness. In the specific domain of network security, Level 1 presumes an understanding of the reporting capabilities of sensors in use. Level 2 combines this low-level data with knowledge of the network architecture being defended and the attack methods in use by the adversary. Level 3 projects the impact of future actions from both the defensive and offensive playbooks.

![Levels of Situation Awareness adapted from Endsley (1995)](image)

Luff, Heath and Svensson provided a very recent assessment of situational awareness in the operations rooms of the London Underground subway system. This high level cognitive task analysis details of situational awareness issues of the operators who monitor the station’s closed circuit television cameras (CCTV) (Luff et al. 2008). The parallels between this work and the work of network security professionals is clear. Instead of a station with hallways, platforms, escalators and patrons, the network security professional has a network with switches, routers, computers, servers and users.

Instead of panhandlers, pickpockets, ticket touts (scalpers) and other criminals, the network
The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness

professional has hackers of varying capabilities and interests. Instead of CCTV cameras, IDSs and other systems provide a window into the activity on a network.

Similar to the operations center staff, the network security professional has multiple duties to attend to. The subway operations center staff does not continually monitor the cameras during the course of the day, but instead uses them to provide awareness of problems. The operations center staff also does not monitor every camera, as the computer network professional does not monitor every IDS or host log on a constant basis. Reported was a ratio over a hundred cameras to as few as 8-10 monitors. Some monitors were set to display certain areas of the station, while others rotated through the remaining cameras. Recording systems provide previous histories of captured images to review the scene after the fact (Luff et al. 2008).

Awareness of some situations could be aided by automated systems. Anomaly detection – detection motion in an unauthorized area, for example, or detecting a passenger going against the flow of traffic in another instance, could be provided with some kind of software assistance (Luff et al. 2008). This type of activity is similar to anomaly detection in ID. Pattern matching is also a similar method with the detection of specific patterns of activity that are known to be related to illegal or other unwanted behavior.

As one would expect, the criminal element in the stations is very aware of the location and fields of view of the camera systems. They will often hide outside of the visible range of the camera, even underneath it, to prevent the station supervisor from seeing their exact actions. However, the more aware supervisors notice the telltale signs of their presence by the actions of others. For instance, a street performer or a beggar could be observed in a camera blind spot by the actions of patrons turning their heads, reaching over and tossing coins in his collection bin. A trained operator will notice and act on this secondary evidence even if it is off camera (Luff et al. 2008).
This awareness developed and maintained by the Underground Supervisors has similarities to that of network security professionals. The network security professional may not be able to identify every portion of an attack, nor does he need to be able to in order to perform his job. A host on the network opening up a connection to an IRC server might be evidence enough of an intrusion, or different stages of an attack can be recognized by and IDS and alerts can be sent. Other signs and signals of intrusion and malware can be reports of computer slowness or inability to perform certain tasks. While the network security professional does not desire system downtime, there are certain kinds of activity on a host computer that indicate the necessity for the machine to be removed from the network and reformatted, even without a full audit trail of attack and compromise.

The process of situational awareness between these two types of workers is very similar. Both must be aware of the terrain of their environments. For the network security professional, it is the network connections, locations of firewalls, switches, critical hosts and other network devices. This parallels the subway supervisor’s knowledge of the station layout, problem areas, escalators, platforms and physical topology. Both types of workers need to be aware of the capabilities and methods of their adversaries. While the subway supervisor does not need to know how to pick someone’s pocket, or even be good at it, he must know how to identify the telltale signs of someone who is engaging in this activity. Similarly, a network security professional does not necessarily need to know how to compromise a host machine with a variety of methods, but must know what signs and signals the IDS, host machine and other sensors will report when the adversary attempts to do the same.

A final parallel is in the multi-tasking requirement of these two types of workers. In many cases, neither one is dedicated to the security task 100% of their work time. More often than not, the station supervisor’s primary duties may be in the management of the efficient operation of the station including scheduling of arrivals/departures, making passenger announcements, and handling all sorts of staff and passenger queries (Luff et al. 2008). The IT professional also has many tasks including the efficient
The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness

operation of the computer and network functions of the organization to include internetworking devices, servers, host computers, software installation, and even first-line application level support. While there are professionals in both realms who are dedicated to the tasks of security, the trend is that these people are tasked with multiple, simultaneous and sometimes competing job duties.

To that end, automated systems can help both kinds of professionals to meet their security responsibilities. Systems that monitor the area of interest and provide alerts can be helpful. However, these alerts must be timely, efficient and without significant false alarms. The history of IDS has proven that high alarm rates will annoy the user and may ultimately result in the system being turned off, especially if the alarms have a high false positive rate.

Undoubtedly, there are many other application domains in which situational awareness in the domain has parallels to the domain of network security. Firefighters deal with situational awareness in austere and threatening conditions, yet still develop effective SA through experience and a rapid cost-benefit risk analysis decision making process. In larger emergency situations, the Incident Command System (ICS) is used for distributed cognition, decision making and communications. Military commanders develop action plans for the battlefield using knowledge of the enemy, terrain, situation and capabilities of their own forces. The Intelligence Preparation of the Battlefield (IPB) process is a documented and rigorous process to make sure that commanders consider all aspects prior to taking action. The processes that these other organizations follow can and should be explored to find common issues, concerns and solutions.

**Cognitive Load Issues in System Design**

The concept that situational awareness as a state of knowledge is adopted, in agreement with Endsley’s definition (Endsley 1995). To that point, it is then presumed that with the current status of the cognitive capabilities of computer systems and algorithms, that knowledge, and therefore awareness, is something that only a person can create or possess. An “awareness machine” would be a monumental
accomplishment. The role of computer systems and algorithms should be in the processing and representation of data, information and knowledge in the support of human cognitive activities such as the development of awareness.

What would be a more reasonable effort would be to develop systems to help reduce the cognitive load of the analyst. Sweller et al. reported on the issues of cognitive load in instructional design, but their recommendations can and should be applied to the development of any system that supports learning. Primarily, the designer must understand that working memory is limited. Long term memory is unlimited, and has the ability to represent complex information through the use of schemas, or skeleton structures that represent the complex information. The access of long term memory is through working memory and is therefore subject to load conditions. Certain types of activities are detrimental to the learning process and increase cognitive load. The split-attention effect is problematic, especially if a single sense is required to focus on separate or multiple objects. Sweller reports that having to focus on a block of text and also a separate diagram increases cognitive load and decreases learning performance. The modality effect is a positive and beneficial use of multiple senses (sight and sound, for instance) to reinforce learning. The repetition effect has a negative impact on learning, if the repetition with multiple, potentially conflicting information can overload and confuse the learner (Sweller et al. 1998). For the network analyst, the split attention effect is common, with high cognitive loads from the need to monitor multiple systems simultaneously, and perform unrelated work. Systems should be designed to focus the attention of the analyst, reduce the split-attention effect, reduce conflicting repetition and take advantage of the modality effect to increase learning through reduced cognitive load.

**High Level Fusion**

Higher level fusion can be described in either the JDL model or the Endsley SA model. Yang, et al. compared the two models and their applications in cyber security. They compare Endsley’s Perception
The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness

to the JDL Model’s levels 0 and 1 in the Cyber Security application domain with activities that detect malicious activity. For Yang, this relates to the activities of intrusion detection and host activity logging and the fusion of these items via low level data association. The JDL Level 2 fusion process “Situation Assessment” and Endsley’s “Comprehension” are manifested in the use of complex modeling to gain an understanding of the attack via the correlation of alerts. The process of threat projection can be developed in JDL Level 3 “Threat Refinement” and Endsley’s “Anticipation”. Yang identifies two possible cyber security activities here: impact (damage) assessment and threat projection, and focuses on the latter (Yang et al. 2009).

![Diagram](image_url)

**Figure 9 - Mapping of IDS Fusion tasks between JDL Model and Endsley SA Model. From Yang et al. (2009)**

Yang et al. developed a system for both the upper levels of this model. The first system, INFERD provided automated alert correlation using a priori knowledge from subject matter experts on both the attack methods and knowledge of the specific vulnerabilities of the defended system. Fusion of these two data sets provided an assessment of the likely outcomes on the given attack on the given network, reducing the time for assessment by limiting the problem set to the specific problem space (Yang et al. 2009).

The second system was the corresponding threat projection module, TANDI, which uses the attack tracks provided by INFERD and overlays an a priori model of the defended network to identify
specific nodes (computers, accounts, databases, etc.) that are subject to compromise. Using knowledge of previous successful and unsuccessful attacks, future attacks can be probabilistically predicted. TANDI combines assessments of possible outcomes and threat levels based on a variety of factors including the intent and skill level of the hacker, value of the potential target and the utility of the vulnerability being used. These provide potentially conflicting predictions and TANDI combines them using a D-S Theory-based combination. A compromised host would receive a value of 1, while evidence of potential compromise can be given weighted scores [0,1] and combined using the Dempster Rule. TANDI was tested with some un-optimized weights in its algorithm, yet still provided reasonable assessments (Yang et al. 2009).

INFERD and TANDI were evaluated with extremely interesting and promising results. However, with these promising results comes a caveat – the threat assessment and prediction algorithms that are based on a priori information are susceptible to the limitations of the knowledge of the experts who provided that information (Yang et al. 2009). Flaws in the basic underlying knowledge will result in flaws in the output. In particular, it appears that while these are excellent systems for the detection of known methods of attack on a known infrastructure, nothing will replace the analyst-in-the-loop having an understanding of the defended network and human instinct to identify the abnormal behavior in a complex system. Systems like INFERD and TANDI are certain to be extremely beneficial in the development of awareness for the human analyst.

**CTA of Computer Network Security Analysts**

According to D’Amico, a Cognitive Task Analysis (CTA) is the study of the cognitive processes of individual and teams of employees to understand their cognitive processes, communication and activities to perform their duties (D’Amico et al. 2007). Brewer identifies a CTA as the field work used to identify specific cognitive processes, skills and knowledge that individuals and teams of workers use to successfully complete their tasks. The CTA is based on previously successful processes of knowledge.
elicitation, HCI, CSCW, UE and naturalistic decision making (Brewer 2005). The objective of the analysis is to inform future system design by the observation of expert workers. While there is considerable variation in the specific methods to employ, semi-structured interviews, direct observation and hypothetical scenario development and discussion are common methods of knowledge elicitation (Biros et al. 2001; D'Amico et al. 2005; D'Amico et al. 2007; Killcrece et al. 2003).

Biros and Eppich conducted a CTA of rapid intrusion detection analysts in the U.S. Air Force. Their CTA identified five cognitive capacities required. They are “recognition of nonlocal Internet Protocol (IP) addresses, identification of source IP addresses, development of a mental image of normalcy, creation and maintenance of analyst situational awareness, and facilitation of knowledge sharing” (Biros et al. 2001). While the first two tasks appear to be rudimentary, the impacts of the last three tasks are comprehensive. In 2001, the simple identification and quick association of IP addresses to being either inside or outside of the defended network seems trivial, but important to building awareness of what belongs to the organization and what is on the outside. The overall development of a mental image of what is normal on a particular network is a common theme within the literature to follow. This supports the development of situational awareness. Knowledge sharing, especially among peers and co-workers, has obvious collaborative work implications.

Killcrece, et al. surveyed the teams and reported the activities that each performed, categorizing them into 1) Reactive, 2) Proactive and 3) Quality Management. The quality management category included tasks that are not directly related specific incidents but are related to the security function, such as training, new product evaluation and disaster recovery planning. Proactive tasks include researching new malware, tuning sensors to detect anticipated attacks, and other activities that anticipate the next possible exploit vector. However the majority of the time of an analyst is spent on reactive tasks, including incident response, threat identification, monitoring of IDS, monitoring system logs and post-incident response documentation (Killcrece et al. 2003).
D’Amico et al. reported the results of a cognitive task analysis of computer network security professionals from various agencies within the U.S. Department of Defense. This CTA provides a significant number of insights into the current work practices of a large scale organization, and one that takes network security seriously. They reported an understanding of the varying job duties, common language (or lack thereof) and a number of other dissimilarities between the varying organizations. However, the process of the work appears to be similar from a high level. In short, automated processes transformed “raw data” into “interesting activities”. From those “interesting activities” an analyst triages the alert data and what remains is the “suspicious activity”. From that “suspicious activity”, “events” are reported, removing those that are outside of scope for the particular agency. Multiple “events” can be aggregated into a single “incident”, where a formal, documented part of the process occurs (D’Amico et al. 2007).

![Data Reduction Processes through the Work of an Analyst](image)

Figure 10 - Data Reduction Processes through the Work of an Analyst - from D’Amico et al. (2008)

D’Amico comes to the same conclusion as Yang in terms of the comparison of Endsley’s SA model and the JDL Model in the work domain of intrusion detection and computer security. JDL Level 1 is compared to Endsley’s Perception – where the analyst performs triage on the alert data and begins to form an opinion on the attacker’s actions. JDL Level 2 is compared with Endsley’s Comprehension – where the analyst combines the suspicious alert data with their own experience and additional data sources to determine if the suspicious activity is an incident or not. This level of correlation is used to
identify patterns in the suspicious activity. Endsley’s Projection corresponds to JDL Level 3 by correlation of activities with other analysts and proactive analysis identifying possible new actions that the attacker might take (D’Amico et al. 2007).

**Visualization in Support of SA**

Network security visualization has the potential to address the human-computer interaction concerns for the analyst as well as represent a variety of data to support the analyst’s mission and tasks. Pin Ren suggested that the VizSec community needs to address several tasks. Those are 1) the generation of high-level visualizations, 2) evaluations these visualizations for their effectiveness and 3) integrating these visualizations into existing security tools so that they can be used by analysts in the field who can provide feedback on their usefulness. He identifies that current IDS technologies quickly overload network administrators, especially due to their high false positive rates, and that visualization technologies hold promise in some, but not all aspects of computer security. While there have been some successes in visualization support of security, there are no standards for evaluation of visualization tools and limited integration of successful visualizations into mainstream security products (Ren 2006).

Goodall suggests that the effort of security visualization is to provide the human analyst with the data in ways that support human cognition and perception to perform the analysis, anomaly detection and correlation identification. While there are systems that are being designed to automate the process of ID, these systems undervalue the input and capabilities of the human in terms of exception handling and the identification of novel patterns. Automated systems do have benefit in increased scalability and the processing of large amounts of data, but the human analytic skill is essential. The visualization process should therefore be in the support of the representation of abstract data to support human cognition (Goodall 2007).
**Network Topology-Based Displays**

A number of visualizations support the analyst by representing data in a network topology-centric view. These visualizations support the grouping and organizing network flow data, traffic data, application-level data and other sources of data overlaid on to some type of logical representation of the network.

Because the Internet is so large, representing source addresses by autonomous systems (AS) has been suggested. Oberheide and colleagues represented raw Border Gateway Protocol (BGP) data to extract all of the top level systems on the network. BGP is the routing protocol used to share routing information between high level networks on the Internet – between large ISPs, large organizations and even countries. These authors represented the ASs via AS Number and contained information such as included address spaces. The visualizations suggested can easily identify BGP mis-configurations if one AS inappropriately announces routing information to the rest of the network (Oberheide et al. 2006). Shearer, Ma and Kohlenberg also explored representing BGP data in a visual format. They identified visualizations that can provide insight into inappropriate or inefficient BGP changes. Route flapping, intentional/unintentional inappropriate route announcements and inefficient route announcements can be easily identified in their BGPeep visualization (Shearer et al. 2008). While these representations of a very important part of Internet connectivity provide valuable information to top level network providers, the utility is limited to these providers and is not necessarily germane to the majority of network security analysts.

![Figure 11 - Representation of BGP data in BGPeep from (Shearer et al. 2008)](image-url)
Mansmann and Vinnik represented the network flow data between their University and the 20,000 different AS’s to support the system analyst’s understanding of the sources and destinations of potentially anomalous traffic and then later tie the AS numbers to physical geographies to support human understanding of the source and destination locations (Mansmann et al. 2006).

Ball, Fink and North explored using polar plots instead of Cartesian plots to represent network topology-focused data. They identified that a polar plot can be useful, but noted that three major issues with the data at the center of the polar plot – distortion, occlusion and smaller area. They suggest that root polar plots address the majority of these concerns, taking the square root of the value to slow the growth of the function as it approaches the edge. Using root polar plots, they also developed a method of displaying varying trust levels of local and non-local IP addresses in a diagram of concentric rings. The more trusted the IP address is, the closer it is to the center of the diagram. The further the point is from the center, the less trusted it is (Ball et al. 2004; Fink et al. 2005).

![Root Polar Plot](image1)

Figure 12 - Examples of Root Polar Plot (left) and IP addresses represented on this style of plot (right) from (Fink et al. 2005)

Li, Liu and Kesidis from Penn State developed a toolkit for visualizing network testbeds such as DETER and Emulab. Their package provides a TCL script generator, network topology builder and a number of visualization tools. The underlying testbed tools provide simulation capabilities for DDoS, worm propagation and traffic generation capabilities within their simulated environments. The tools that the Penn State team developed provided for the automation of large scale network implementation.
and generated visualizations of malware attack propagation on that infrastructure. The system provides visual representations of bandwidth saturation, overloaded routers, CPU load and memory utilization and other symptoms of the malware infestation in a user-modifiable representation using interactive views (Li et al. 2006). While the representation of some kinds of malware systems will be apparent on default views of this kind of system, the exploration of the views will be highly dependent on the analyst’s knowledge of what to look for and where. This representation of network data simulation represents interesting possibilities for analyst training, exploration of behavioral cues that might be present in novel malware infestations and a possible development of real-time applications on live network data.

Mansmann and Vinnik proposed a method for visualizing anomaly based IDS. They note that machine learning algorithms fail to identify anomalous behavior. The high possible false positive rate requires that the data be represented in a visual form for the analyst to be able to explore the data and use the unique human qualities of intuition, perception, and additional domain-specific knowledge. They propose a display that takes both logical hierarchy and physical geography into account. Pixel level displays were also explored. If each pixel were assigned to a single IP address, for instance, about 9 million IP addresses, each one represented by a single pixel on an extremely large and high resolution display screen or wall. Pixel color could differentiate metrics of interest such as traffic volume. Combining the logical, physical and pixel-level concepts results in a combined display with a variety of view options and high levels of detail. The authors propose some hypothetical uses of the system by identifying malware behavior through specific port or protocol utilization, but only provide that their display has the potential of displaying that information if the analyst knows what to look for (Mansmann et al. 2006). While this kind of display can graphically provide information to the analyst, the issue of cognitive load remains unresolved.
Pike, Scherrer and Zabriskie speculated that there may be geopolitical motivations behind certain cyber attacks. They represented network packet flow data from organizational and regional perspectives. They divided the flow data into smaller levels of organization called “cliques”, which could represent regions (cities, countries, etc.), organizations or even single IP addresses. Alternatively, behavior could also be used to identify a clique. This high-level, exploratory based representation is manifested in their “dashboard”, based on a Von Mises circle (Pike et al. 2008). High level representations of network data that is tied to a geopolitical map certainly has its benefits in developing awareness for the analyst.
**Network Traffic-Based Representations**

Other researchers suggest that the representation of the flow of the traffic is more important than representing the data in terms of the network hierarchy. Fink and colleagues presented a utilization of polar root plots to represent the flows of data on pixel-based representations. Each pixel represents a port rather than a host. The source to destination port mapping provides an end-to-end representation of the activity (Fink et al. 2004).

![Diagram of Local and Remote Host-Application Resource Visualization](image)

*Figure 15 - Representation of host to port to remote port to remote host of network traffic from (Fink et al. 2004)*

Blue et al. developed some rudimentary visualizations to represent single host relationships based on network data. Their visualization represents hosts as circles and connections between them as edges in a simple network diagram. Edge and node coloring, and user-defined grouping complete their visualization to indicate traffic types, flows and other similar characteristics. However, the analysis of their design was limited to a single-expert heuristic evaluation. They also explored a treemap based representation, but because the underlying data was anonymized, limiting the possible functional aspects of the representation due to lack of ability to group the data (Blue et al. 2008).
Data can be collected through products like Cisco Netflow and displaying them with a variety of visual representations. Taylor, Brooks and McHugh developed 3D impulse graphs to represent data volumes by port over time. Normal, expected traffic patterns can be compared to abnormal patterns based on regularity or irregularity of the plots. For instance, a mail server might show very regular port 110 (POP3) and 25 (SMTP) traffic patterns. However, a system that has been compromised and misappropriated to be a mail relay might show the same pattern, even though it is not expected to. Also, a compromised machine might show regular patterns of activity on unexpected port numbers. (Taylor et al. 2008) Fisher, et al. also used Netflow, but their visualizations were more appropriate for large-scale networks. For instance, large-scale botnet attacks show interesting patterns on an IP-to-IP connection map. Connections using specific services such as time servers, DNS or IRC traffic may provide interesting information about systems using those services. However, more interesting are their representations of potential compromise of a single host based on thresholds of number of connections or activity (Fischer et al. 2008).
IDS-based and Track-Based Displays

Using the tracks developed from INFERD, Mathew et al. developed a visualization to display multi-stage attacks on the network. Their representation is based on attack graphs and displaying the current information about a potential multi-stage attack as it develops. It matches this growing threat to a template of a known multi-stage attack method in what is described as a “guidance template” (Mathew et al. 2006). This type of representation is highly dependent on the track information provided by INFERD and also upon the a priori knowledge about the attack methods.

Saunders and colleagues represented NetFlow and IDS data simultaneously on a single display. They identified that to be intuitive, a visualization system must not require focused attention, and that the glyphs shown must change only based on minimal features. They chose color to represent features in the data and perceptual grouping based on location to identify hosts that had common features. Perceptual size indicates a quantity of data exchanged and a simple X identifies port activity that is suspect (Saunders et al. 2005).

A more interesting and potentially useful visualization is presented by Foresti and Agutter. VisAlert combines the traditional “3 Ws” into a single display. When, Where and What are all
The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness

represented in a circular visualization. The “Where” layer is a network topology in the center of the visualization that identifies network hosts. The “What” layer identifies the type of IDS alert that is being represented. These events fall on the outside of the circle and connect the hosts by lines or edges. Then “When” is displayed with concentric circles of “What” layers, growing older as they move out from the center of the circle. Thickness of the beams that connect a host to an alert indicate the persistence of a particular problem (Foresti et al. 2007).

**Figure 18 - Representing the Three Ws from (Foresti et al. 2007)**

**Current State SA in Cyber Security**

Visualization of network data and IDS Alerts has the potential to make significant impacts on the daily work of a security analyst. However, the current status of network security visualization, as represented above, does not directly account for the cognitive issues that confront the analyst. More attention in visualization development needs to be paid to reducing cognitive load through reducing or
eliminating the split attention effect, increasing the modality effect or decreasing conflicting repetition. Additionally, the representation of data in the Endsley SA or JDL (levels 1-3) models can support an understanding of the part of the situational awareness problem that a particular visualization is intended to address. For instance, an excellent representation of alert data might be very helpful in Level 1/Perception. Even if it does not represent a development of higher levels of awareness, presentation to the analyst for human-based fusion is still extremely important.

Cognitive Task Analysis appears to be a good method for identifying the needs of the security analyst, but more of this work needs to be done. Data fusion and the development of awareness appear to be in the realm of the cognitive capabilities of the human analyst. What is obviously different between D’Amico’s CTA and Yang’s proposed system is the location of where the correlation processes take place. D’Amico describes the correlations taking place in the mind of the human analyst and between multiple analysts in a collaborative work environment. Yang describes and automated system that performs the correlations at two different levels and reports the results to the analyst for action. While the promise of systems like INFERD and TANDI represent intriguing possibilities, the bulk of the work in the field today is performed by people. Those people need supporting technologies in terms of data fusion and visualization to address their needs.
Section 4 – Future Research Plans

Future research in this application domain of situational awareness will certainly be addressed from many different perspectives. Computer scientists will continue to develop algorithms for fusing data. They will also develop visualization technologies to represent network data and IDS alert data in an attempt to assist the security analyst in understanding what is going on in the network. Additionally, engineers from many different application domains will use the JDL data fusion model and others to apply the concepts of data fusion for their own uses. Their developments, both theoretical and algorithmic should be analyzed and adopted to the domain of intrusion detection. More importantly, the development of algorithms for data fusion in ID should be organized around a shared model, similar to Endsley’s SA model or JDL levels 1-3. Understanding the potential benefits of the algorithm in terms of the benefits that the algorithm, source data and output will bring to awareness, comprehension or projection into the future will frame the technical solution’s potential for the development of awareness.

There is certainly plenty of room in each of these levels of awareness development for additional contributions.

Cognitive scientists will continue to develop the concepts of situational awareness, and apply them to many other application domains, such as firefighting, police, and the military. From a very high level, they will learn how to support the development of situational awareness for practitioners in those domains. The needs of those practitioners will be documented in rich, ethnographic-like studies. Researchers will use of semi-structured interviews, surveys and direct observation in the form of cognitive task analysis and document those findings for many different combinations of situations.

While military and federal government security analysts represent one source of data, CTA should be performed for information security professionals from industry, healthcare, education, small business, large scale enterprises, public sector organizations and national governments from a variety of cultural, economic and social backgrounds. It is only through this rich exploration of the daily work tasks that a
whole picture of how the job is done will be developed. Through this understanding, tool development will shift from random, haphazard contributions into a more organized effort that will truly support human cognition of situational awareness in this domain.

The new researcher considering studying intrusion detection, data fusion and situational awareness has a myriad of choices in which to develop expertise and contribute to the body of knowledge for this subject area. It is in the domain of the computer scientists to develop the algorithms and visualizations. The role of the cognitive scientists is to understand the processes of human learning and comprehension. However, the role of the interdisciplinary scientist is much more comprehensive. We must first understand the problem and develop the rich understanding of the current practices of computer and network security analysts. At the same time, we must organize the technical contributions of computer scientists using a well-accepted reference model and select those contributions that will address the development and maintenance of situational awareness. As successes in this process are implemented in the real-world, the interdisciplinary scientist can then direct the future contributions of their disciplinary counterparts to build on that success. This iterative process will take decades to hone and will be marked by lifetimes of careers. However, in the rapid pace of technological development, better tools are needed today. Development must be simultaneous including new technologies, model building and, most importantly, understanding of needs of practitioners.

**Shared Model for Data Fusion in ID**

A model needs to be developed and adopted on which to measure and represent the technical contributions. The lack of a model of how and where data fusion can be applied to create better situational awareness in cyber security means that researchers will continue to make a variety of uncoordinated, albeit, interesting advances.
For instance,

- Fusing output from dissimilar IDSs that are observing the same data stream to provide complementary detection capabilities.
- Fusing output from similar IDSs with different frames of reference can provide a network-centric view of alerts across the enterprise.
- Fusing output from IDS and host health systems can provide a system vulnerability assessment.
- Fusing output from an IDS and an antivirus system can provide information about the use of blended threats.
- Fusing output from the IDS and a centralized authentication authority can provide information about the use of compromised accounts.

**Proposition 1 – IDS and Situational Awareness**

IDSs are necessary, but not sufficient, for Situational Awareness. Awareness cannot be created in an automated system. Awareness requires the sentience of a human being. Systems can be created to aid in the development and maintenance of situational awareness.

**Proposition 2 – Data Fusion and Situational Awareness**

Situational Awareness requires the fusion of information and knowledge about the defensive posture of the network landscape and the offensive actions and capabilities of intruders. High level fusion requires both perspectives to understand the capabilities of the attacker and the likelihood of success on the particular network. Fusing output from different types of network security sensors that are observing different parts of the network security landscape will provide richer awareness.

**Proposition 3 – Data Fusion Algorithms**

Specific data fusion algorithms may be useful for in specific data fusion applications to aid in the development and maintenance of awareness, even if they do not bring about awareness on their own. Understanding which algorithms bring about specific outcomes in terms of awareness is important. Algorithms must be suitable for the inputs and will produce specific results. Understanding where those inputs come from and what the intended results are requires the reference model.
Cognitive Task Analysis

As indicated above, more CTA work needs to be performed and this work will be industry specific. While D’Amico performed a CTA with military and government analysts, the work of analysts in this environment is highly segmented and focused on the specific tasks of network security. Large enterprises may also have the resources and personnel to dedicate to the specific task of computer and network security, but may not have an attack/defense posture the same as military and government organizations. In contrast, highly regulated industries, like healthcare and finance are more heavily burdened with reporting and compliance, rather than the active work of securing their systems. In small and medium sized businesses, in particular, network security tasks receive little attention. Institutions of higher education provide a hybrid model, especially where budgetary authority for information technology expenditures is fragmented throughout individual colleges, schools, departments and work units at a large institution.

A CTA of network security analysts in a large educational institution could provide valuable insight as to the daily duties and awareness needs of IT professionals in a variety of organization types and sizes. The split attention effect is compounded not just on attention for multiple network security monitoring systems, but there is also a need to balance the security function with the daily administration of computers, systems and networks and end-user support. Because this author is a member of the community of computer professionals, gaining access and developing a common rapport could be more straight-forward than if this work was performed by an outsider. This author would start with the development of CTA at lower levels of the organization (department and college-level IT support) and work the way up through the college-level network security officers to the centralized computer security function at the university. Additional information could be gained by performing similar analyses and nearby educational institutions including state-owned, state-related and private universities.
**HCI and Visualization – Guided by Cognitive Science**

The selection of visualization methods must be guided by cognitive science’s understanding of the capabilities of the human analyst. The current visualization solutions are interesting, but do not address the main issues of awareness development and ignore the drawbacks of split attention, redundant and conflicting repetition, and the benefits of the modality effect. Interesting visualizations of direct network data, IDS alert data and the outputs of fusion-based algorithms are potentially helpful. However, the selection of which of these representations will aid in the development of situational awareness must be tested in the real-world environment. As the author has access to a community of computer and network security professionals, the testing and evaluation of selected visualizations will provided feedback on the usefulness of those in actual practice.

**Conclusion**

Computer and Network Security is a very complex and difficult task which is generally entrusted to the security analyst. To date, basic intrusion detection systems have been employed to provide the analyst with some kind of understanding of unauthorized tasks that are taking place on the network. However, these tools are inadequate for the task of developing situational awareness and are often produce the opposite effect as high volumes of alerts numb the analyst resulting in the system being ignored or disconnected. The promise of data fusion and new visualization techniques give hope that the analyst will one day be able to gain better situational awareness of the activities on the network through automated systems. The work required to get to this point will be interdisciplinary, combining the efforts of data fusion, visualization and cognitive science. The objectives that are sought will be best understood through rich data collection through the process of cognitive task analysis.
References


Snapp, S., Brentano, J., Dias, G., Goan, T., Heberlein, L., Ho, C., Levitt, K., Mukherjee, B., Smaha, S., and Grance, T. "DIDS (distributed intrusion detection system)-motivation, architecture, and an early
The Roles of Intrusion Detection and Data Fusion in Cyber Security Situational Awareness


Educational psychology review (10:3) 1998, pp 251-296.


