Issues and Challenges in Higher Level Fusion: Threat/Impact Assessment and Intent Modeling (A Panel Summary)

John J. Salerno  
Air Force Research Laboratory  
Rome, NY USA  
john.salerno@rl.af.mil

Shanchieh Jay Yang  
Department of Computer Engineering  
Rochester Institute of Technology  
Rochester, NY, USA  
jay.yang@rit.edu

Ivan Kadar  
Interlink Systems Sciences, Inc.  
Lake Success NY, USA  
ikadar@SystemsSciences.com

Moises Sudit  
Center for Multisource Information Fusion  
CUBRC/University at Buffalo  
Buffalo NY, USA  
sudit@cubrc.org or sudit@buffalo.edu

George P. Tadda  
Air Force Research Laboratory  
Rome, NY USA  
george.tadda@rl.af.mil

Jared Holsopple  
CUBRC  
Buffalo NY, USA  
holsopple@cubrc.org

Abstract – Many say that we live in the information age, but in reality if you ask any analyst today they would say we live in the data age. The amount of data being presented and displayed to the analyst is overwhelming – to a point that in many cases they are missing the salient or key activities of interest. Analysts are spending the majority of their time filtering through the data rather than performing analysis. Over the past 10 years, there has been an increasing emphasis on research in higher level fusion or what many are calling situation awareness. In this paper, we describe a collection of research addressing the challenges of enabling situation awareness. We will review our reference model and provide a discussion of a flow through the model to include how we can rank various activities based on their impact and threat. We also provide a number of algorithms that have been implemented and then tested and evaluated using a set of performance metrics.

Keywords: Impact Assessment, Threat Assessment, Situation Awareness, Knowledge of “Us”, Knowledge of “Them”, Intent, Opportunity

1 Background

In [6] we introduced a concept called the Data to Information Ration or DIR. The idea behind this metric was to measure the amount of compression or reduction that can be achieved by aggregating events/objects into groups and activities. As an example, consider a military force, composed of a number of geographically disperse units and containing a multitude of vehicles. If we were to track and display all of these individual objects, the display would very likely be black from the density. However, if we were able to group these objects into clusters, identify these clusters as combat units, and display icons that represent them, the display would be more readily understandable. A second example comes from the cyber domain. An analyst is typically required to try to identify an attack from within thousands of alerts (from Intrusion Detection Systems, log files, firewalls, etc.) However, if there was a capability that could aggregate alerts together into what we call attack tracks, an analyst could be reviewing hundreds of tracks rather than the thousands of observations/alerts that they do today [8]. But, aggregation alone is not sufficient. We need to also provide a way to draw attention to those attack tracks (or more generally, activities) that are important. This is done by ranking or prioritizing the activities by assessing the impact (and thus the potential threat) that an ongoing activity may have on us, our assets, or to the mission or business process. How do we define an impact or threat? What type of data, information or knowledge do we need to understand a given activity’s impact or threat? In the remainder of this paper we will discuss what we mean by impact and threat, a process based on our reference model, the types of knowledge needed to support situation assessment and enable a decision maker’s overall situation awareness, and a number of algorithms that have been implemented and then tested using a set of defined metrics. These metrics
are also discussed along with a specific use case. Before we continue we need to mention that some sections within this paper provide sample algorithms and in some cases experimental results while others provide a general description. In these latter cases, little or no research has been performed to date but we wanted to provide a comprehensive paper that discusses the current state of the art.

2 Defining Impact/Threat Assessment

Impact and threat have been specifically addressed by two different sets of researchers. Bosse, Roy and Wark [10] define situation assessment as “a quantitative evaluation that has to do with the notions of judgment, appraisal, and relevance.” Two components of situation assessment are: impact and threat assessment. Impact assessment is defined in [10] as “the force of impression of one thing on another; an impelling or compelling effect. There is the notion of influence: of one thing influencing another...impact assessment has to do with the estimation and prediction of the effects planned, estimated, or predicted actions by the participants...including interactions between action plans of multiple players.” Bosse, Roy and Wark go on further and define threat as “an expression of intention to inflict evil, injury, or damage. The focus of threat analysis is to assess the likelihood of truly hostile actions and, were they to occur, projected possible outcomes.”

Steinberg [11] states that: “Threat Assessment involves assessing situations to determine whether detrimental events are likely to occur.” Per the JDL Data Fusion Model, Threat Assessment is a level 3 data fusion process. Indeed, the original model [3] used ‘Threat Assessment’ as the general name for level 3 fusion; indicative of the importance of that topic. In subsequent revisions [4], the concept of level 3 has been broadened to that of Impact Assessment. Steinberg goes on to decompose threat into capability, opportunity, and intent as principal factors in predicting (intentional) actions.

- **Capability** involves an agent’s physical means to perform an act;
- **Opportunity** involves spatial-temporal relationships between the agent and the situation elements to be acted upon;
- **Intent** involves the will to perform an act.

A slightly different look at situation assessment is found in Kadar [30] who introduced the Perceptual Reasoning Paradigm (PRM). Viewed as a “meta-level information management system”, PRM consists of a feedback planning/resource management system whose interacting elements are: “assess”, “anticipate” and “preplan/act”. That is:

- **Gather/Assess** current, **Anticipate** future (hypotheses by learning), and **Preplan/Act** (predict) on information requirements as well as likely intent and threats,
- **Plan** the allocation of information/sensor/system resources and acquisition of data through the control of a specific distributed multisource sensors/systems resource manager,
- **Interpret and Act** on acquired (sensor, spatial and contextual) data in light of the overall situation by interpreting conflicting/misleading information to either identify or rule out the potential or existence of intent.

The elements of the fundamental PRM construct is shown in Figure 1 which depicts the interrelations among the elements described above for adaptive information gathering, anticipation, assessment, and prediction.

![Figure 1 - PRM Model](image-url)

Referring to Figure 1, the “assess module,” responding to dynamically managed and received multisource information, uses additional information from its associated knowledge base and from the “anticipate module” to form a database of “likely current situations” which include potential intents/threats. The “anticipate module” provides information on “likely future situations” that are used for short- and long-duration planning. This planning is based on the “likely current situations” from the “assess” module; prior, learned, process and tactical knowledge and associated hypotheses. The “likely current situation” information is fed back to the “predict module”, which provides “plans of actions and goals”. The “assess module” also provides current situations information to the “predict module” which, along with its knowledge base and likely future situations information from the “anticipate module”, (based in part on associated process knowledge), issues assessments, identifies potential intent and threats, and as needed, request actions from...
the resource manager for additional information to confirm or negate conflicting hypotheses thus closing the outer loop via the systems/sensors manager. In [7] we provided a set of definitions and a combined reference model based on many years of research in this area. The model, as shown in Figure 2, was built by combining the JDL Data Fusion model [3, 4] and Endsley’s SA Model [1]. As part of [7], we’ve refined how one can think of JDL Levels 1 and 2 as well as described differences between JDL Levels 2/3 and Endsley’s idea of projection.

2.1 A Snapshot in Time
Figure 3 expands upon the reference model and looks at it as a process at an instance in time. Observables are the input to the process that provides a view of what is occurring in the world (primitive elements of the environment). It is assumed that any attributes associated with the observables have been normalized, cleansed, and transformed into a form that can be used by the subsequent processes. The observables are cues into the activities that a decision maker needs or is interested in (and thus we refer to these as Activities of Interest, AOI) as a way to gain or maintain awareness. The AOI are based on missions, goals, policies, or in general the “things” of interest. We define the set of AOIs at an instance in time as the current situation. As observables enter the process, they are categorized and (1) associated with a new stage or step within an existing, ongoing activity; (2) associated with no existing activity and hence become the start of a new activity; or (3) can be a trigger leading to the combination, merging, or removal of existing activities. The aggregation process is similar to tracking individual objects (as defined by JDL Level 1) and why we consider this part of the process, even though dealing with events, still Level 1. Objects are no longer a physical entity like a tank but an activity – a collection of events and observables. The classical tracking problem of association also comes into play when associating an observable to an activity or step of an activity. At any given time, say \( t \), we have a set of ongoing activities (defined earlier as the current situation). We are now interested in analyzing the meaning of these activities. This is considered to be Situation Assessment (as shown on the right side of the reference model in Figure 2). The overall objective of Situation Assessment is to determine if any of the ongoing activities have an impact to "us" or if they can have a future impact to “us”. The former looks at the current activities and assessing the impact that the activities have had (follows the upper path of Figure 3).
Since these activities have already happened, we refer to this as “Damage” Assessment, i.e., has any of the identified activities caused a current impact and specifically has it caused harm that requires development of a recovery plan to resolve the effect(s) that the activity has caused. In order to accomplish this type of assessment, one not only needs the current, known activities, but also what each activity means to “us” (i.e., Does the given activity impact us in some way?). The data needed to perform this analysis is part of what we refer to as “Knowledge of Us” (see Section 4). Thus, this part of the process identifies to the decision maker whether there is a current impact to any capabilities or assets and whether there is an impact on performing the mission.

Additionally, a decision maker may be interested in a view of what the adversary (or competitor) is doing or may possibly do. This has generally been described as “getting inside the adversary’s OODA (Observe, Orientate, Decide and Act) loop”. The sooner we understand what the adversary can/might do, the sooner more options become available to the decision maker. This is addressed by the lower path of Figure 3. The first step of the process is to take each AOI and project it forward based on a priori knowledge included in the model. Here we don’t discuss time itself, i.e., we are not projecting the activities based on time, but rather on the next step in the activity. In some cases it could take milliseconds to go from one stage to another and in other cases it could be days, or longer. The number of stages that we look forward is defined under “Configuration Data”. Based solely on the models themselves, we have projected each current activity one or more steps forward; however, these projected or possible futures do not take into account whether they are plausible. In order to determine plausibility, we need to consider additional knowledge. We need both the “Knowledge of Them” (see Section 3) and “Knowledge of Us”. Specifically, we need to know if the adversary has the capability, capacity, the intent/goal, and have they exhibited similar behavior in the past. We also need to know whether they have the opportunity to accomplish the intent(s)/goal(s).
This opportunity is based in many cases on the vulnerabilities of us (provided as part of the “Knowledge of Us”). Thus, starting with the list of possible futures, we use the “Knowledge of Them” and “Knowledge of Us” to constrain the possible into the plausible for each activity of interest. But again what do these plausible futures mean to a decision maker? To answer this question, we again use the “Knowledge of Us” to identify potential impacts and threats to meeting our objective(s). From this portion of the process we get not only future potential impacts/threats but we can also use this knowledge to determine our future collection requirements. Using each of the plausible futures, we can identify the key differentiating events that will assist us in determining which of them may actually be unfolding. The key differentiating events can then determine collection requirements needed to increase the certainty in identifying whether a plausible future is occurring.

[1] One of the dangers in a reference model such as the one in Figure 2 is that it can be perceived as a sequential flow of data or information rather than a descriptive model of components and ideas. To help circumvent this danger, Figure 3 attempts to define a process flow and end products that are based on the concepts of the reference model as its framework. A primary feature of the process model is that it defines components that can be implemented as automated computer applications or shared human/computer systems that can then be tied together within system architectures. It also describes the flow of information and when key data sources come into play. We’ll now discuss in detail what is meant by “Knowledge of Them” and “Knowledge of Us.”

3 Developing the “Knowledge of Them”

In this section, we discuss “knowledge of them” and describe algorithms that have been implemented and tested. The following paragraphs discuss (1) Past Behavior, (2) Opportunity, (3) Capability/Capacity and (3) Intent/Goals.

3.1 Past Behavior

An important element in analyzing an adversary is to capture past behavior patterns, as a result of routines, habits, uses of toolkits, preferential human analytics, and so on. Several approaches can be used to capture adversary behavior. A logical choice of extracting patterns is to utilize data mining techniques; yet, these techniques are generally developed for offline process and limited for applications that require real-time or near-real-time performance. Another approach is to develop a set of adversary Course-of-Action (eCOA) templates for matching and correlating observables. While this works for problem domains where eCOAs are well defined and relatively stable, unbounded and fast-evolving eCOAs (such as cyber warfare) may require additional automated learning and clustering of behaviors.

In the presence of diverse, unknown, and fast-evolving adversary behaviors, we developed a near-real-time Fuzzy inference system that combines predictive outputs from a Variable Length Markov Model (VLMM) [14]. The system, called F-VLMM, captures sequential patterns exhibited in observed adversary activities, and predicts the next likely adversary actions by combining different feature estimates via fuzzy logic. VLMM is used because its ability to consolidate the patterns exhibited in different Markov orders and to be realized for near-real-time operation. Figure 4 shows the VLMM algorithm that gives the predicted probability of next character $c$ given a finite length context and the model suffix-tree. This prediction is done for each feature and the predictions for different features are then combined using fuzzy logic to reflect human analytics. For the case of predicting cyber attacks, Figure 5 shows the fuzzy membership functions used for predictions based on network service exposures and predictions based on subnet IP addresses, respectively. The fuzzified predictions are then combined using the fuzzy logic AND operator to produce the overall projection score that reflects the plausibility of a target being attacked in the near future.

---

```plaintext
predict( string context, character c )
accumulated_escape = 1
for i from 1 to context.length do
    sub_context = context[1, context.length]
    append_content = sub_context & c
    nom = # matches for append_content in suffix tree
    denom = # matches for sub_context in suffix tree
    probability = nom / denom
    escape_prob = compute_escape(context)
    prediction = prediction * probability * (1 - escape_prob) * accumulated_escape
    accumulated_escape = accumulated_escape * escape_prob
end for

escape_prob = compute_escape(context)
nom = # of matches for c in suffix tree
denom = total number of observed characters
probability = nom / denom
prediction = prediction * probability * (1 - escape_prob) * accumulated_escape
accumulated_escape = accumulated_escape * escape_prob
probability = 1 / alphabet_size
prediction = prediction * probability * accumulated_escape
return prediction

compute_escape( string context )
nom = # of characters following context in suffix tree
denom = # of matches for context in suffix tree
escape_prob = 1 / (1 - denom)
return escape_prob

Figure 4 – VLMM prediction algorithm
```

Through simulation over different networks and attack scenarios, F-VLMM achieved a percentile ranking of 80% - 90% when compared to ground truth. That is, the
actual attacked targets had a predicted rank of 80% - 90% which reflected the plausibility of that target being attacked.

Figure 5 – Fuzzy membership functions used to combine VLMM predictions for cyber attacks.

3.2 Opportunity

Opportunity assessment provides a critical understanding of the adversary but heavily relies on an analysis of blue’s vulnerabilities that could be exposed to red (“them”). Traditional human analysts estimate adversary opportunities based on experience and the knowledge of us. Often, opportunity is also co-assessed with adversary capability, to differentiate which opportunities can be actually exploited by the adversary. As problem domains become more complex and adversary behavior and capability become more unpredictable, a more robust, systematic, and automated opportunity assessment is needed.

Consider a virtualized terrain that models the operational environment consisting of accessibilities, vulnerabilities and interdependencies between red, blue and grey assets, in physical, cultural, and cyber forms. The terrain can be, for example, an ontology or graph-based model. By overlaying observables of each adversary activity onto this virtualized terrain, theoretically, one can extrapolate the progression of each activity, and further differentiate the hypothesized plausible futures based on the difficulty to extend the progression. In reality, constructing and dynamically adjusting such a virtualized terrain model with high accuracy may or may not be feasible for different problem domains.

A graph-based model representing the interdependencies between cyber assets has been proposed [13] and shown effective for impact assessment [43] and opportunity assessment [16]. Another example of virtualized terrain that can be potentially used for opportunity assessment is the human terrain [17, 18]. An emerging challenge of opportunity assessment is to develop a robust terrain model that spans across multiple domains.

In the case of assessing opportunity for cyber attacks, observables may be overlaid onto the virtual terrain [15] and extrapolated paths differentiated by examining firewall rules, service vulnerabilities, and the state of compromised and targeted systems [16]. In the process of searching for opportunities, or vulnerable targets, a computational challenge results from the traversal of the virtual terrain graph. An efficient breadth-first search algorithm, as shown in Figure 6, was developed to aggregate the attacked nodes and the associated rules, enabling the algorithm to run in near-real-time for cyber attacks.

The main benefit of automating opportunity assessment is to reduce the search space from all possible futures to plausible futures. Simulated studies have shown a reduction of the search space by 50% - 90%. The promising yet widely varying performance is due to the specific scenarios, where some attacks that are in early stages showed limited yet hard-to-differentiate futures, and others already deep in the network revealed many but differentiable futures.

Figure 6 – A breadth-first search algorithm used for efficient search of opportunities given an attack’s current state.

3.3 Capability/Capacity

While opportunity assessment systematically deduces exposed vulnerabilities, the outcomes must be further analyzed by cross-referencing to individual capabilities. It is critical to combine the capability and opportunity assessments, since individually they mean little to the estimation of adversary actions.

Traditional capability assessment relies heavily on adversary profiling (human, weaponry, etc.). When the adversary is hard to identify or little is known about them, adaptive statistical profiling, subsequence matching and clustering may be used. For analyzing cyber attacks, similarly capable attacks are identified by examining services being attacked and the severity of the attack. A statistical profiling algorithm based on attacked services [16] was developed. Multiple cyber features are used to determine similarity between attacks using an enhanced Longest Common Subsequence (LCS) algorithm [44]. In addition, Divisive Hierarchical Clustering (DHC), which has been effective in identifying communities for social network analyses [45], is used to group different types of attack maneuvers in a computer network.

The goal of automated capability analysis is not necessarily to identify the exact adversary capability, which requires expert knowledge, but to differentiate between categories of capabilities. Our results show that categorical capability assessment achieves a wide range
of effectiveness, depending on how well the feature space differentiates various types of adversaries. In a series of simulation studies, our algorithms showed success in differentiating ground truth critical attacks from noise attacks in different scenarios.

3.4 Intent/Goals

From [10], intent (and by implication goals) can be viewed as the determination or resolve to do a certain thing, or the state of mind with which something is done. Where, “the notion of intent revolves around the ideas of aim, goal, target, objective, plan, and purpose.” Furthermore, from the dictionary definition of purpose: “an anticipated outcome that is intended or that guides ones’ planned actions.”

As a matter of fact, the concept of intent has always been at the root of some of tort law’s most basic categories. All of the above definitions clearly show that “intent” is not directly observable.

As noted in [10], (paraphrased Op.cit.): intent is an intangible concept that cannot be directly observed by sensors. It must rather be inferred from the observation of other aspects of the world that becomes indicators of intent. In turn, this requires a model of intent that is sufficiently precise and detailed, providing an understanding of the numerous factors that drive intent. Previous efforts towards the definition of a model for intent have included a military perspective on commander's intent, a belief-desire-intent framework, planning-based models of intent, and explicit and implicit intent and subject matter experts (described in subsequent paragraphs). Having a desire alone does not allow the execution of intent. One needs an opportunity which is a favorable juncture of circumstances for taking actions. Opportunities make it possible to carry out one’s intent given sufficient capabilities [4, 5]. That is, an opportunity is the presence of an operating environment in which potential targets of an action are present and are susceptible to being acted upon [4].

Furthermore, adversary intent is arguably the most challenging task to estimate for most problem domains, and relies mostly on subject matter experts (SMEs) which is part of the Knowledge of "Them". Belief combination schemes can be employed to assist in fusing SME opinions. In addition, intent may be estimated in its worst-case sense [19]. Specifically, if little or no prior knowledge is known about the specific adversary, e.g., insurgents in unknown territory or random hackers from the Internet, one might assume the intended targets to be the most critical asset or information with respect to one or more missions. This mission-driven, worst-case intent estimation can be verified and adapted by tracing the adversary actions.

For example, let B and R be the sets of blue and red assets, respectively. Also let \( t_{a,m} \) denote the target end state of an asset \( a \) with respect to a mission \( m \). The impact score of a mission \( (I_m) \) then can be derived as a function of the asset states (or asset impact score) \( s_a \)

\[
I_m = \max \left\{ 0, \frac{\sum_{a \in B} w_{a,m}(s_a - t_{a,m}) + \sum_{a \in R} w_{a,m}(t_{a,m} - s_a)}{\sum_{a \in B} w_{a,m}(1-t_{a,m}) + \sum_{a \in R} w_{a,m}(t_{a,m} - 0)} \right\}
\]

where \( w_{a,m} \) is the weight function of asset \( a \) with respect to the mission \( m \) and \( \sum_{a \in B \cup R} w_{a,m} = 1 \). Note that \( I_m = 0 \) if \( s(a) = t_{a,m} \) for all assets, i.e., mission accomplished. We define the impact scores and asset states to be between 0 and 1, indicating the level of which the asset or mission is affected negatively by observed activities; the higher the value the more damage the asset or mission suffers.

Figure 7 shows an example where the estimated impact follows closely to the actual impact imposed by a simulated adversary activity.

Figure 7 - An example of worst-case intent estimation

4 Developing the “Knowledge of Us”

In its simplest sense, “knowledge of us” is information describing blue (ourselves) environments, assets, and missions that is useful for higher level analysis. In the cyber domain, this information could include, with many other possibilities; results of vulnerability assessments, logs or alerts from network management or security information management systems, device configuration reports, network topology maps, the services a device provides, and more recently, information about how particular network devices are used to support missions (or business processes). This would be information needed to perform a damage assessment as described in figure 2 and paragraph 2.1 as well as the information needed for performing an impact/threat assessment. We suspect that the information needed for a damage assessment may be limited to only how the effected device is used in currently performed tasks because it has already been affected. Knowing that it is vulnerable to an attack that already occurred is good forensic information but not overly useful in assessing the current impact or damage done to the device. The bulk of information would be used for the analysis necessary for assessing the impact/threat of plausible futures.

The greatest technical challenge regarding this information is its maintenance and currency. Knowing the impact to a business process weeks or even hours after a malicious activity occurred isn’t particularly useful especially if there is an immediate need for some decision, action, or continuation of the business process (nor is it particularly useful for basic network defense).
A large portion of the information listed above can already be collected to varying degrees of success using automated technologies. However, the information isn’t always current or actively updated because of network load considerations, difficulties in the automated techniques, and data complexities. Often there is a lack of any automated technique for capturing the information. In order to perform the depths of analyses described in this paper and to provide the greatest level of awareness to a decision maker, it becomes highly important that this data be available, accurate, and current. Additionally, there’s a further complicating issue when you consider that a single device can support a multitude of tasks that change over time or that may have periodicity. Just as a single device could be used for multiple purposes, a single service may require a collection of devices. All of which adds complexity to data availability, accuracy, and currency.

A collection of beginning work in this area has identified the general set of data useful for analysis; see Figure 8 [42]. The layers of the figure describe models of the network, models of systems, applications, users and models of the business processes with linkages between the layers describing dependencies. An ontology describing these layers and linkages was developed and is described in [43]. Other work on the models and linkages are described in [15, 44]. At issue still is keeping these models useful given the highly dynamic nature of operating domains. The tendency is to capture the data in snapshots that quickly become out of date or that rely on a human to directly maintain the information either cognitively or by updating data stores. Ideally, the data would be automatically collected, the models updated, and the collection be accomplished in such a way as to ensure it is current and accurate and doesn’t overload the network it describes.

The certainty of observations \((o)\) are used to determine the plausibility of a current red activity \((R)\) occurring which can be projected to multiple Plausible Futures \((F)\)’s for that activity. For each activity (whether current or future) we can calculate the Plausibility \((P)\) and the Impact/Threat \((T)\) of all activities to an asset or mission. This calculation is based on features \((f)\) of “knowledge of them” and “knowledge of us” which can be grouped into the components that define impact: capability, opportunity and intent.

5 Ranking Based on Impact/Threat

This section describes our methodology in measuring Plausibility and Impact/Threat as well as extrapolating to the observations that need to be collected to try to refine the assessment process. It is important to note that we are going to emphasize the mathematical techniques rather than the source and validity of the input parameters. We are not trying to minimize the criticality of trusting the input data, since regardless of the sophistication or correctness of the methods, “garbage-in, garbage-out.” In previous sections there are some discussions on the origin of the parameters that will be used in this discussion as well as the trustworthiness of such parameters. Figure 9 shows the different inputs that will be used in our calculations.

The certainty of observations \((o)\) are used to determine the plausibility of a current red activity \((R)\) occurring which can be projected to multiple Plausible Futures \((F)\)’s for that activity. For each activity (whether current or future) we can calculate the Plausibility \((P)\) and the Impact/Threat \((T)\) of all activities to an asset or mission. This calculation is based on features \((f)\) of “knowledge of them” and “knowledge of us” which can be grouped into the components that define impact: capability, opportunity and intent.

5.2 Plausibility of Current Situation

Researchers in the past have tried to combine the possibility of a red activity and the threat into a single unifying value. Although there is some merit in given decision-makers a single value in making a determination of the priorities he/she should focus, this aggregation should not be done when asking the question of occurrence of an activity in the present. The main reason is that all other factors are determined from understanding the certainty of the current situation. We can argue that the current situation either already impacted us (in which case we are responding) or it will impact us (in which case we are projecting). In the process of ranking Impact/Threat we are only interested on those situations in which we can affect the outcome of the things that “will” happen. So the first step is to understand the confidence (plausibility) that the current situation is truth. In [20] we suggested an Entropy-based approach to measure the credibility of a red activity being in a particular state.
\[
P = \frac{\log(n+1)-\log(n-\sum_{j=1}^{n} p_j+1)}{\log(n+1)}
\]

where

- \( n \) = number of objects
- \( p_j \) = certainty of observation \( o_j \) for \( j = 1,\ldots,n \)
- \( P_i \) = Plausibility of overall Activity \( i \) occurring
given \( p_j \)'s for \( i = 1,\ldots,m \)

So consider the example in Figure 10, where each node represents a Cyber Attack from one IP to another IP. The edges in the graph represent the steps as a hacker will go through the network. In this case there are two levels of observations: an atomic level represented by \( o_i \) and the aggregated level which are the real input to the credibility function. As we discuss in [20] we can use any Yager Aggregator \([21, 22, 23]\) which in turn gives the plausibility \( p_i \) of a node.

So the Entropy combination for the example can be calculated to be:

\[
P_i = \log(6) - \log(6 - (.68 + .75 + .3 + .51 + .35)) = 0.315
\]

(log throughout this document is base 2)

**Figure 10 - Calculation of Plausibility**

It is important to note that this combination provides an overall plausibility of the current Red Activity as a whole. This method allows us to measure subsets of the tree as needed for evaluating certain aspects of a current situation occurring rather than the overall activity. A final point of interest is the reason to use this Entropy combination rather than other type of combinations. The logarithmic nature of the function allows for decaying of the overall plausibility value rather than getting potential uniform stovepipe information. For example if all the plausibility in the nodes would be 0.5 and you use a simple average function the overall plausibility of the example in Figure 8 would be 0.5. On the other hand if we use the Entropy function the overall plausibility is 0.3. That is, there is an influence of each node’s value to the other nodes of the red activity under consideration.

### 5.3 Evaluating Impact/Threat

As discussed earlier impact and threat make sense from the perspective of the “potential” effect to an asset or mission from the current believable state of the adversary. So the features that compose capability, opportunity and intent can be projected at different times into the future. Moreover, this projection process can generate multiple plausible futures for a single current state depending on possible actions. So we will demonstrate how to evaluate impact and/or threat for a single plausible future and it should be straightforward to extrapolate to multiple futures. This calculation is based on Relative Entropy \([24, 25, 26]\) which is also called Kullback-Leibler divergence between two probability mass functions. In simple terms, we can view relative entropy, as the distance between two distributions. This distance \(D(r\parallel q)\) represents the “inefficiency” of distribution \(q\) assuming that \(r\) is the truth distribution. While the term distance gives an intuitive representation, it is well known that \(D(r\parallel q)\) is actually not a distance since it is not symmetric and doesn’t satisfy the triangle inequality. The mathematical representation of relative entropy is:

\[
D(r\parallel q) = \sum_{x \in X} r(x) \log \frac{r(x)}{q(x)}
\]

where

- \( X \) = Set of possible states

The way we implement this concept is by making each \( r_{ka} \) the probability of a feature \( k \) affecting asset or mission \( a \) (\( r_{ka}=1 \)) and \( q_{ka} \) the estimated probability based on the observations of that feature to that asset or mission. So the states in our case will be either True or False (\( X=\{0,1\} \)). Consider an extension of the example of Figure 10 as illustrated in Figure 11.

**Figure 11 - Evaluation of Impact/Threat**

In this example the dashed lines demonstrate potential moves that the adversary can perform to accomplish their intended goal. Also, we are going to assume that in the cyber domain the time it takes to go from every node to any other node is instantaneous, so there is no reason to talk about multiple projection times in the cyber domain.
For example let’s assume that feature 1 represents the capability of a hacker to go from node 3 to node 6 in Figure 11. In the worst case scenario, the probability of the hacker ability to go from 3 to 6 is \( r_{1,6} = 1 \). Also assume that our behavioral analysis has shown that the observed capability of the current observed hacker is very low to traverse from 3 to 6, so \( q_{1,6} = 0.2 \). So if there were no other feature we would like the relative entropy between these two binary distributions, which can be simplified to the following equation:

\[
D(r_{ka} \| q_{ka}) = (1 - r_{ka}) \log \left( \frac{1 - r_{ka}}{1 - q_{ka}} \right) + r_{ka} \log \left( \frac{r_{ka}}{q_{ka}} \right)
\]

\[
D(1 \| q_{ka}) = 0 \log \left( \frac{0}{1 - q_{ka}} \right) + 1 \log \left( \frac{1}{q_{ka}} \right) = \log \left( \frac{1}{q_{ka}} \right).
\]  

(4)

In general there will be multiple features, so we need to sum the relative entropy of all the features from the current situation of red to potential threat of blue assets. For example, consider the following list of features:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_1 )</td>
<td>Hacker Sophistication</td>
<td>Capability</td>
</tr>
<tr>
<td>( f_2 )</td>
<td>Link Vulnerability</td>
<td>Opportunity</td>
</tr>
<tr>
<td>( f_3 )</td>
<td>Host Data Criticality</td>
<td>Opportunity</td>
</tr>
<tr>
<td>( f_4 )</td>
<td>Hacker Physical Location</td>
<td>Intent</td>
</tr>
</tbody>
</table>

Table 1 - Features

For each of these features the probability of 1 is identified as the most threatening and 0 as the least threatening. Moreover, each feature can have an importance weight \( w_i \) in relation to the other features. Now, we can introduce the normalized value of threat to a particular node by normalizing the sum of the relative entropies for each plausible future in relation to the maximum possible threat:

\[
T = \max_{1 \leq i \leq m} \left( 1 - \frac{\sum_{k=1}^{h} w_k r_{ka} \log \left( \frac{1}{q_{ka}} \right)}{6.64 \sum_{k=1}^{h} w_k} \right)
\]

(5)

The constant 6.64 is the value for assuming that the minimum value that \( q_{ka} \) can take is 0.01 instead of zero which will give us numerical difficulties. Obviously the constant will change values if we assume that \( q_{ka} \) can take a smaller value. For example if \( q_{ka} = 0.001 \) then the constant will become 9.97. For the example in Figure 11, each of the four features we would have \( q_{1,6} \), \( q_{2,5} \), \( q_{3,4} \) and \( q_{4,3} \) and the corresponding \( D(1 \| q_{ka}) \) as shown in the table below for each asset that could be threatened.

A quick analysis of the table above shows some intuitive rankings between the different assets that are being considered. For example although node 6 has a weighted average threat value of 0.1 the relative entropy is much higher because there is a logarithmic augmentation since all of the values are complementing each other. When comparing the rankings, we get some sensible decisions such as although Node 7 has more important data than Node 8, the fact that Node 8 is more vulnerable than Node 7 makes Node 8 a more threatened asset. Finally, asset 3 demonstrates that if a number of features are threatening, but the most important is very low, there is a discount on the overall threatening value of that asset. Although a number of other functions could also be constructed to assess threat or impact, the one proposed has a number of provable properties that are intuitive and sensible for most domains.

6 Support for Collection Requirements

Once we have determined threat values for plausible futures, the next question that arises is, what observations and in what priority should they be collected in order to maximize the overall value of the mission and the accuracy of our current and future assessments? We have developed a mathematical model that will answer this question based on a number of factors. First let us introduce the model:

Variables:

\[
x_j = \begin{cases} 1 & \text{If observation } j \text{ is collected } \quad j = 1, \ldots, n \\ 0 & \text{Otherwise} \end{cases}
\]

\[
y_u = \begin{cases} 1 & \text{If Collection Requirement Action (CRA) } k \text{ is taken } \quad u = 1, \ldots, d \\ 0 & \text{Otherwise} \end{cases}
\]

\[
v = \text{Value of the mission}
\]

Parameters:

\[
c_u = \text{Cost of executing CRA } u \quad (u = 1, \ldots, d)
\]

\[
t_u^r = \text{Response time of CRA } u \quad (u = 1, \ldots, d)
\]

\[
t_j^p = \text{Process time for observation } j \quad (j = 1, \ldots, n)
\]

\[
B = \text{Budget}
\]

\[
T^R = \text{Limit of response time}
\]

\[
T^P = \text{Limit of process time}
\]

\[
v_j = \text{Value of observation } j \text{ to overall mission given current mission state } \quad (j = 1, \ldots, n)
\]

\[
O_u = \text{Set of observations } j \text{ that will be collected through CRA } u \quad (u = 1, \ldots, d)
\]

Table 2 - Plausible Threatened Assets

<table>
<thead>
<tr>
<th>Plausible Threatened Assets</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>6: F→X</td>
<td>0.1</td>
</tr>
<tr>
<td>7: X→Y</td>
<td>0.6</td>
</tr>
<tr>
<td>8: E→Z</td>
<td>0.6</td>
</tr>
<tr>
<td>3: J→S</td>
<td>0.9</td>
</tr>
<tr>
<td>4: G→H</td>
<td>0.1</td>
</tr>
<tr>
<td>5: I→K</td>
<td>0.4</td>
</tr>
<tr>
<td>6: J→L</td>
<td>0.4</td>
</tr>
<tr>
<td>7: K→M</td>
<td>0.9</td>
</tr>
<tr>
<td>8: L→N</td>
<td>0.5</td>
</tr>
<tr>
<td>9: M→N</td>
<td>0.8</td>
</tr>
<tr>
<td>10: N→O</td>
<td>0.49</td>
</tr>
</tbody>
</table>

Table 2 - Plausible Threatened Assets
Metrics

How well does a system work? Once we have defined the system and its purpose we can then develop metrics to evaluate how well it performs. Such metrics were described in [7] and classified into four categories or dimensions: (1) Confidence, (2) Purity, (3) Cost Utility and (4) Timeliness.

The first two measures tell us how well the lower level algorithms work in combining the observations into activities and thus how well our system is doing in correctly identifying and tracking the activities or overall situation. The third category provides us an indication as to how well the assessment capabilities work, while the fourth provides an overall measure of the performance of the system to provide the right information in a sufficient amount of time for the decision maker to make a decision.

\[
\text{Confidence} \text{ is a measure of how well the system detects the true activities. It is composed of three factors: (1) Recall, (2) Precision and (3) Fragmentation. Recall measures the percentage of activities correctly identified by the Level 1 system (Correct Detections) in relation to the number of “real” activities as defined in the ground truth (Known Activities) and is defined in Equation 6.}
\]

\[
\text{Recall} = \frac{\text{Correct Detections}}{\text{Known Activities}} \quad (6)
\]

Precision is the percentage of activities correctly identified by the Level 1 system (Correct Detections) in relation to the total number activities (Detected Activities) provided by the Level 1 system and is defined in Equation 7. Fragmentation is defined as the percentage of activities reported as multiple activities that should have been reported as a single activity and is defined in Equation 8. Confidence is typically reported as a probability.

\[
\text{Precision} = \frac{\text{Correct Detections}}{\text{Detected Activities}} \quad (7)
\]

and

\[
\text{Frag} = \frac{\text{No. Results that Identify the same Known Activity}}{\text{Detected Activities}} \quad (8)
\]

The second dimension, Purity, characterizes the quality of the proposed activities (i.e., were the observations correctly matched and connected to the right activity track). To measure purity we define a “Mis-assignment Rate” as the percentage of evidence or observations that were incorrectly assigned to a given activity and Evidence Recall as the percentage of evidence or alerts detected in relation to the “total known” events or observations. After applying purity metrics we found that they added no significant value in our assessment of how well the system worked and thus no longer use them.

The third dimension, Cost Utility was not used as defined in [7]. After implementing this metric, we found that it provided little information about the performance of a system and subsequently we redefined it under our cyber work as the “Attack” score. What we were trying to measure was how effective the system is in ranking the activities of interest in a predetermined order. For example, in the cyber domain we are interested in those activities (or what we called potential attacks) that were most detrimental to our operations based on either impact/threat or mission. The cyber work is described in detail in [9]. This ranking can be based on “most likely” or “most dangerous”. Based on this, we have again renamed this metric to “Activities of Interest (AOI)” Score which we believe more accurately describes what
insight it provides and implies that we are not interested in attacks only.

The final and fourth dimension is Timeliness. This dimension attempts to measure the ability of the system to respond within time requirements of a particular domain. For example, we are interested in how quickly the system can identify the activity or activities of interest. But this value alone is meaningless unless we take into account whether this time is soon enough for us to take action. Combined it is hoped that these measures can correctly characterize how well the system performs. Figure 12 shows how the given metrics are mapped against our reference model.

![Figure 12. Metrics Mapped to Reference Model](Image)

7.1 Activities of Interest (AOI) Score
Focusing on the third category, how can the AOI score provide us insight into our assessment process? Recall that the objective of the AOI Score is one to measure how well the system has identified the activities that are most important to us. The criteria for what is most important are dependent on the decision maker’s or analyst’s needs. In determining potential adversary Courses of Actions (COAs) the decision maker is interested in which COAs are most likely and which are most dangerous – each answer using different information as part of the analysis and providing the decision maker with a different perspective. For example ranking the “most likely”, needs to take into account the adversary’s capability/capacity (what can they do), intent (what is/are their goals) and opportunity. Most dangerous must not only take into account the information required to determine the most likely but also Blue’s vulnerabilities and an assessment as to whether the adversary will be successful and to what extent will Blue incur damage. The ranking of each can also change between the current analysis and the projection of future impact/threat.

We will illustrate the basic concept through an example from our cyber work. In our example individual observations are provided by such systems as Intrusion Detection Systems (IDSs), Netflow, System Logs, etc. and aggregated into AOI tracks. This would be the output of our level 1 capability and can be simply displayed as a list (one form of output) of the activities. If no other knowledge is available, this list would be just that, a list of activities in no order with no indication of “importance”, however, in the real world, the number of activities can be very large (hundreds to thousands based on pruning or lack of) – some ordering is needed to place the most important ones that the analyst or decision maker should be concerned with at the top. In our cyber example we would like to prioritize the activities based on their current of future impact/threat. So how can we measure both what the additional knowledge provides and how well our ranking algorithms work?

7.2 Measuring How Well We Are Doing
We can compute the AOI score by using Equation 9 shown below.

\[
AOI\ Score = \frac{NAOI \times NA - \sum_{i=1}^{NAOIR} P_i}{NAOI \times NA - \sum_{i=1}^{NAOIR} i}
\]

(9)

where

- NAOI = Number of Activities of Interest in Ground Truth
- NAOIR = Number of AOIs in Results
- NA = Total Number of Activities in Ground Truth
- \(P_i\) = Position of the \(i\)th Activity of Interest

If any (or all) of the AOIs are not part of the results list or if their position is greater than the total number of activities in the ground truth, we set the position value \(P_i\) for those AOI(s) equal to the Total Number of Activities in the Ground Truth. By adding this condition, if there are no AOIs included/identified in the results list, the AOI score will equal 0. Whereas, if there is only a subset of the actual AOIs identified, the system will get credit for only those.

Let us consider a simple example. Given the ground truth and the results provided as output by our Level 1 system (Situation Recognition and Identification) without any further knowledge or assessment as shown in Table 1, we can compute a baseline. This baseline can then be used to determine the value added by additional knowledge and processing provided by situation assessment.

We first compute how well our level 1 system has identified the activities that make up what we call the
situation. To compute recall we use Equation 1 (6/8 = 0.75 or 75%). In this case, our system has identified more activities than what actually is occurring (one not even there, a false positive and the other is a fragment). Optimum recall would be 100%. Let us next examine precision. We compute precision using Equation 2 (6/6=1.0 or 100%). In this case the system has correctly identified all the activities in the ground truth. The last metric we will examine within the first dimension is fragmentation. Fragmentation is computed using Equation 3 (1/8=.125 or 12.5%). In summary if we had a capability that provided the results as shown above we would have a recall of 75%, a precision of 100% and a fragmentation of 12.5%. What do these numbers tell us? There were 25% more activities identified than there actually were and of the 25% half of them (12.5%) were because they should have been associated with an existing activity (fragments). Let us now consider the question “Of the activities that have been identified are there any of them that we should be concerned with?” Our next metric, AOI score, will tell us how we are doing in answering this question.

Table 3 - Data for Example 1

<table>
<thead>
<tr>
<th>Ground Truth</th>
<th>Proposed Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>GT0 Activity 1 (AOI 1)</td>
<td>R0 Activity 4</td>
</tr>
<tr>
<td>GT1 Activity 2 (AOI 2)</td>
<td>R1 Activity 3</td>
</tr>
<tr>
<td>GT2 Activity 3</td>
<td>R2 Activity 2 (AOI 2)</td>
</tr>
<tr>
<td>GT3 Activity 4</td>
<td>R3 Activity not part of GT</td>
</tr>
<tr>
<td>GT4 Activity 5</td>
<td>R4 Activity 5</td>
</tr>
<tr>
<td>GT5 Activity 6</td>
<td>R5 Activity 1 (AOI 1)</td>
</tr>
</tbody>
</table>

Let us first compute a baseline AOI score where we have no further information about the activities. To compute the AOI score we have the number of activities of interest in the ground truth as 2 and the total number of activities in the ground truth as 6. The remaining 4, that we do not consider to be AOIs, are either activities that have no or minimal impact or threat to us. We next need to know the ordering the system identifies for those activities that are of interest (as identified by in the results). In our example the first “important” activity is third in the proposed list while the second “important” activity is in the sixth position (Table 1). We simply add these two values together for a value of 9. The final value we need to compute is just the geometric sum of the “important” activities which is simply 2 activities: 2+1=3. Substituting in the values into Equation 4 we have a value of ((2)(6)-9)/((2)(6)-3) = 3/9 or 0.33. This means that an analyst or user would have to consider roughly 2/3’rds of the activities before “seeing” the most important activities.

Now assume that we are provided additional knowledge about the activities of Table 3 after further analysis is provided by Level 2, Situation Analysis. For example if we were looking at a list of cyber activities we might have more information on the computers (their connectivity, operating system, services and applications) and what mission they perform. Using this knowledge we can then determine how important the activity is based on impact or threat to the mission. Vulnerabilities come in to play also; for instance, am I vulnerable to the attack being executed against me making that activity more important if I am vulnerable and less important if not. To compute the value of this additional information in the ranking we simply apply Equation 9, a second time. Table 4 provides an example.

Table 4 - Data for Example 2

<table>
<thead>
<tr>
<th>Proposed Activities</th>
<th>Ground Truth</th>
</tr>
</thead>
<tbody>
<tr>
<td>R0 Activity 4</td>
<td>GT1 Activity 2 (AOI 2)</td>
</tr>
<tr>
<td>R1 Activity 3</td>
<td>GT2 Activity 3</td>
</tr>
<tr>
<td>R2 Activity 2 (AOI 2)</td>
<td>GT3 Activity 4</td>
</tr>
<tr>
<td>R3 Activity not part of GT</td>
<td>GT4 Activity 5</td>
</tr>
<tr>
<td>R4 Activity 5</td>
<td>GT5 Activity 6</td>
</tr>
<tr>
<td>R5 Activity 1 (AOI 1)</td>
<td>GT0 Activity 1 (AOI 1)</td>
</tr>
<tr>
<td>R6 Activity (Frag, should be part of Activity 2)</td>
<td>GT1 Activity 2 (AOI 2)</td>
</tr>
<tr>
<td>R7 Activity 6</td>
<td>GT2 Activity 4</td>
</tr>
</tbody>
</table>

The only value that changes in our equation is the sum of the positions of the activities of interest. This new value is 2+3=5 and our new value for the AOI score becomes ((2)(6)-5)/((2)(6)-3) = 7/9 or 0.78. One can easily see that if we got the most important activities in positions 1 and 2 then the AOI score would be 1.0. Similar computations can be made for Level 3, plausible futures and their impact/threat. Similar to Level 2 we need only ground truth for the given scenario. We note here that the AOI score does not distinguish between how important the AOIs are just...
the fact that they are AOI. Extensions to this metric can be made to take into account such a scenario.

8 Example AOI Analysis using Cyber Security

We can illustrate the benefit of incorporating impact assessment through showing improvements in AOI scores by introducing impact assessment techniques for a given data set. Ground truth for four different scenarios was compiled from scenarios simulated over a real computer network. Each scenario consisted of at least one “stepping stone” attack, which is an attack by which a hacker compromises at least one computer on a target computer network and use it to hack into one or more computers in that same target network.

Ground truth for each of the scenarios consisted of a number of activities corresponding to normal network traffic as well as general reconnaissance traffic from the Internet. Each scenario consisted of at least two AOIs, each of which corresponded to a successful stepping stone attack.

The data set was evaluated with two integrated information fusion tools: INFERD (Information Fusion Engine for Real-Time Decision Making) [20] and FuSIA (Future Situation and Impact Assessment) [16]. INFERD is a domain agnostic aggregation and tracking tool that utilizes a behavior-based model, or guidance template, to correlate and aggregate observables together to identify and rank activities. FuSIA is an impact assessment tool that incorporates Knowledge of “Us” and Knowledge of “Them” to identify the impact and plausible futures of activities using some of the algorithms presented in this paper.

INFERD’s guidance template requires minimal a priori knowledge regarding the Knowledge of “Us”. As a result, it is unable to implicitly determine false positive actions (observables that indicate malicious behavior that were actually not harmful to the assets or missions). It is, however, able to rank activities based on the “depth” of the attack, which corresponds to the perceived severity of the attack based solely on the actions. The INFERD guidance template can be separated into multiple stages, where a higher stage indicates a higher threatening activity than one that has reached a lower stage. We can therefore define the depth measure as follows:

\[
\text{depth} = \frac{\text{highest\_stage}}{\text{total\_stages}+1} \tag{10}
\]

Where highest\_stage is the maximum stage the attack has reached and total\_stages is the total number of stages in the guidance template.

FuSIA is able to receive correlated tracks from INFERD and calculate an impact to different hosts in the network. Knowledge of “Us” is modeled through the virtual terrain [15], which models vulnerabilities, firewall rules, and other computer network components. By incorporating the virtual terrain, FuSIA identifies impacts to different hosts on a computer network, which can be used to refine activity rankings. For this analysis, we consider the measure “max host impact (MHI)” which is calculated as follows:

\[
\text{MHI} = \max_{h \in H} i_h \tag{11}
\]

Where H is the set of hosts on the defended network and \(i_h \in [0,1]\) corresponds to the impact score for host h. The impact score is calculated using the virtual terrain through the analysis of known vulnerabilities and firewall rules.

We now analyze the performance of the rankings using the AOI score outlined in the previous section by comparing the depth score (INFERD only) and max host impact (INFERD and FuSIA). It should be noted that multiple activities can have the same value. When this happens, it makes sense to calculate an “upper” AOI score and a “lower” AOI score, similar to the way percentile rankings are traditionally assessed. This range of possible AOI scores assumes that ties are broken arbitrarily, so it is possible that an AOI may be arbitrarily ranked higher (best case) or lower (worst case) than all other activities with the same score. In Figure 13, each metric is displayed by a box, where the top of the box is the upper AOI score and the bottom is the lower AOI score for that scenario. In all four scenarios, the use of impact assessment increased the AOI scores and reduced the range of AOI scores. Even with impact assessment, the INFERD-FuSIA system still ranked the AOIs in the middle of the activities.

However, further inspection of the scenario details shows that one AOI was not detected because it only had observables from a sensor that was not considered by the INFERD-FuSIA system. Therefore, this result makes sense since there were no usable observables to infer the presence of an attack. Scenario 4 shows that
both measures were able to rank the AOIs very high, but the introduction of impact assessment was able to filter out false positives that decreased the range of possible AOI scores.

9 Conclusion

This paper has described a reference model and a flow or thread through it for a given time, t. The SA Reference Model provides a set of definitions that can serve as a reference for describing systems that aid with SA while the Process Model captures a process flow at a single point in time. Together, the two models provide a common set of definitions for situation awareness. The goal in presenting the reference model and process thread was to describe how one can identify significant activities of interest or of concern to oneself and to one’s goals/objectives. In doing so, the hope is to be able to focus the analyst’s attention onto what is important, thus minimizing the current the work overload and maximizing the decision maker’s situation awareness. Products include plausible adversarial futures ranked based on threat (Most “Dangerous” and Most “Likely” and generally refer to as enemy Courses of Actions, eCOA), a list of collection requirements and possible tip offs based on differentiating events and anticipated futures.

10 References


[38] Little, E.G. and Rogova, G.L., An Ontological Analysis of Threat and Vulnerability,
Proceedings of the 9th International Conference on Information Fusion (Fusion 2006), Florence, Italy, 10-13 July 2006.


