DTB Project: A Behavioral Model for Detecting Insider Threats

Paulo C. G. Costa, Kathryn B. Laskey, Mehul Revankar, Sepideh Mirza, Ghazi Alghamdi
George Mason University
Systems Engineering and Operations Research Dept., MSN 4A5
4400 University Drive
Fairfax, VA, 22030-4444, USA
[pcosta, klaskey, mrevanka, smirza, galghamd]@gmu.edu

Daniel Barbará, Thomas Shackelford
George Mason University
Information and Software Engineering Dept., MSN 4A4
4400 University Drive
Fairfax, VA, 22030-4444, USA
[dbarbara, tschakel]@gmu.edu

Edward J. Wright
Information Extraction and Transport, Inc.
1911 North Fort Myer, Suite 600
Arlington, VA, 22209, USA
ewright@iet.com

Keywords: Methods for Counter Denial and Deception, Novel Analysis Methods, Multi-Entity Bayesian Networks, Data Mining, Document Relevance, Behavior Modeling and Simulation, All Source, Counter Intelligence

Abstract

This paper describes the Detection of Threat Behavior (DTB) project, a joint effort being conducted by George Mason University (GMU) and Information Extraction and Transport, Inc. (IET). DTB uses novel approaches for detecting insiders in tightly controlled computing environments. Innovations include a distributed system of dynamically generated document-centric intelligent agents for document control, object-oriented hybrid logic-based and probabilistic modeling to characterize and detect illicit insider behaviors, and automated data collection and data mining of the operational environment to continually learn and update the underlying statistical and probabilistic nature of characteristic behaviors. To evaluate the DTB concept, we are conducting a human subjects experiment, which we will also include in our discussion.

1. Introduction

The overall idea of the DTB project is to model user queries and detect situations in which users in sensitive positions may be accessing documents outside their assigned areas of responsibility. This novel approach to detecting insider threats assumes a controlled environment in which rules for accessing information are clearly defined and, ideally, tightly enforced.

Although such environments provide little encouragement to insider threats, unusual access patterns are not easily perceived. In fact, documented cases in which insiders using unsophisticated tactics to outsmart standard security systems (CNN.com 1998, 2001) leave a very uncomfortable open question: how about the sophisticated ones?

Catching more elaborate patterns that might be characteristic of users attempting illegal activities such as disclosure of classified information is a daunting task that we tackle with a powerful inference method. The flexible

Funding for the DTB Project is provided by the Advanced Research and Development Activity (ARDA), under contract NBCHC030059, issued by the Department of the Interior. Additional support has also being provided by the US Navy.

modeling framework provided by multi-entity Bayesian networks (MEBN) makes it a natural candidate for modeling this complex problem. MEBN's ability to represent complex interrelationships among entities and its mathematically sound inference make it a required for feeding its inference engine proved to be a perfect match to our challenges.

2. Multi-Entity Bayesian Networks

MEBN logic (Laskey 2004) integrates First Order Predicate Calculus with Bayesian probability. It expresses probabilistic knowledge as a collection of MEBN fragments (MFrags) organized into MEBN Theories (MTheories). An MFrag represents a conditional probability distribution of the instances of its resident random variables given the values of instances of their parents in the Fragment graphs and given the context constraints.

A collection of MFrags represents a joint probability distribution over an unbounded, possibly infinite number of instances of its random variables. The joint distribution is specified by means of the local distributions together with the conditional independence relationships implied by the fragment graphs. Context terms are used to specify constraints under which the local distributions in an MFrag apply. A collection of MFrags that satisfies consistency constraints ensuring the existence of a unique joint probability distribution over its random variables is called an MTheory. MTheories can express probability distributions over truth-values of arbitrary First Order Logic sequences and can be used to express domain-specific ontologies that capture statistical regularities in a particular domain of application.

In addition, MTheories can represent particular facts relevant to a given reasoning problem. Conditioning a prior distribution represented by an MTheory on its findings is the basis of probabilistic inference with MEBN logic. Bayesian conditioning provides built-in machinery for learning structure and parameters of an MTheory from data consisting of instances of the MTheory's random variables.

Currently, MEBN logic is being implemented by IET's Quiddity*Suite™, a knowledge-based probabilistic reasoning toolkit, which we used to implement our models (see Alghamdi et al. 2004 for an initial discussion on the modeling efforts).

3. DTB Architecture

The DTB project integrates three different software applications, which we describe by following the general information flow under the DTB architecture depicted in Figure 1.

Initially, information on queries and overall system usage (e.g. document accesses, login times, copy and paste operations, etc.) is collected with Glass Box, a Java-based user monitoring application available to researchers on ARDA's Novel Intelligence from Massive Data project (NIMD) at http://glassbox.labworks.org.

The general information collected on system usage is stored into the Glass Box Data Store, whereas specific data regarding the user's queries goes to the Wolfie data store. Wolfie is a Data Mining application that assesses the relevance of each user query to his/her assigned task. Finally, the results from Wolfie are used as evidence to feed the Insider Bayesian Network model (IBN), a collection of MFrags (i.e. an MTheory) written in Quiddity*Suite™ and stored in the MFrags Knowledge Base depicted in Figure 1. IBN is a MEBN behavioral model that uses the evidence provided on each user to assess the likelihood that his/her behavior patterns over a series of sessions is an indicative of malicious intent.

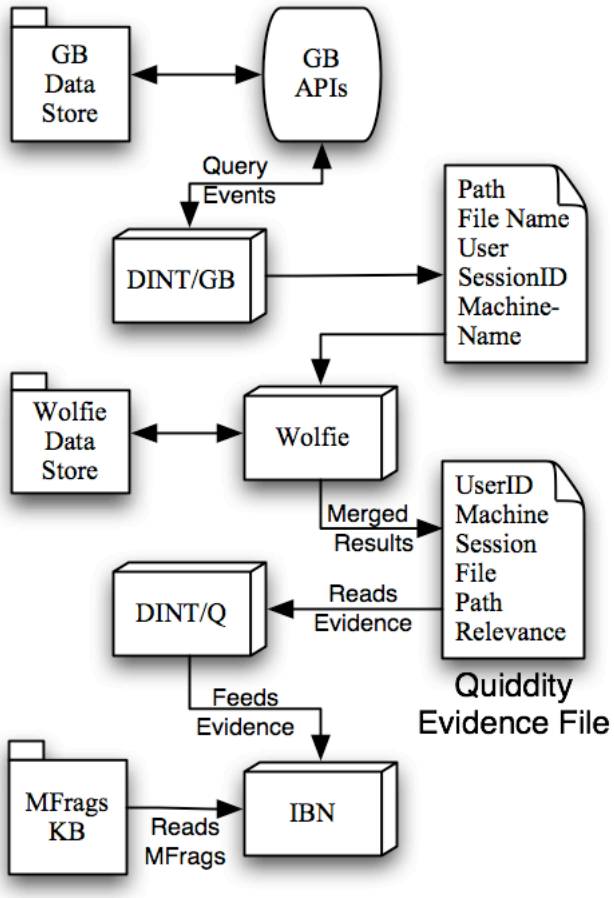

Figure 1 – The Overall Architecture of the DTB Project

In order to integrate those different applications, we developed a Data Integration Module (DINT), which controls the information flow within the DTB architecture. Apart from Glass Box, all the other modules have been developed within the DTB project, so we now present it in more detail.

3.1 The Wolfie Data Mining Algorithm

Wolfie is a text extraction and analysis software package that consists of multiple java and SQL programs. The backbone of the system is a relational database that is used to store both raw and processed data. Java pro-

gramming language was used to develop a file parser to extract document information and a data analyzer that provides detailed statistics linking a given document with one or more topics, based on data retrieved from SQL routines. A baseline set of documents was used to train and test the relevance analyzer. In the sequence, new Java routines are used to determine the relevance of the remaining documents with respect to a set of topics. In our ongoing research, we have tested multiple text classification methodologies, analyzing each one to determine the most suited for determining document relevance.

When a new document is introduced its relevance is determined using both parametric and non-parametric models as well as other mathematical routines to find a level of confidence for each probability found. Where this differs from most document retrieval and analysis programs is that it provides a probability and confidence level for every topic in the topic tree rather than a binary answer to whether a document is relevant to a topic or not.

As documents can belong to either single or multiple topics, Wolfie provides a probability and a confidence level for each topic in the topic tree. This information along with other data collected from the system is then passed to IBN, which determines the likelihood of a user is exceeding her/his authority or access privileges.

3.2 IBN Model

Modeling insider user behavior is a complex problem that involves reasoning about many objects and their relationships as well. In such cases, the use of standard Bayesian networks to encompass all problem instances is impractical. Standard BNs have only propositional expressiveness, which basically limits its use to problems where the same set of random variables applies to all problem instances and only the evidence is different between problems. In the DTB project, we not only have to deal with a large number of objects in the inferential process, but the relationships between these objects differs from one problem instance to another as well. Thus, we have opted for using a MEBN-based solution, which enables us to have First Order Logic expressiveness combined with the ability of performing plausible reasoning.

The IBN model consists of a MTheory that describes the insider problem domain. Each MFrag within that MTheory corresponds to a segment of insider user behavior. Breaking down the problem to fragments gives us the ability to thoroughly examine and investigate each fragment and also support modularity and reusability. Our current model consists of seven MFrags that model queries and document accesses performed by users. We will discuss each MFrag, explaining its structure and role.

User—The user MFrag represents an individual user's profile. We currently use motive, intention, assignment and other activity when determining threat. Most of these slots are references to other fragments but it is useful to have a single user reference fragment for use throughout the inferential process.

User Background—We currently model three areas that may serve as indicators that a user is likely to be a threat: political activities, personal background, and financial background. In each area we made fairly coarse distinctions such as serious concerns, minor concerns, and no concerns. Here, our main focus was on measurable computer usage behaviors, and we consider those parts of the model as placeholders for a more detailed model that might be inserted if desired.

User Intention—For this network, we classify users' intentions as either "normal" or "threat." We assume that users' intentions may change over time. We assume a given user has a global intention that does not change over the time represented by the model and a session intention that may change from one login session to the next. The session intention is influenced by both the global intention and the previous session's intention. The session intention influences the queries a user performs and thus can be inferred from the pattern of documents accessed over time.

User Normal Assignment—Each user has an assigned region and an assigned task. We have created fictional regions and tasks for our proof of concept network but we could easily substitute real regions and tasks. Our approach is scalable and we can increase the numbers of both regions and tasks as necessary.

User Clandestine Assignment—Our goal is to build a model that not only identifies malicious users but also indicates the nature of the potential threat. A malicious user will have a clandestine region and a clandestine task in addition to the normal assignment. Currently we model the threat by looking for an information source the malicious user may be trying to identify, as well as any regions and/or tasks which the user is showing interest in addition to his or her assigned region and task. The user's other intention also influences the documents he or she accesses, and can be inferred from the pattern of document retrieval behavior.

Document—Documents have sources, region and task classifications. Each document is rated with respect to how relevant it is to each task and region, providing a measure of how relevant it is to each of the possible tasks and regions. Techniques to measure document relevancy that can be used for this purpose are already covered in other research efforts (e.g. Anderson et al. 2004, Chakrabarty et al. 1999).

Login Behavior—User login events such as login/logout times, failed login attempts, and session duration will be monitored for unusual behaviors. For this model, we assume that a login time outside working hours indicates abnormal behavior. Based on the historical profile of the users, this MFrag captures the typical session duration of users for any given session. The historical profile of a user will indicate if the time spent for a specific session matches his/her normal habits. A mismatch will increase the belief on an abnormal behavior. It will also increase our belief on a masquerader. If a user failed to login once or twice then this will increase the

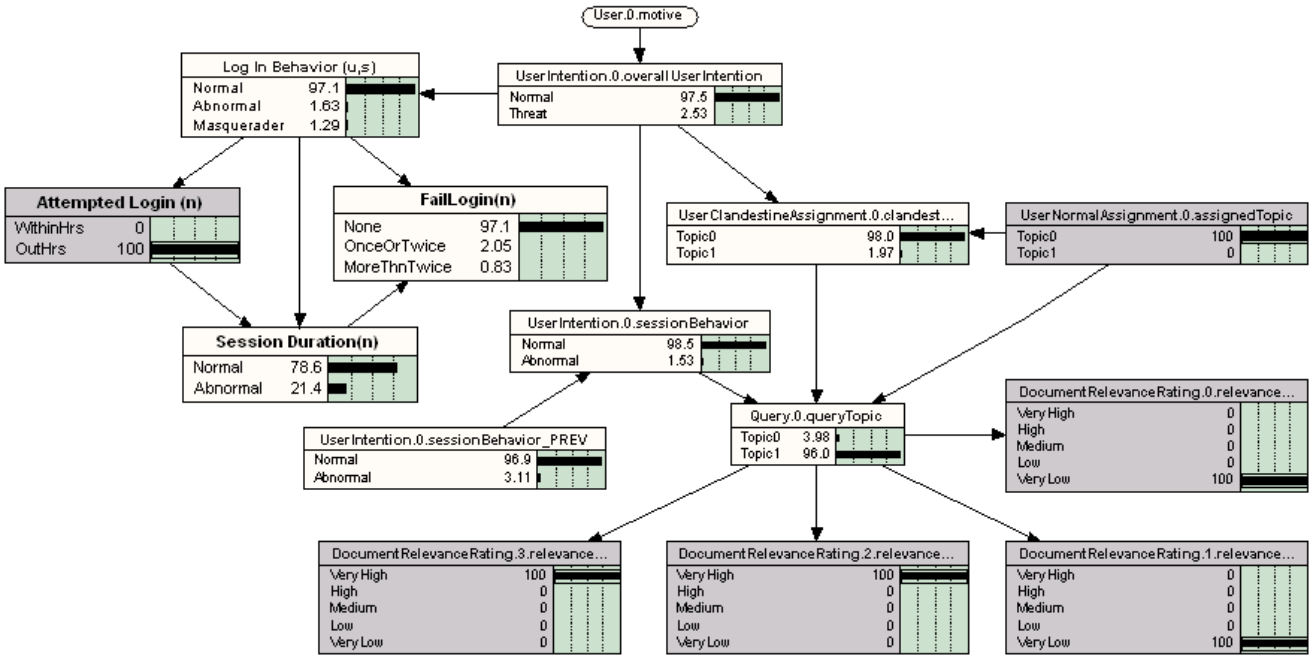

Figure 2 – Resulting IBN MFrags after a query was made

belief on abnormal behavior but not necessarily on a masquerader event (normal users tend to misspell their passwords). However, if a user failed to login more than twice then this will increase our belief that we may have a masquerading event.

Query – Users perform queries that result in document accesses. We assume in any given query that the user is seeking information about a source, task and region. The query source represents a source that provides intelligence information, as opposed to the author of a document. If a user is attempting to identify a given source, then whether or not the user explicitly queries on the source, task or region, the user is likely to access documents citing this source. A query is also likely to return a document relevant to the query task or region.

Figure 2 shows the MFrags assembled into a unified model for a single session, a single query, and a single document access. In this example, the user's assigned topic has been provided as evidence, as well as the results of a query, shown by the setting of evidence on the four Document Relevance Rating nodes, and on the belief of an Attempted Login.

The IBN model accepts external preformatted data from collection systems, such as Glass Box, that describe insider activities (events) via the DINT/Quiddity module (DINT/Q). The IBN model processes these events to infer a user's session and global intention. For each user that is logged into the system, IBN will construct a user situation-specific profile to monitor user behavior. In order to achieve a continuous flow of evidence to support the construction of these situation specific profiles we rely on DINT.

3.3 The Data Integration Module (DINT)

The three different software modules of the project were not originally designed to interact with each other, an issue we addressed with the development of DINT. As an example of what types of challenges DINT faces, let's suppose that Glass Box recorded one particular file access event to a file named *terrorism_in_countryX.pdf*. This information is not a valid input for IBN, which expects a numerical value as evidence to be used to infer the posterior probability of that document being related to terrorism to CountryX. In this specific example, DINT would transform the collected data (i.e. log of file access events) into a format that the IBN model understands.

This transformation is partly done by the Wolfie algorithm and partly by DINT itself. Wolfie estimates the degree of relevance of a document to a topic from the topic list (e.g. terrorism) whereas DINT currently looks at the session login logout times of the user and determines whether they are in range or out of range with respect to the predetermined standards specific to the agency using the DTB system. As DINT evolves in future it would be able to transform the entire log activities, such as applications used, memory devices accessed, keyboard events etc.

As denoted in Figure 1, DINT is currently further subdivided into DINT/Q and DINT/GB. The sub-module that interacts with IBN model is named DINT/Q, whist DINT/GB interacts with both Glass Box software and the Wolfie algorithm.

The primary responsibility of each sub-module is to provide seamless interaction between the evidence collection part (GB and Wolfie) and the inferential part of

DTB (the IBN Model). The components communicate via clearly defined and documented interfaces. This modular design facilitates maintenance and permits the components to be refined and extended independently of each other. The secondary responsibility is to automate the process of collecting evidence and its continuous input to the IBN model. This is a process that continuously runs in the background, monitoring each user's activities without interfering with her/his task, and providing and alert to the system administrator when a particular sequence of events triggers the alarm of a possible insider threat to the system.

DINT provides a smooth interaction among the different internal parts of the DTB system. We now address the need of extending this feature for the systems that will interact with the DTB.

4. Interoperability

Our concept is intended to deal with a community with many possible users, both inside the Intel community and outside it. Like most complex domains, the Intelligence community does not have a commonly accepted conceptualization of its rules, policies, or vocabulary, making any attempt to build an interoperable model very difficult. As a simple example, how would our model cope with cases in which different organizations have different names for the same concept? Also, agencies may have different security and access policies. As an example, in some agencies access to USB ports and floppy disks is permitted, while in others the use of such devices is a sure passport to indictment.

Our approach to these and similar issues is the use of Ontologies, a modeling technique that formalizes the semantics of the domain being modeled. By providing formal representations of semantics, ontologies provide a uniform way to communicate our vision and to adapt our technology to new organizations and concepts. As our ontology editor, we used the open source software Protégé (http://protege.stanford.edu/), which has been flexible enough for us to represent the entities, attributes and relationships characteristic of our domain.

In the initial part of the project, we kept our focus on building the behavioral model and on devising data mining algorithms capable of extracting the document relevance data that will feed that model, while also developing two ontologies in parallel. The first, the *Insider behavior ontology* (*IB*), describes the MEBN model of an insider threat behavior.

The second ontology, the *Organization and Task Ontology* (*OT*), is shown in figure 3 and portrays the various aspects of an internal organization. Among those we can cite its internal rules, details such as "need to know" policy, individual clearance, and compartment type (and its respective meanings with respect to data access), the data mining algorithms we use to capture document relevance,

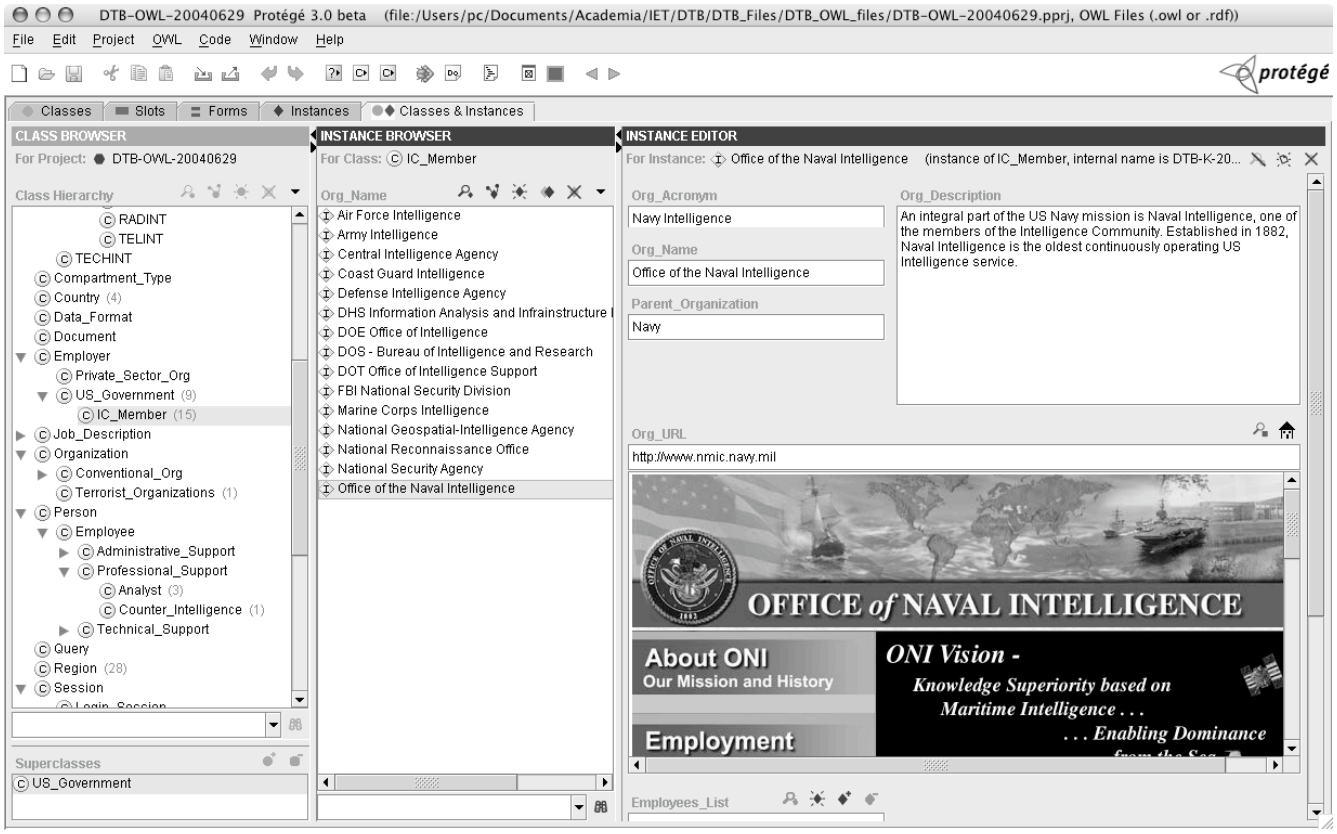

Figure 3 – The Organization and Task Ontology

and other particularities of the Intelligence domain.

It is important to note that both ontologies were made "in sync" with the developing phase of its subjects, which means they evolved in parallel with the building efforts of the MEBN model and the data mining algorithms. By doing so, during the modeling phase we were forced not only to think about the specific model details but also to enforce consistency of terminology throughout the model and the domain.

Once a smooth interaction between the internal components of the DTB system and its interoperability with the possible user environments was addressed, our next step is to verify whether the system meets its main goals.

5. Model Evaluation Strategy

A novel approach demands carefully designed evaluation. In our case, we started by exposing our behavioral model to domain specialists from outside our team, who suggested new aspects to be addressed and helped in fine-tuning our prior probabilities. After achieving a model that satisfied our external evaluators, we proceeded with simulation and sensitivity analysis experiments, in which we varied some of the parameters in order to analyze the overall robustness of our model.

Finally, a system designed to monitor human users should be evaluated with actual human users. For this purpose, we currently are conducting an experimental evaluation involving students from George Mason University's School of Information Technology and Engineering. Subjects perform research and analysis in an environment designed to mimic the target environment for our system. In addition to their overt task, some of the subjects are given a "clandestine" task involving a topic different from their assigned task. The purpose of the experiment is to evaluate how well our system can detect the subjects who are performing a clandestine task.

The results of both the computer and human subject experiments are expressed in terms of *probability of detection (PD)* and *probability of false alarms (PFA)*. The probability of detection is the probability of correctly detecting a threat behavior, while a false alarm happens when we declare a user to be a *Threat* while he or she is *Normal*. Table 1 shows the confusion matrix for the base case of our simulation study.

Detected Behavior	*Ground Truth*	
	Threat	**Normal**
Threat	True Positive (PD)	False Positive (PFA)
Normal	True Negative	False Negative

Table1: Confusion matrix for detecting insider user behavior

By varying the threshold, PD and PFA can be traded off against each other. A useful tool for evaluating classifiers is the receiver operating characteristic (ROC) curve, which plots PD against PFA. The area under the ROC curve (AUC) is a threshold-independent measure of the quality of a classifier. The ROC curves along with the AUC's are used as the basis for analyzing the results of the computational simulation, sensitivity analysis, and the live subjects' experiments.

6. Conclusion

Although standard access control methods provide some measure of control against insider abuse, more protection is required. The potentially disastrous consequences of even a single successful breach argue for the development of more sophisticated methods of detecting malicious insiders. Strictly enforced policies are highly efficient to prevent such criminals to operate, but experience proves that relying on this alone is just not enough.

We presented a novel approach to insider threat detection, which we believe has the potential to greatly increase the efficacy and efficiency of current systems. Uncovering improper human behavior along a series of events was an intractable approach that is now feasible due to the recent advances in the field of Bayesian inference technology. The main contribution of the DTB project is to transform those advances into reliable applications for the security arena.

Bibliography

Alghamdi, G., Laskey, K. B., Wang, X., Barbara, D., Shackleford, T., Wright, E. J. and Fitzgerald, J. 2004. Detecting Threatening Behavior Using Bayesian Networks. Conference on Behavioral Representation in Modeling and Simulation - BRIMS, Arlington, VA, May 17-20, 2004.

Anderson, K., Carzaniga, A., Heimbigner, D. and Wolf, A. 2004. Event-based Document Sensing for Insider Threats. Technical Report CU-CS-968-04. Department of Computer Science, University of Colorado. February, 6 2004. Boulder, CO: 24 pages.

Chakrabarty, S., van den Berg, M. and Dom, B. 1999. Focused crawling: a new approach to topic-specific Web resource discovery. Computer Networks 31(11-16) pp. 1623-1640.

CNN.com. 1998. Rationalizing Treason: An interview with Aldrich Ames. Cold War Experience - Espionage Series Retrieved January 20, 2005, from http://www.cnn.com/SPECIALS/cold.war/experience/spies/interviews/ames/.

CNN.com. 2001. The Case Against Robert Hanssen. from http://www.cnn.com/SPECIALS/2001/hanssen/.

Laskey, K. B. 2004, 2004/10/16. MEBN: Bayesian Logic for Open-World Reasoning. Retrieved Dec 8, 2004, from http://ite.gmu.edu/~klaskey/publications.html.