Procedia Computer Science

Volume 29, 2014, Pages 1299–1313

ICCS 2014. 14th International Conference on Computational Science

Static versus Dynamic Data Information Fusion analysis using DDDAS for Cyber Security Trust

Erik Blasch[1], Youssif Al-Nashif [2], Salim Hariri [2]

[1] Air Force Research Lab, Information Directorate

[2] NSF Center for Cloud and Autonomic Computing, The University of Arizona

erik.blasch.1@us.af.mil, alnashif@ece.arizona.edu, hariri@ece.arizona.edu

Abstract

Information fusion includes signals, features, and decision-level analysis over various types of data including imagery, text, and cyber security detection. With the maturity of data processing, the explosion of big data, and the need for user acceptance; the Dynamic Data-Driven Application System (DDDAS) philosophy fosters insights into the usability of information systems solutions. In this paper, we explore a notion of an adaptive adjustment of secure communication trust analysis that seeks a balance between standard static solutions versus dynamic-data driven updates. A use case is provided in determining trust for a cyber security scenario exploring comparisons of Bayesian versus evidential reasoning for dynamic security detection updates. Using the evidential reasoning proportional conflict redistribution (PCR) method, we demonstrate improved trust for dynamically changing detections of denial of service attacks.

1 Introduction

Information fusion (Blasch, *et al*., 2012) has a well-documented following of different methods, processes, and techniques emerging from control, probability, and communication theories. Information fusion systems designs require methods for big data analysis, secure communications, and support to end users. Current information fusion systems use probability, estimation, and signal processing. Extending theses techniques to operational needs requires an assessment of some of the fundamental assumptions such as secure communications over various data, applications, and systems. Specifically, the key focus of this paper is based on the question of measuring trust in static versus dynamic information fusion systems.

Static versus dynamic information fusion comes from three perspectives such as data, models, and processing. As related to information fusion techniques, many studies exist on centralized versus distributed processing, single versus multiple models, and stovepipe versus multi-modal data. In each case, static information fusion rests in centralized processing from single model estimation over a

Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2014

doi:10.1016/j.procs.2014.05.117

single source of data. On the other extreme is distributed processing, using multiple-models over multi-modal data; which in reality is supposed to cover the entire gamut of big data solutions captured in large-scale systems designs. In reality, with such an ambitious goal, there are always fundamental assumptions that tailor the system design to the user needs. For example, a system could be designed to capture all image data being collected from surveillance sensors; however filtering collections over a specific area, for a designated time internal, at a given frequency helps to refine answers to user requests. Thus, as a user selects the details of importance, responses should be accessible, complete, and trustworthy.

Dynamic information fusion is a key analysis of the paper of which we focus on trust. If a machine is processing all the data, then time and usability constraints cannot be satisfied. Thus, either the user or the machine must determine the appropriate set of data, models, and processing that is needed for a specific application. Trust analysis is required to determine security and reliability constraints, and DDDAS provides a fresh look at the balance between static and dynamic information fusion. In this paper, we explore the notions of dynamic information fusion towards decision making as cyber detections change.

In Section 2 we overview information fusion and DDDAS. Section 3 discusses the notions of trust as a means to balance between information fusion and dynamic data detections. Section 4 compares Bayesian versus evidential reasoning. Section 5 provides a use-case for analysis for cyber trust and Section 6 provides conclusions..

2 Information Fusion and DDDAS

Information fusion and DDDAS overlap in many areas such as data measurements, statistical reasoning, and software development for various applications. Recently, there is an interest in both communities to address big data, software structures, and user applications. The intersection of these areas includes methods of information management (Blasch, 2006) in assessing trust in data access, dynamic processing, and distribution for applications-based end users.

2.1 Information Fusion

The *Data Fusion Information Group* (DFIG) model, shown in Figure 1, provides the various attributes of an information fusion systems design. Information fusion concepts are divided between Low-level Information Fusion (LLIF) and High-level Information Fusion (HLIF) (Blasch, *et al.*, 2012). LLIF (L0-1) composes data registration (Level 0 [L0]) and explicit object assessment (L1) such as an aircraft location and identity (Yang, 2009). HLIF (L2-6) composes much of the open discussions in the last decade. The levels, to denote processing, include situation (L2) and impact (L3) assessment with resource (L4), user (L5) (Blasch, 2002), and mission (L6) refinement (Blasch, 2005). Here we focus on Level 5 fusion by addressing cybersecurity trust in systems design.

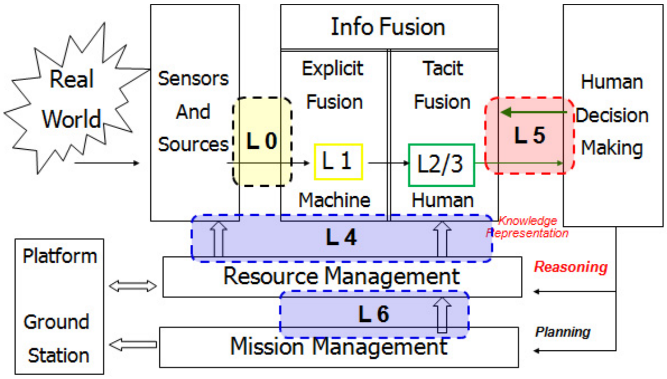

Figure 1. DFIG Information Fusion model (L = Information Fusion Level).

Data access for information fusion requires an information management (IM) model of the enterprise architecture, as shown in Figure 2. The IM model illustrates the coordination and flow of data through the enterprise with the various layers (Blasch, *et al*., 2012).

People or autonomous agents interact with the managed information enterprise environment by producing and consuming information. Various actors and their activities/services within an IM enterprise surround the IM model that transforms data into information. Within the IM model, there are various services that are needed to process the managed information objects (MIOs). Security is the first level of interaction between users and data.

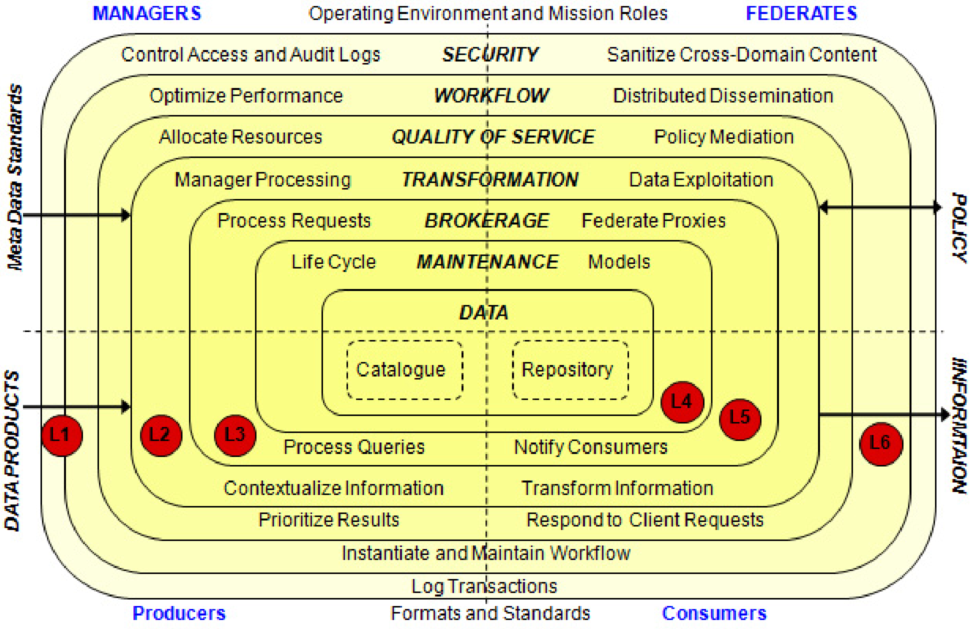

Figure 2. Information Management (IM) Model.

A set of service layers are defined that use artifacts to perform specific services. An artifact is a piece of information that is acted upon by a service or that influences the behavior of the service (e.g., a policy). The service layers defined by the model are: Security, Workflow, Quality of Service (QoS), Transformation, Brokerage, and Maintenance. These services are intelligent agents that utilize the information space within the architecture, such as cloud computing and machine analytics. Access to the data requires secure communications which is dynamic, data-type driven, and application specific.

2.2 Dynamic Data Driven Application Systems (DDDAS)

DDDAS is focused on applications modeling (scenarios), mathematical and statistical algorithms (theory), measurement systems, and systems software as shown in Figure 3. For a systems application, user mission needs drive data access over the scenarios. The available data is processed from measurements to information using theoretical principles. The data-driven results are presented to the user through visualizations; however the trust in the data is compounded by data quality, the model fidelity, and systems availability of which software is an integral part to a systems application.

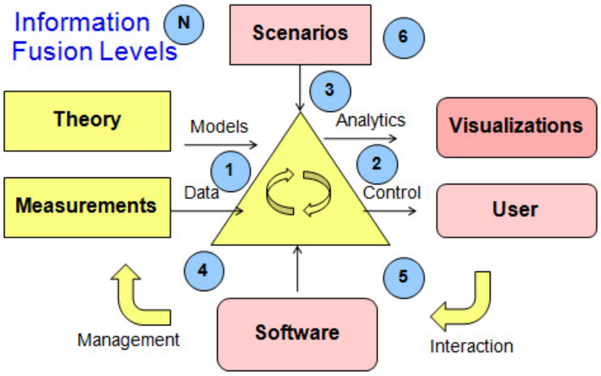

Figure 3. DDDAS Aligned with Information Fusion.

Using a cyber example for DDDAS, the application is secure data communications to meet mission needs (L6). While not a one-to-one mapping, it can be assumed that data management, driven by scenarios, identifies cyber threat attacks (L3) such as denial of service attacks. The theory and measurements come from the models of normal behavior (L1) which use computational methods to support cyber situation awareness (L2) visualization. The user (L5) interacts with the machine through data management (L4), as new measurements arrive. Current research seeks distributed, faster, and more reliable communication systems to enable such processing and coordination between the man and their machines, however, measurement of trust is paramount.

3 Trust in Information Processing

Several theories and working models of trust in automation have been proposed. Information which is presented for decision-aiding is not uniformly trusted and incorporated into situation awareness. Three proposed increasing levels, or 'stages of trust', for human-human interactions include: Predictability, Dependability, and Faith (Rempel, *et al.*, 1985). Participants progress through these stages over time in a relationship. The same was anticipated in human-automation interactions, either via training or experience. The main idea is that as trust develops, people will make decisions based upon the trust that the system will continue to behave in new situations as it has demonstrated in the past. Building upon Rempel's stages, (Muir & Moray, 1996) postulated that

$$\text{Trust} = \text{Predictability} + \text{Dependability} + \text{Faith} + \text{Competence} + \text{Responsibility} + \text{Reliability}$$

and further defined the construct of *Distrust*: which (1) can be caused by operator feeling that the automation is undependable, unreliable, unpredictable, etc. and a (2) set of dimensions related to automation failures, which may cause distrust in automated systems (location of failure, causes of failure or corruption, time patterns of failure).

Table 1, adapted below from (Muir & Moray, 1996), depicts the quadrant of trust and distrust behaviors with respect to good or poor quality of the automation. Basically, the outcome of a wrong decision to trust the automation is worse than the outcome of a wrong decision to not trust the automation. Hence, security is enforced to not trust a poor decision.

Operator’s trust & allocation of function	Quality of the automation	
	‘Good’	**‘Poor’**
Trusts and uses the automation	*Appropriate Trust* (optimize system performance)	*False Trust* (risk automated disaster)
Distrusts and rejects the automation	*False Distrust* (lose benefits of automation, inc. workload)	*Appropriate Distrust* (optimize system performance)

Table 1: Trust, Distrust, and Mistrust, (adapted from Muir and Moray, 1996)

Trust in the automation clearly impacts a user mental model of secure communications. Therefore, dynamic models must be devised to account for different levels of attention, trust, and interactions in Human in the Loop (HIL) and Human on the Loop (HOL) designs. A user must be given permission to refine the assessment for final decision for validity and reliability of the information presented. User Trust issues then are confidence (correct detection), security (impacts), integrity (what you know), dependability (timely), reliable (accurate), controllability, familiar (practice and training), and consistent (reliable).

Trust in information processing involves many issues; however, here we focus on the development of a cyber domain trust stack as shown in Figure 4. The trust stack composes policies, trust authority, collecting raw metrics and behavior analysis, leading to authentication and authorization, and then secure communications. Similar to the information management model, polices are important to determine whether data access is available. Likewise, sensor management gets access to raw metrics (Blasch, 2004) that need to be analyzed for situation awareness. The problem not being full addressed is the impeding results for secure communications. In what follows, we discuss the main functions to be provided by each layer in the trust stack shown in Figure 4.

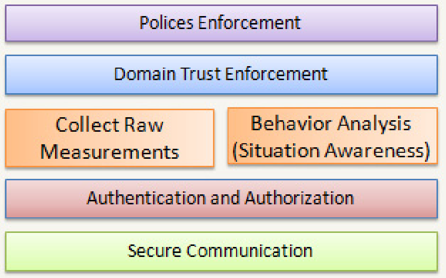

Figure 4. Trust Stack.

3.1 Secure Communications, Authentication, and Authorization

Secure communications is an important property to guarantee the confidentiality and integrity of the messages used to evaluate trust in the system. Certificates are used to verify the identify of communicating end-devices (Kaliski, 1993). The communication channel is encrypted using DES (Data Encryption Standard, 2010) in CFB64 (Cipher Feedback) mode. In this CFB mode, the first 8 bytes of the key generated used to encrypt the first block of data. This encrypted data is then used as a key for the second block. This process is repeated until the last block is encrypted. The DES is still used in legacy virtual private networks (VPNs) and could benefit from a DDDAS trust analysis even used with multiple protocol authentication systems such as Kerberos.

Multiple protocols have been developed over the years for password-based authentication, biometric authentication, and remote user authentication. In order to evaluate the trust of different entities with many users, multiple systems, and multiple domains, we assume the use of remote user authentication. Remote Authentication Dial-In User Service (RADIUS) (Willens, *et al*, 2000) is a famous client/server protocol to allow remote entities to communicate with a server to authenticate remote users. RADIUS gives organization ability to maintain user profiles in a specific database that the remote servers share.

The Domain Trust Enforcement (DTE) agent performs the authorization process for the end-to-end adaptive trust. Based on the results of the authentication process and the received trust level, the DTE agent grants or denies authorization to access the resources, i.e., allow or deny the communication between the different entities.

3.2 Collecting Raw Measurements

Much software, both commercial and open source, are available and provide important health and security information, such as Nagios (Nass, 2009). This information can be used to extract metrics that can be used to evaluate the trust of different entities. These metrics can be divided into multiple categories based on their source: User, Application, Machine, Connection, or Security Software Alerts. In order to evaluate the trust, the metrics need to be quantified and normalized (e.g., between 0 and 1) to a common scale. Table 2 shows a set of measured metrics and their quantification function and Figure 5 shows these categories with some example metrics.

Category	***Metric***	***Quantification***
User	Password Strength	$Q = \begin{cases} 0, \text{ Password Length<8} \\ 0.1 + 0.9 \cdot \dfrac{\text{Password Length}}{\text{Maximum Password Length}}, \text{ Otherwise} \end{cases}$
User	Days since last password change	$Q = \begin{cases} 0, \text{ \#days>Maximum Number Of Days} \\ 1 - \dfrac{\text{\#days}}{\text{Maximum Number of Days}}, \text{ Otherwise} \end{cases}$
User	Number of authentication failures	$Q = \begin{cases} 0, \text{ \#failures>Maximum Number Of Allowed Failures} \\ 1 - \dfrac{\text{\#failures}}{\text{Maximum Number Of Allowed Failures}}, \text{ Otherwise} \end{cases}$
User	Lock Outs	$Q = \begin{cases} 0, \text{ \#Lock Outs>Maximum Number Of Allowed Lock Outs} \\ 1 - \dfrac{\text{\#Lock Outs}}{\text{Maximum Number Of Allowed Lock Outs}}, \text{ Otherwise} \end{cases}$
Application	Developer Reputation	$Q = \dfrac{\text{Reputation}}{\text{Maximum Reputation}}$
Application	Who manages the software	$Q = \begin{cases} 1, \text{ Global Adminstrator} \\ 0.5, \text{ Local Administrator} \\ 0, \text{ No Administrator} \end{cases}$
Connection	Number of hops	$Q = \begin{cases} 0, \text{ \#Hops>Maximum Number Of Hops} \\ 1 - \dfrac{\text{\#Hops}}{\text{Maximum Number of Hops}}, \text{ Otherwise} \end{cases}$
Connection	Number of discarded Packets	$Q = \begin{cases} 0, \text{ \#Discarded Packet>Maximum \#Discarded Packet} \\ 1 - \dfrac{\text{\#Discarded Packet}}{\text{Maximum \#Discarded Packet}}, \text{ Otherwise} \end{cases}$
Machine	Firmware version	$Q = \begin{cases} 1, \text{ Up to date} \\ 0.5, \text{ 1 Version Behind} \\ 0, \text{ Otherwise} \end{cases}$

Machine	Shared Folders	$Q = \begin{cases} 1, \text{No Shared Folders} \\ 0.5, \text{Shared User Folders} \\ 0, \text{Shared System Folders} \end{cases}$
Analyzer	Integrity Check	$Q = \begin{cases} 1, \text{No Probelm} \\ 0.5, \text{Problem in user data} \\ 0, \text{Problem in system integrity} \end{cases}$
Analyzer	Virus Alerts	$Q = \begin{cases} 1, \text{No Alert} \\ 0.5, \text{Virus Found in a document} \\ 0.25, \text{Virus Found in an executable} \\ 0, \text{worm found} \end{cases}$

Table 2: Examples of metric quantification

3.3 Behavior Analysis

Behavior analysis techniques apply statistical and data mining techniques to determine the current operating zone of the execution environment (situation awareness) and also project its behavior in the near future. The operating point (OP) of an environment can be defined as a point in an n-dimensional space with respect to well-defined attributes. An acceptable operating zone can be defined by combining the normal operating values for each attribute. At runtime, the operating point moves from one zone to another and that point might move to a zone where the environment does not meet its trust and security requirements. We use these movements in the OP to adjust the trust value of the current environment as will be discussed in further detail in the Domain Trust Authority section. By continuously performing behavior analysis of the environment, we can then proactively predict and detect the anomalous behaviors that might have been caused by malicious attacks. Furthermore, once it is determined that the environment's operating point is moving outside the normal zone, it will adopt its trust value and then determine the appropriate proactive management techniques that can bring back the environment situation to a normal operating zone.

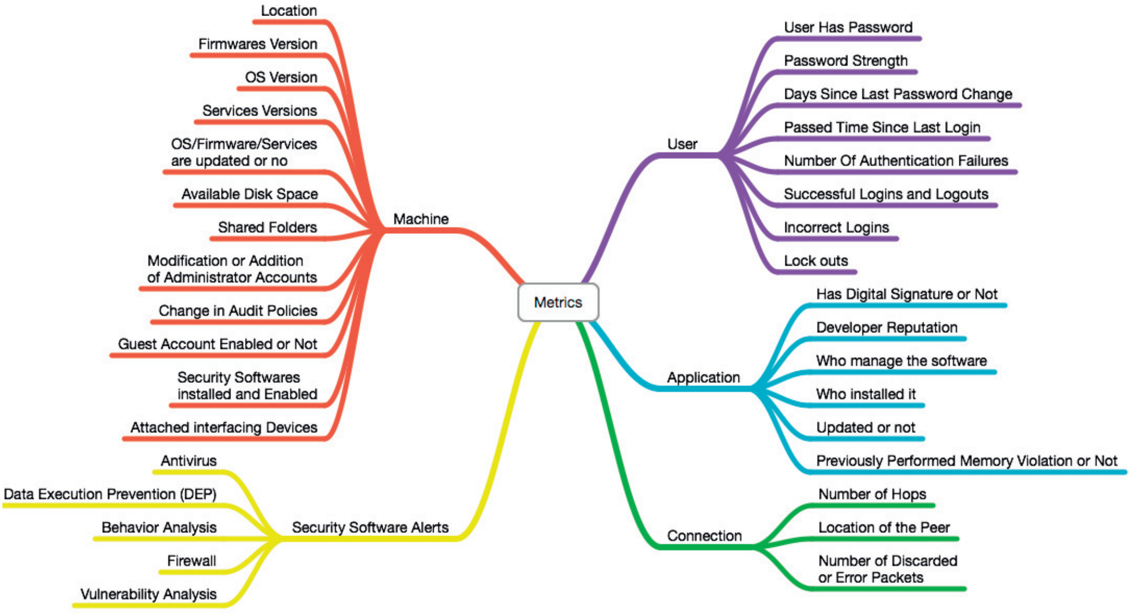
Metrics
Machine
Location
Firmwares Version
OS Version
Services Versions
OS/Firmware/Services are updated or no
Available Disk Space
Shared Folders
Modification or Addition of Administrator Accounts
Change in Audit Policies
Guest Account Enabled or Not
Security Softwares installed and Enabled
Attached interfacing Devices
Security Software Alerts
Antivirus
Data Execution Prevention (DEP)
Behavior Analysis
Firewall
Vulnerability Analysis
User
User Has Password
Password Strength
Days Since Last Password Change
Passed Time Since Last Login
Number Of Authentication Failures
Successful Logins and Logouts
Incorrect Logins
Lock outs
Application
Has Digital Signature or Not
Developer Reputation
Who manage the software
Who installed it
Updated or not
Previously Performed Memory Violation or Not
Connection
Number of Hops
Location of the Peer
Number of Discarded or Error Packets

Figure 5. Trust Metrics.

3.4 Domain Trust Authority

DTA evaluates the end-to-end trust over secure communications. It defines a tuple (machine, application, user, data) to be an entity and all communications among entities has a certain context. Thus authentication is conducted per entity. Every entity has a trust level associated with it. In order to measure the trust, trust's metrics are introduced, and they take values between 0 and 1. Where 0 represents the distrust and 1 represent the blind or full trust. The trust measurements for all entities are stored in an entity call Trust Authority. The NIST standard SP 800-53 (NIST, 2010) is used and it defines four levels of trust:

Level	Distrust	Low Trust	Moderate	High Trust
Trust Value	0.00	0.33	0.66	1.00

Initially, a risk and impact analysis is performed to quantify the impact of each component on the overall operations of the network. Common Vulnerabilities and Exposures (CVE) and Common Vulnerability Scoring System (CVSS) are used to evaluate the initial impact for both software and the environment, and reputations of the users are used to assign their initial impacts. Based on the initial impact analysis, the initial trust values for each entity is determined. The risk and impact analysis performed is in consistence with the NIST "Recommended Security Controls for Federal Information Systems and Organizations" report. According to the NIST report, risk measures the extent to which entities are threatened by circumstances or events. The risk is a function of impact and its probability of occurrence. Risks arise from the loss of confidentiality, integrity, and/or availability of information and resources. Thus the initial trust *T* can be viewed as an inverse function of the risk *R*:

$$T = 1 / R \qquad (1)$$

Where the risk of an entity *i* is a function of the impact *imp*:

$$R_{\mathrm{i}} = imp_{\mathrm{i}}\,(\text{confidentiality}) \bullet Pr\; imp_{\mathrm{i}}\,(\text{confidentiality}) + imp_{\mathrm{i}}\,(\text{integrity}) \bullet Pr\; imp_{\mathrm{i}}\,(\text{integrity}) + imp_{\mathrm{i}}\,(\text{availability}) \bullet Pr\; imp_{\mathrm{i}}\,(\text{availability}) \qquad (2)$$

When a new entity is added, it has to register with the Mutual Authentication (MA) module and then its initial trust value can be quantified according to Equations 1 and 2.

Verify Trust

When an entity communicates with another entity, an Autonomic Trust Management (ATM) agent obtains the trust level of the entity that needs to interact with from the Trust Authority (TA), see Figure 6. If the trust level of the remote entity is below the minimum required trust level set in the policies, then the communication is dropped. By continuously checking with TA module, any interacting entities will not be able to communicate if they do not meet the end-to-end trust policies. Once the component trust level is verified, they can proceed and interact securely using the secure communications.

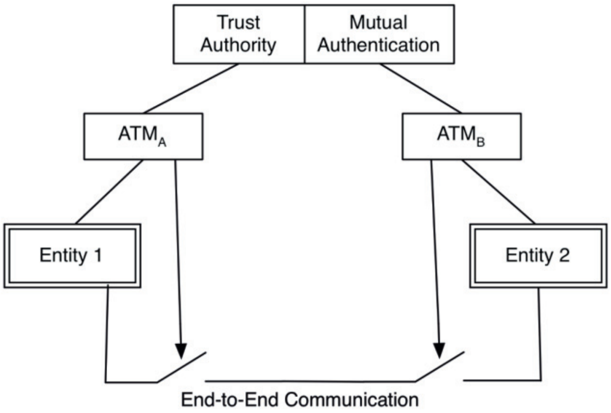

Figure 6. Adaptive End-to-End Trust

Adaptive Trust

The trust value assigned to each component is not static and is updated continuously. The Trust Authority module is the one responsible for re-evaluating the trust at runtime. As mentioned in the previous section, the trust is measured per entity and the trust levels are between 0 and 1.

$$T(E) \in [0,\ 1] \tag{3}$$

Each interaction between entities is governed by a context C. Thus, trust level for entities is computed per context:

$$T(E,\ C) \in [0,\ 1] \tag{4}$$

A Forgiveness Factor, F, is assigned to provide an adaptive mechanism for compromised entities to start gaining trust after all existing vulnerabilities have been fixed. Based on the impact of the entity on the overall operations, we can control the time it takes for that entity to recover its trust level. Monitoring, measuring, and quantifying trust metrics are required, and they are performed by the ATM. M_i will denote the collected trust metric, where i is the metric identifier. The function $m_i()$ is a quantifying function that returns a measurement between 0 and 1 for the metric M_i.

The overall trust for an entity is computed using two types of trust: 1) self-measured trust and 2) reputation-measured trust. The self-measured trust T_s is the trust that is evaluated based on the measurement performed by the ATM agent that manages the entity. While the reputation-measured trust, T_p is based on the trust metrics collected from peers based on a previous recent interaction with the entity for which the trust is being re-evaluated. The T_s and T_p are given by following equations:

$$T_{\mathrm{S}}(E,C) = T(ATM_{\mathrm{E}},C)\cdot\sum_{i=1}^{\mathrm{L}} I_{\mathrm{i}}(C)\cdot m_{\mathrm{i}}(M_{\mathrm{i}})$$

$$T_{\mathrm{P}}(E,C) = \frac{1}{K}\sum_{j=1}^{\mathrm{K}} T(ATM_{\mathrm{j}},C)\cdot\sum_{i=1}^{\mathrm{L}} I_{\mathrm{i}}(C)\cdot m_{\mathrm{i}}(M_{\mathrm{i}}) \tag{5}$$

The values of the metric weight I_i for metric i is determined based on the feature selection technique, where:

$$\sum_{i=1}^{\mathrm{L}} I_{\mathrm{i}}(C) = 1 \tag{6}$$

Based on the context and the type of operations, the end-to-end trust is evaluated using three trust evaluation strategies: Optimistic, Pessimistic, and Average. The end-to-end trust for each strategy can be evaluated as follows:

Trust Confidence	**Trust Evaluation Strategy**
Optimistic	$T(E, C) = max\ \{T_S\ (E, C), T_P\ (E, C)\}$
Average	$T(E, C) = ave\ \{T_S\ (E, C), T_P\ (E, C)\}$
Pessimistic	$T(E, C) = min\ \{T_S\ (E, C), T_P\ (E, C)\}$

Once $T(E,C)$ is computed, then it is mapped to the nearest of trust level: (High, Moderate, Low, and None).

The Trust Authority module continuously evaluates the trust for all components and their entities whenever new metrics are obtained from the ATM agents that require an update to entity trust evaluation above depending on the trust evaluation strategy. Various reasoning evaluation strategies exist, such as that of Bayesian, Evidential Reasoning, and Belief Functions (Blasch, *et al*, 2013), that can be used to evaluate trust.

In a DDDAS cyber environment, there are many levels of information fusion, but to build a trustworthy DDDAS environment, we need to check the trust of each level of information fusion. The Domain Trust Authority is the place to verify the trust of each entity passing information within the DDDAS environment. When the trust level drops below certain threshold; the incoming data can be dropped to enable secure communications. What follows are the DDDAS theory, simulations, measurements, and software analysis for Information fusion levels of cyber data, situation/behavior assessment, information management, and user refinement.

3.5 Bayes versus Evidential Reasoning

A fundamental technique for data fusion is Bayes Rule. Recently, (Dezert, *et al*., 2012) has shown that Dempster's rule is consistent with probability calculus and Bayesian reasoning if and only if the prior $P(X)$ is uniform. However, when the $P(X)$ is not uniform, then Dempster's rule gives a different result. Both (Yen, 1986) and (Mahler, 1996) developed methods to account for non-uniform priors. Others have also tried to compare Bayes and evidential reasoning (ER) methods (Mahler, 2005, Blasch, *et al*., 2013). Assuming that we have multiple measurements $Z = \{Z_1, Z_2, \ldots, Z_N\}$ for cyber detection D being monitored, Bayesian and ER methods are developed next.

3.6 Relating Bayes to Evidential Reasoning

Assuming conditional independence, one has the Bayes method:

$$\mathrm{P}(X \mid Z_1 \cap Z_2) = \frac{\mathrm{P}(X \mid Z_1)\,\mathrm{P}(X \mid Z_2) \,/\, \mathrm{P}(X)}{\sum\limits_{i=1}^{N} \mathrm{P}(X_i \mid Z_1)\,\mathrm{P}(X_i \mid Z_2) \,/\, \mathrm{P}(X_i)} \tag{7}$$

With no information from Z_1 or Z_2, then $\mathrm{P}(X \mid Z_1, Z_2) = \mathrm{P}(X)$. Without Z_2, then $\mathrm{P}(X \mid Z_1, Z_2) = \mathrm{P}(X \mid Z_1)$ and without Z_1, then $\mathrm{P}(X \mid Z_1, Z_2) = \mathrm{P}(X \mid Z_2)$. Using Dezert's formulation, then the denominator can be expressed as a normalization coefficient:

$$m_{12}(\varnothing) = 1 - \sum_{X_i\,;\,X_j \mid X_i \cap X_j} \mathrm{P}(X_i \mid Z_1)\,\mathrm{P}(X_i \mid Z_2) \tag{8}$$

Using this relation, then the total probability mass of the conflicting information is

$$P(X \mid Z_1 \cap Z_2) = \frac{1}{1 - m_{12}(\varnothing)} \bullet P(X \mid Z_1)\, P(X \mid Z_2) \tag{9}$$

which corresponds to Dempster's rule of combination using Bayesian belief masses with uniform priors. When the prior's are not uniform, then Dempster's rule is not consistent with Bayes' Rule. For example, let $m_0(X) = P(X)$, $m_1(X) = P(X \mid Z_1)$, and $m_2(X) = P(X \mid Z_2)$, then

$$m(X) = \frac{m_0(X)\ m_1(X)\ m_2(X)}{1 - m_{012}(\varnothing)} = \frac{P(X)\ P(X \mid Z_1)\ P(X \mid Z_2)}{\sum_{i=1}^{N} P(X_i)\ P(X_i \mid Z_1)\ P(X_i \mid Z_2)} \tag{10}$$

Thus, methods are needed to deal with non-uniform priors and appropriately redistribute the conflicting masses.

3.7 Proportional Conflict Redistribution

Recent advances in DS methods include *Dezert-Smarandache Theory* (DSmT). DSmT is an extension to the Dempster-Shafer method of evidential reasoning which has been detailed in numerous papers and texts: *Advances and applications of DSmT for information fusion (Collected works)*, Vols. 1-3 (Dezert, *et al.*, 2009). In (Dezert, *et al.*, 2002) introduced the methods for the reasoning and in presented the hyper power-set notation for DSmT (Dezert, *et al.*, 2003). Recent applications include the DSmT Proportional Conflict Redistribution rule 5 (PCR5) applied to target tracking (Blasch, 2013).

The key contributions of DSmT are the redistributions of masses such that no refinement of the frame Θ is possible unless a series of constraints are known. For example, Shafer's model (Shafer, 1976) is the most constrained DSm hybrid model in DSmT. Since Shafer's model, authors have continued to refine the method to more precisely address the combination of conflicting beliefs (Josang, *et al.*, 2006) and generalization of the combination rules (Smaradache, *et al.*, 2005, Daniel, 2006). An adaptive combination rule (Florea, *et al.*, 2006) and rules for quantitative and qualitative combinations (Martin, 2008) have been proposed. Recent examples for sensor applications include electronic support measures, (Djiknavorian, *et al.*, 2010), physiological monitoring sensors (Lee, *et al.*, 2010), and seismic-acoustic sensing (Blasch, *et al.*, 2011).

Here we use the *Proportional Conflict Redistribution* rule no. 5 (PCR5)*. We replace Smets' rule (Smets, 2005) by the more effective PCR5 to cyber detection probabilities. All details, justifications with examples on PCR*n* fusion rules and DSm transformations can be found in the DSmT compiled texts (Dezert, *et al.*, 2009 Vols. 2 & 3). A comparison of the methods is shown in Figure 7.

* Note: PCR used here is from information fusion technology and not the a Platform Configuration Register (PCR) of the Trusted Platform Module (TPM) hardware technology.

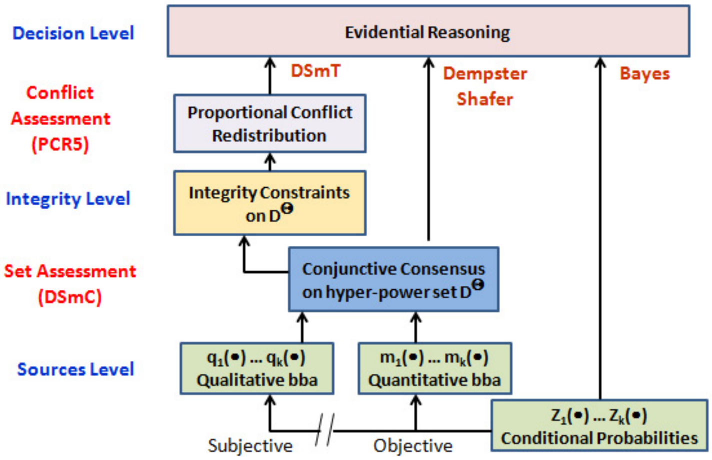
Decision Level
Evidential Reasoning
DSmT
Dempster Shafer
Bayes
Conflict Assessment (PCR5)
Proportional Conflict Redistribution
Integrity Level
Integrity Constraints on D^Θ
Set Assessment (DSmC)
Conjunctive Consensus on hyper-power set D^Θ
Sources Level
q1(•) ... qk(•) Qualitative bba
m1(•) ... mk(•) Quantitative bba
Z1(•) ... Zk(•) Conditional Probabilities
Subjective
Objective

Figure 7. Comparison of Bayesian, Dempster-Shafer, and PCR5 Fusion Theories

In the DSmT framework, the PCR5 is used generally to combine the basic belief assignment (bba)'s. PCR5 transfers the conflicting mass only to the elements involved in the conflict and proportionally to their individual masses, so that the specificity of the information is entirely preserved in this fusion process. Let $m_1(.)$ and $m_2(.)$ be two independent bba's, then the PCR5 rule is defined as follows (see Dezert, *et al*., 2009, Vol. 2 for full justification and examples): $m_{PCR5}(\varnothing) = 0$ and $\forall X \in 2^\Theta \setminus \{\varnothing\}$, where $\varnothing$ is the null set and 2^Θ is the power set:

$$m_{PCR5}(X) = \sum_{\substack{X_1;\, X_2 \in 2^\Theta \\ X_1 \cap X_2 = X}} m_1(X_1) + m_2(X_2) + \sum_{\substack{X_2 \in 2^\Theta \\ X_2 \cap X = \varnothing}} \left[\frac{m_1(X_1)^2\, m_2(X_2)}{m_1(X_1) + m_2(X_2)} + \frac{m_1(X_1)\, m_2(X_2)^2}{m_1(X_1) + m_2(X_2)} \right] \quad \text{`(11)}$$

where $\cap$ is the interesting and all denominators in the equation above are different from zero. If a denominator is zero, that fraction is discarded. Additional properties and extensions of PCR5 for combining qualitative bba's can be found in (Dezert, 2009, Vol. 2 & 3) with examples and results. All propositions/sets are in a canonical form.

3.8 Example of DDDAS Cyber Trust Analysis

In this example, we assume that policies are accepted and that the trust stack must determine whether the dynamic data is trustworthy. The application system collects raw measurements on the data intrusion (such as denial of service attacks) and situation awareness is needed. Conventional information fusion processing would include Bayesian analysis to determine the state of the attack. However, here we use the PCR5 rule which distributes the conflicting information over the partial states. Figure 8 shows the results for a normal system being attacked and the different methods (Bayes, DS, and PCR5) to access the dynamic attack. Trust is then determined with percent improvement in analysis. Since the cyber classification of attack versus no attack is not consistent, there is some conflict in the processing of the measurement data going from an measurements of attack and vice versa. The constant changing of measurements requires acknowledgment of the change and data conflict as measured using the PCR5 method.

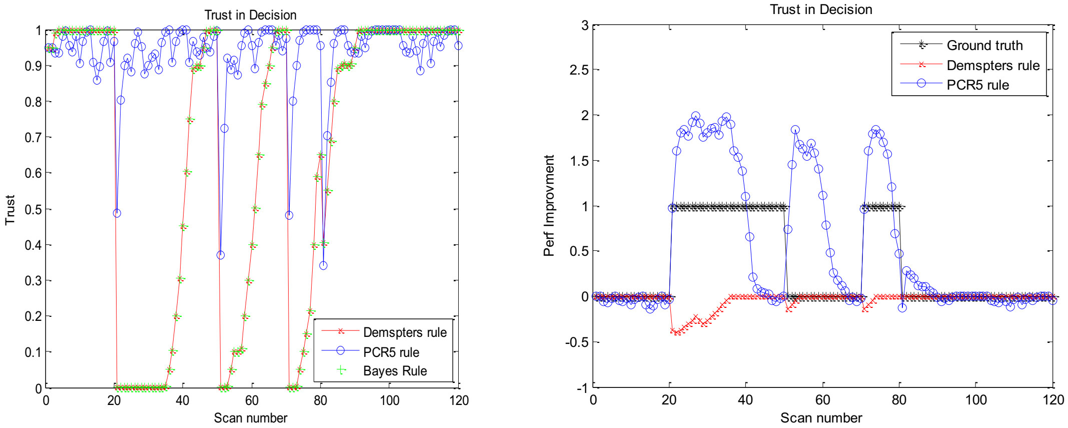

Figure 8. Results of Bayesian, Dempster-Shafer, and PCR5 Fusion Theories for trust.

The improvement of PCR5 over Bayes is shown in Figure 8 and compared with the modest improvement from DS. The average performance improvement of PCR5 is 46% and DS is 2%, which is data and application dependent. When comparing the results, it can be seen that when a system goes from a normal to an attack state, PCR5 responds quicker in analyzing the attack, resulting in maintaining trust in the decision. Such issues of data reliability, statistical credibility, and application survivability all contribute to the presentation of information to an application-based user. While the analysis is based on behavioral situation awareness, it is understood that polices and secure communications can leverage this information for domain trust analysis and authentication and authorization that can map measurements to software requirements.

3.9 Policies Enforcement

Policies are an important component of cyber trust (Blasch, 2012) as shown in Figure 9. As an example, a policy is administered for retrieval of information. Policy information determines the attributes for decisions. Determining the decision leads to enforcement. Such a decision is based on trust processing from which effective enforcement can support secure communications.

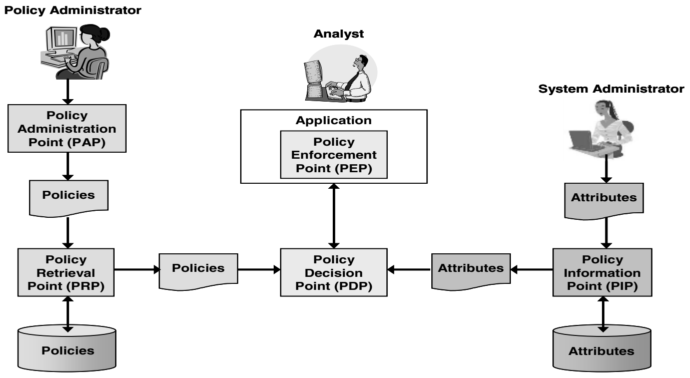
Policy Administrator
Analyst
System Administrator
Policy Administration Point (PAP)
Application
Policy Enforcement Point (PEP)
Policies
Attributes
Policy Retrieval Point (PRP)
Policies
Policy Decision Point (PDP)
Attributes
Policy Information Point (PIP)
Policies
Attributes

Figure 9. Policy-Based Fusion of Information requiring Trust (Blasch, 2012)

There are many possible information fusion strategies to enable data access from policies. Here we demonstrate an analysis of Bayesian versus evidential reasoning for determining cyber situation awareness trust. Future work includes threat intent (Shen, *et al*., 2009), impact assessment (Shen, *et al*., 2007), transition behaviors (Du, *et al*., 2011) and developing advanced forensics analysis (Yu, *et al*., 2013).

4 Conclusions

Information fusion (IF) and Dynamic Data-Driven Application Systems (DDDAS) are emerging techniques to deal with big data, multiple models, and decision making. One topic of interest to both fields of study is a measure of trust. In this paper, we explored a system for cyber security fusion which addresses system-level application issues of model building, data analysis, and polices for application trust. IF and data-driven applications utilize a common framework of probability analysis and here we explored a novel technique of PCR5 that builds on Bayesian and Dempster-Shafer theory to determine trust. Future research would include real world data, complete analysis of the trust stack, and sensitivity of models/measurements in secure cyber situation awareness trust analysis.

Acknowledgements

This work is partially supported by AFOSR DDDAS award number FA95550-12-1-0241, and National Science Foundation research projects NSF IIP-0758579, NCS-0855087 and IIP-1127873.

References

Blasch, E., Plano, S. (2002) "JDL Level 5 Fusion model 'user refinement' issues and applications in group Tracking," *Proc. SPIE,* Vol. 4729.

Blasch, E., Plano, S. (2003) "Level 5: User Refinement to aid the Fusion Process," *Proc. of SPIE,* 5099, 2003.

Blasch, E., Pribilski, M., Daughtery, B., Roscoe, B., and Gunsett, J. (2004) "Fusion Metrics for Dynamic Situation Analysis," *Proc. of SPIE,* Vol. 5429.

Blasch, E., Plano, S. (2005) "DFIG Level 5 (User Refinement) issues supporting Situational Assessment Reasoning," *Int. Conf. on Info Fusion.*

Blasch, E. (2006) "Level 5 (User Refinement) issues supporting Information Fusion Management," *Int. Conf. on Info Fusion.*

Blasch, E., Kadar, I., Salerno, J., Kokar, M. M., Das, S., Powell, *et al.*. (2006) "Issues and Challenges in Situation Assessment (Level 2 Fusion)," *J. of Advances in Information Fusion,* Vol. 1, No. 2, pp. 122 - 139, Dec.

Blasch, E., Dezert, J., Valin, P. (2011) "DSMT Applied to Seismic and Acoustic Sensor Fusion," *Proc. IEEE Nat. Aerospace Electronics Conf (NAECON).*

Blasch, E., Bosse, E., Lambert, D. A. (2012), *High-Level Information Fusion Management and Systems Design*, Artech House, Norwood, MA.

Blasch, E., Dezert, J., Pannetier, B. (2013) "Overview of Dempster-Shafer and Belief Function Tracking Methods," *Proc. SPIE*, Vol. 8745,

Blasch, E., Steinberg, A., Das, S., Llinas, J., Chong, C.-Y., Kessler, O., Waltz, E., White, F., (2013) "Revisiting the JDL model for information Exploitation," *Int'l Conf. on Info Fusion*.

Blasch, E. (2013) "Enhanced Air Operations Using JView for an Air-Ground Fused Situation Awareness UDOP," *AIAA/IEEE Digital Avionics Systems Conference*, Oct..

Chen, G., Shen, D., Kwan, C., Cruz, J., *et al.*, (2007) "Game Theoretic Approach to Threat Prediction and Situation Awareness," *Journal of Advances in Information Fusion,* Vol. 2, No. 1, 1-14, June.

Culbertson, J., and Sturtz, K., (2013) "A Categorical Foundation for Bayesian Probability," *Applied Categorical Structures*.

Daniel, M., (2006) "Generalization of the Classic Combination Rules to DSm Hyper-Power Sets," *Information & Security, An Int'l J.*, Vol. 20.

Data Encryption Standard (2010), http://blog.fpmurphy.com/2010/04/openssl-des-api.html

Dezert, J. (2002) "Foundations for a new theory of plausible and paradoxical reasoning," *Information & Security, An Int'l J.*, ed. by Prof. Tzv. Semerdjiev, Vol. 9.

Dezert, J. Smarandache, F. (2003) "On the generation of hyper-powersets for the DSmT," *Int. Conf. on Info Fusion*.

Dezert, J. Smarandache, F., (2009) *Advances and applications of DSmT for information fusion (Collected works)*, Vols. 1-3, American Research Press, http://www.gallup.unm.edu/~smarandache/DSmT.htm

Dezert, J. (2012) "Non-Bayesian Reasoning for Information Fusion – A Tribute to Lofti Zadeh," *submitted to J. of Adv. of Information Fusion.*

Djiknavorian, P., Grenier, D., Valin, P. (2010) "Approximation in DSm theory for fusing ESM reports," *Int. Workshop on Belief functions* 2010, April.

Du, H. Yang, S. J. (2011) "Characterizing Transition Behaviors in Internet Attack Sequences," in *IEEE ICCCN'11.*

Dsouza, G., Rodriguez, G., Al-Nashif, Y., Hariri, S. (2013) "Resilient Dynamic Data Driven Application Systems (rDDDAS)," *International Conference on Computational Science.*

Dsouza, G., Hariri, S., Al-Nashif, Y., Rodriguez, G. (2013) "Building resilient cloud services using DDDAS and moving target defense," *Int. J. Cloud Computing..*

Florea, M. C., Dezert, J., Valin, P., Smarandache, F., Jousselme, A-L., (2006) "Adaptive combination rule and proportional conflict redistribution rule for information fusion," *COGIS '06 Conf.*,

Josang, A., Daniel, M. (2006) "Strategies for Combining Conflict Dogmatic Beliefs," *Int. Conf. on Info Fusion.*

Kaliski, B. (1993) "A Survey of Encryption Standards," *IEEE Micro*, Issue, 6, December.

Lee, Z. H., Choir, J. S., Elmasri, R. (2010). "A Static Evidential Network for Context Reasoning in Home-Based Care," *IEEE Trans. Sys., Man, and Cyber-Part A; Sys & Humans*, Vol. 40, No. 6, Nov.

Mahler, R.P. (1996) "Combining ambiguous evidence with respect to ambiguous a priori knowledge, I: Boolean logic," *IEEE Trans. Sys., Man & Cyber., Part A*, Vol. 26, pp. 27–41.

Mahler, R., (2005) "Can the Bayesian and Dempster-Shafer approaches be reconciled? Yes," *Int'l Conf. on Information Fusion.*

Martin, A., Osswald, C., Dezert, J., Smarandache, F. (2008) "General Combination Rules for Qualitative and Quantitative Beliefs," *J. of Advances in Information Fusion*, Vol. 3, No. 2, Dec.

Muir, B. and Moray, N. (1996) "Trust in automation: Part II. Experimental studies of trust and human Intervention in a process control simulation," *Ergonomics*, 39 (3), 429-460.

Nass, S. J., Levit, L. A., Gostin, L. O. (2009). Beyond the HIPAA Privacy Rule: Enhancing Privacy, Improving Health Through Research, National Academies Press.

NIST, (revision, 2010) "Recommended Security Controls for Federal Information Systems and Organizations," NIST Special Publication 800-53, Revision 3.

Rempel, J. K., Holmes, J. G., and Zanna, M. P. (1985) "Trust in Close Relationships," *Journal of Personality and Social Psychology*, 49 (1), 95-112.

Shafer, G. (1976) *A Mathematical Theory of Evidence*, Princeton, NJ: Princeton Univ. Press.

Shen, D., Chen, G. et al., (2007) "Strategies Comparison for Game Theoretic Cyber Situational Awareness and Impact Assessment," *Int. Conf. on Info Fusion..*

Shen, D., Chen, G., *et al.* (2009) "An Adaptive Markov Game Model for Cyber Threat Intent Inference", invited Ch. 21 in *Theory and Novel Applications of Machine Learning*, M. J. Er and Y. Zhou. (Eds.), IN-TECH.

Smaradache, F., Dezert, J., (2005) "Information fusion based on new proportional conflict redistribution rules," *Int. Conf. Inf. Fusion.*

Smets, P., (2005) "Analyzing the Combination of Conflicting Belief Functions," *Int. Conf. on Info Fusion.*

Willens, S., *et al*, (2000) Remote Authentication Dial-In User Service (RADIUS), accessed at http://tools.ietf.org/search/rfc2865

Yang, C., Blasch, E. (2009) "Kalman Filtering with Nonlinear State Constraints," *IEEE Trans. Aerospace and Electronic Systems*, Vol. 45, No. 1, 70-84, Jan.

Yen, J. (1986) "A reasoning model based on the extended Dempster Shafer theory," *Nat Conf. on Artificial Intelligence.*

Yu, W., Fu, X., *et al.* (2013) "On Effectiveness of Hopping-Based Techniques for Network Forensic Traceback," *Int'l J. of Networked and Distributed Computing*, Vol. 1, No. 3, 2013.