2013 5th International Conference on Cyber Conflict
K. Podins, J. Stinissen, M. Maybaum (Eds.)

Information Sharing Models for Cooperative Cyber Defence

Jorge L. Hernandez-Ardieta
Cybersecurity Unit, Indra
Madrid, Spain
jlhardieta@indra.es

Juan E. Tapiador
COSEC Lab, Dept. of Computer Science
Universidad Carlos III de Madrid
Leganes, Madrid, Spain
jestevez@inf.uc3m.es

Guillermo Suarez-Tangil
COSEC Lab, Dept. of Computer Science
Universidad Carlos III de Madrid
Leganes, Madrid, Spain
guillermo.suarez.tangil@uc3m.es

Abstract: The globalisation and increasing complexity of modern cyber security operations have made it virtually impossible for any organisation to properly manage cyber threats and cyber incidents without leveraging various collaboration instruments with different partners and allies. This is especially relevant in certain areas of national security, like the protection of critical infrastructures, where the partnership amongst public and private sectors is paramount to adequately protect those infrastructures from emerging threats.

Over the last years consensus has emerged that sharing information about threats, actors, tactics and other cyber security information will play a central role in deploying an effective cooperative cyber defence. Near real-time information sharing has recently gained momentum as a means to redress the imbalance between defenders and attackers. In practical terms, the majority of current efforts in this area revolve around the idea of developing infrastructures and mechanisms that facilitate information sharing, notably through standardization of data formats and exchange protocols. While developing and deploying such an infrastructure is certainly essential to solve the problem of "how" to effectively share information, we believe that some key aspects still remain unaddressed, namely those related to deciding on "what" to share, "with whom", "when", as well as reasoning about the repercussions of sharing sensitive data.

In this paper, we argue that effective policies for near real-time information sharing must rely on, at least, two pillars. First, formal models to estimate the subjective value of the information shared should be developed. Second, trust/reputation models that consider the dynamic behaviour and changing factors of the sharing community have to be identified. For the latter, we propose to model information sharing communities as directed graphs, with nodes representing community members and edges modelling sharing relationships among them. Relevant properties of both nodes and edges are captured through attributes attached to each of them, which subsequently facilitate reasoning about particular data exchanges.

Keywords: *Cyber security, Cyber defence, Information sharing, Cooperation*

1. INTRODUCTION

Cyber conflicts are intensifying at a steady pace, both in prevalence, complexity and potential impact on individual organisations, nations, and the society at large. Besides, they have largely gone global, and globalisation has brought about a number of complications to cyber defence operations. On the one hand, interdependences amongst networks and information systems make localised and uncoordinated countermeasures rather ineffective, as they cannot ensure that no weak links are left in the chain. On the other hand, the attack landscape has evolved considerably in the last years, with a substantial rise in attacks involving a large number of distributed entities (e.g., botnets and DDoS) [1]; the emergence of markets where zero-day vulnerabilities are bought and sold on a regular basis [2]; or the advent of remarkably complex pieces of malware and cyber weapons [3][4][5], to name just a few. One major consequence of this new state of affairs in cyber security is a serious imbalance between the capabilities of attackers and defenders. As a matter of fact, at the moment it is virtually impossible for any organisation to prepare for and respond to cyber incidents without leveraging various collaboration instruments with other partners and allies. Examples abound in some areas of national security, such as the protection of critical infrastructures, where partnerships amongst public and private sectors are paramount to adequately mitigate risks and manage cyber attacks.

Over the last years consensus has emerged that sharing information about threats, actors, tactics and other cyber security information will be key to succeed in cyber defence. This sentiment has certainly not emerged from one day to the next, as proved, for example, by the efforts conducted over the last decade or so to categorise cyber security information, standardise data formats and exchange protocols, and develop infrastructures and mechanisms that facilitate sharing (see, e.g., [6] or the Cyber Defense Data Exchange and Collaboration Infrastructure (CDXI) being built by NATO [7]). While this is clearly essential to solve the problem of *how* to effectively share, some other relevant dimensions of the problem have received far less attention, notably those related to deciding on *what* to share, *with whom*, *when*, as well as reasoning about and adapting to the *repercussions* of sharing. One plausible cause for this is the fact that cyber security information sharing has largely been –and still is– a human-driven activity, where decisions are made one at a time and, in many cases, without an explicit elucidation of the rationale that motivates the decision. We believe, however, that addressing most of these questions will eventually become vital, particularly for scenarios where prompt responses to cyber threats are mandatory and, therefore, sharing decisions need to be made on a policy basis, in near real time, and with very little human involvement.

In this paper, we argue that the problem of sharing cyber security information can

be reformulated as one of *risk-based decision-making*. Thus, we seek procedures to answer questions such as: what are the benefits and the risks of sharing right now this piece of information with such party? Our choosing of this approach is motivated by two main facts:

a. On the one hand, taking an algorithmic approach on sharing will force us to quantify factors such as risks (and, implicitly, the value of information) and trust on sources and recipients. Even though these are challenging issues, a body of work in other contexts is slowly emerging. We believe that the cyber security community should adapt and adopt some of these techniques, particularly in scenarios where there is a need-to-share but the risks of doing so are not properly managed.

b. On the other hand, policies for information sharing must be elucidated and formally analysed. But policy making is a complex issue, and a given set of rules might well have unforeseen consequences, hence the need for automated techniques that provide optimal responses.

However, the ability to automatically making sharing decisions requires reasoning over formal structures (models) of most of the relevant elements involved, including the information itself, its value, the risks associated with disclosure (not only by us, but afterwards by partners receiving the information, either inadvertently or on purpose), our perception of the sharing community and the relationships among partners, etc.

In the remaining of this paper, we attempt to elucidate some of these questions, discuss challenges and identify areas where more efforts are needed. In Section 2, we review a number of research lines where problems similar to those appearing in this domain have been explored for a number of years. In Section 3 we formalise sharing communities as graphs and reformulate some key properties of partners and exchanges among them in graph-theoretical terms. This allows us to define sharing policies as algorithms running at each node. Section 4 develops the basis for a network-based model of cyber security information. Building upon the formats already developed, we point out the need for richer models where individual pieces of information can be annotated with labels reflecting, for example, our perception of its value or the trust we have on it being true. Moreover, connections among data need to be construed and made explicit, offering a view of an *information network* rather than a (more or less structured) list of items. In Section 5 we propose and discuss a risk-aware sharing algorithm. Section 6 concludes the paper by pointing out open problems and some lines of work that we are currently exploring.

2. RELATED WORK

In this section we review a number of research areas connected with the general problem of cyber security information sharing. In some cases, the connection is straightforward, although related to very concrete problems; in others, challenges similar to those appearing in this domain have been approached with techniques that might prove useful if conveniently adapted.

A. *STRUCTURED MODELS OF CYBER SECURITY INFORMATION*

As a discipline, cyber security deals with heterogeneous information related to the assets and configurations present in a system; the threats and tactics used by attackers; indicators of on-going incidents; countermeasures applied to mitigate risks; etc. Over the last decade, considerable efforts have been devoted to categorise such information and standardise data formats and exchange protocols, most notably through the Making Security Measurable (MSM) [6] initiative led by MITRE. Key aims of MSM include *"improving the measurability of security through registries of baseline security data, providing standardized languages as means for accurately communicating the information, defining proper usage, and helping establish community approaches for standardized processes."*[1]

MSM presents a comprehensive architecture for cyber security measurement and management, where current standards are grouped into processes and mapped to the different knowledge areas. Current MSM standards can be grouped into 6 major knowledge areas, each of which refers to a process (put in parentheses): **A**sset definition (inventory); **C**onfiguration guidance (analysis); **V**ulnerability alerts (analysis); **T**hreat alerts (analysis); risk/attack **I**ndicators (intrusion detection); and incident **R**eport (management). MSM standards and knowledge areas. Table I relates current MSM standards to these areas[2]:

---

[1] See http://measurablesecurity.mitre.org

[2] We refer the reader to *Appendix A for a description of MSM's acronyms, and to MSM's main website for further details.*

Table I. MSM standards and knowledge areas.

	CPE	OVAL	SWID	XCCDF	CCE	OCIL	CCSS	CVE	CWE	CVSS	CAPEC	CVRF	MAEC	CybOX	IndEX	STIX	IODEF	CPE	CEE	RID	RID-T	CYBEX	CWSS
A	•	•	•															•					
C		•		•	•	•	•																
V		•						•	•	•		•											
T								•	•	•	•		•	•	•	•	•		•	•	•		
I	•							•					•	•	•	•	•	•	•	•	•		
R	•	•			•			•	•	•			•		•	•	•			•	•	•	•

In the near future, it seems quite plausible that information sharing activities will be supported by infrastructures and mechanisms based on these standards, either in their current form or in subsequent revisions and developments.

B. COLLABORATIVE ATTACK DETECTION SYSTEMS

Many cyber attacks can only be detected by gathering and correlating evidences obtained at different locations [8]. In some cases, such evidences may come from sources unavailable to us and over which we have little control. This is, for example, the case of organisations that choose to share information about detected security events, possibly in near real-time, so as to minimise risk exposure or the impact of on-going cyber attacks. The so-called Collaborative Intrusion Detection Systems (CIDS) [9][1] constitute a clear example of the benefits that information sharing can offer to modern cyber defence capabilities. In principle, they have the potential to detect attacks that affect different Internet networks by correlating attack alerts. Besides, they also could reduce the costs involved in attack detection by sharing intrusion detection resources among networks.

CIDS consist of multiple distributed detection units logically organised in a network topology. In centralised systems, such as DIDS [10], DShield [11] and NSTAT [12], each sensor shares alerts with a central correlation unit. Hierarchical approaches (e.g., GrIDS [13], EMERALD [14] and DSOC [15]) attempt to address the scalability issues of centralised approaches by organising detection units into a tree-like topology. Finally, fully distributed approaches such as DOMINO [16] or the one proposed in [17] work in a P2P fashion, with nodes participating in a periodic exchange of information. We refer the reader to [1] for a more comprehensive account of existing CIDS technology.

Unfortunately, CIDS involving different partners are rare nowadays, as organisations are particularly reluctant to share sensitive information with almost any other actor. Apart from privacy issues, trust plays an important role in CIDS too. In most cases, the overall detection accuracy depends on all parties exhibiting honest behaviour,

particularly in terms of the trustworthiness of reported alerts. These issues are ignored or inadequately addressed in existing CIDS, in part because most of them were not conceived for an information sharing setting involving multiple and heterogeneous organisations.

C. TRUST AND REPUTATION MANAGEMENT SYSTEMS

In many fully distributed applications there is often a lack of a central authority in charge of monitoring users and reporting about their behaviour. In these scenarios, users often have to make decisions about who to trust for certain tasks (e.g. selecting routes in a MANET). Trust and reputation management systems have proliferated lately as a potential solution to this problem. Roughly speaking, these systems are based on the principle that users might quantify other users' behaviour by collecting and aggregating recommendations referring to past interactions with them. The interested reader can find surveys of trust systems in [18][19][20].

Possibly the central rationale underlying the utility of trust and reputation systems is that the behaviour exhibited by an entity in the past can be used to predict the expected outcome of future interactions. For cyber security information sharing scenarios, we anticipate that trust and reputation will play a key role in tasks such as deciding on whether to share some information with someone or not, or assessing the reliability and accuracy of pieces of data coming from questionable sources (e.g., using the aggregated value of previous data provided by one party as proxy for the a priori value of future information).

D. FLEXIBLE ACCESS CONTROL MODELS BASED ON RISK ESTIMATES

Imposing restrictions on sensitive information flows is a long-established problem in computer security. Traditional models of multi-level security, such as Bell-La Padula [21], deal with this problem by associating security clearances with subjects, security classifications with objects, and providing clear decision rules as to whether an access request should be granted or not. However, such mechanisms encode for a pre-determined calculation of risks and benefits, and in many modern situations preclude effective operations that can be justified on a risk basis when the specifics of the context are taken into account. The JASON Report [22] raised concerns about the inability of many organisations, particularly those in the national security and intelligence arena, to rapidly process, share and disseminate large quantities of sensitive information, in part due to the inflexibility of current access control models. Even worse, organisations are increasingly resorting to ad hoc means to surpass these restrictions, such as granting temporary authorisations for high-

sensitive objects or, as mentioned in [22], to follow the line of the old saying "it is better to ask for forgiveness rather than for permission."

Motivated by these issues, a number of works have proposed in the last years more flexible access control models based on an explicit quantification of the risk associated with every access request. For example, FuzzyMLS [23] replaces the classical binary allow/deny decision in BLP by a risk estimate that extends BLP rules to a continuous case. In [24], the model is extended to support uncertainty in security labels and clearances, and to account for the time dimension of sensitivity. Works in this area have proliferated in the last years, with a variety of proposals, including risk-based access control built on fuzzy inferences [25]; attribute-based risk-adaptive models [26]; role-and-risk based models [27]; benefit and risk access control [28]; and many others. Although the majority of these works explicitly target the particularities of information sharing settings, to the best of our knowledge none addresses cyber security information sharing.

3. A FORMAL MODEL OF INFORMATION SHARING COMMUNITIES

A. *COMMUNITY STRUCTURE*

We represent an information sharing community as a weighted directed graph (digraph) $G = (V, E)$, where V is the set of nodes or vertices that represent the entities that are member of the community, while E is the set of the edges or links that represent the information flows permitted within the community. For each edge $e = (u,v) = uv$, we denote by $e^{-1} = vu = (v,u)$ its inverse, if it exists.

In information sharing terminology, u corresponds to an originator of information, while v is a recipient of information. Therefore, edges restrict not only the members that may share information amongst them but also who distributes the information within the community and with whom. Please note that the originator does not necessarily correspond to the source of the information. The latter is the entity that produces an item of information. As the source does not need to be a member of the community, for simplicity we do not consider them in our model. Thus, an originator u that shares information with a recipient v can transmit information produced on its own (i.e. u is the source as well), forward information received from other nodes (i.e. u behaves as a forwarder of information), or both.

A graph that permits multiple edges between nodes is called a multigraph. We generalize the representation given above for an information sharing community to formally include the multigraph notation:

$G = (V, E, \Psi)$

where $E = \{e_1, e_2, ..., e_m\}$ is a set of symbols representing the edges of the graph, and $\Psi: E \rightarrow E(V)$ is a function that attaches an ordered pair of nodes to each $e \in E$: $\Psi(e) = uv$, u and v being nodes.

In our digraph (information sharing domain), if $\Psi(e_1) = \Psi(e_2)$, then $e_1 = e_2$. As a digraph has directed edges, two different edges that have the same ends (e.g. uv, vu) must have a different predecessor node (originator). In other words, the direction of each edge must be opposite to the other. This restriction conditions the structure of the multigraph, as there cannot be two equally directed edges between two nodes u and v. We do not consider loops (edges with ends uv / $u = v$) either, as sharing information with oneself is given per se.

It should be noted that the graph representing an information sharing community may contain cycles, and this will depend solely on the community structure.

B. LINKS BETWEEN NODES

Definition 1. Let $e_i = u_i u_{i+1} \in E$ *for* $i \in [1,k]$. The sequence $W = e_1 e_2 e_3 ... e_k$ is a walk of length k from u_1 to u_{k+1}. It should be noted that e_i and e_{i+1} must be adjacent $\forall\, i \in [1, k-1]$. For simplicity, we write $W: u_1 \rightarrow u_2 \rightarrow u_3 \rightarrow ... \rightarrow u_k \rightarrow u_{k+1}$ or $W: u_1 \sim^n u_{k+1}$ to represent a walk of length n from u_1 to u_{k+1}.

Definition 2. A walk $W = e_1 e_2 e_3 ... e_k: u \sim v$ is a directed walk, if $e_k \in E$, $\forall\, i \in [1,k]$, $u \in e_1$ is the originator of information and $v \in e_k$ is the latest recipient of the information. A directed path $P: u_1 \rightarrow u_k$ is a directed walk where $u_i \neq u_j$, $\forall\, i \neq j$. A directed cycle is a directed path where $u_1 = u_{k+1}$.

We consider that one of the next four possibilities can occur in an information sharing community between any two indistinct nodes *(u, v)*:

(1) there is no directed path that connects u and v, and therefore they cannot share information between them, neither directly nor indirectly.

(2) there is no edge that connects u and v, that is, there is no direct connection between them. However, they could share information using a directed path that connects them indirectly (e.g. $W: u \rightarrow w \rightarrow v$).

(3) there is a directed edge from u to v or from v to u.

(4) there are two directed edges that connect both nodes, being u and v both originators and recipients of information.

Definition 3. Two nodes are unconnected if there is no directed edge or path that connects both nodes. Two nodes are strongly connected (adjacent) if there is a directed edge that connects them independently of the edge direction. Finally, two nodes are weakly connected if there is a directed path that connects them but where there is no directed edge between them.

C. TYPES OF NODES

We classify the nodes in a community using the indegree ($deg^-(u)$) and outdegree ($deg^+(u)$) properties of a node, which specify the number of head and tail endpoints, respectively, adjacent to a node. Formally:

$$deg^-(u) = |\{e \in G / e = xu\}|$$

$$deg^+(u) = |\{e \in G / e = ux\}|$$

In graph theory, the node with $deg^-(u)$ equals zero is called a source, while a node with $deg^+(u)$ equals zero is called a sink. In our information sharing community scenario, we identify three types of nodes, two of them according to the balance between their indegrees and outdegrees. Let Ω be the difference between $deg^-(u)$ and $deg^+(u)$ of a node n.

$$\Omega(u) = deg^-(u) - deg^+(u)$$

Definition 4. We say that a node $u \in V$ is a distributor if $\Omega(u) << 0$, and $\Omega(u) \in N^-$ (negative integers). A distributor is expected to receive information from a few originators and provide information to many recipients.

Definition 5. We say that a node $u \in V$ is a collector if $\Omega(u) >> 0$, and $\Omega(u) \in N^+$ (positive integers). A collector is expected to receive information from many originators and provide information to few recipients. When $deg^+(u) = 0$ (sink), the information received by the collector is not further shared with other community members.

On the other hand, we use the betweenness centrality property to define the third type of node in a community. The betweenness quantifies the number of times a node is part of the shortest path between two other nodes. This provides a measure of the relevance of that entity within a community in terms of presence in information sharing routes. Given a connected graph G with a weight function α: $E \rightarrow N$, the shortest path between two nodes u and $v \in G$ is the path P with the minimum total

weighted distance between u and v:

$$d^{\alpha}{}_{G}(u, v) = min\{\sum_{e \in P} \alpha(e) / P: u \sim v\}$$

Thus, the betweennes centrality $Cb(u)$ of a node u is given by

$$Cb(u) = \sum_{r \neq k \neq v} \sigma_{r,k}(u) / \sigma_{r,k}$$

where $\sigma_{r,k}(u)$ is the number of shortest paths between any two nodes r and $k \in V$ that pass through u, and $\sigma_{r,k}$ is the total number of shortest paths between any two nodes r and k ∈ V. For example, in a centralized sharing approach (e.g. an Information Sharing and Analysis Center), the central node has values of betweenness much higher than any other node of the community, while sources and sinks are expected to have a zero betweenness centrality.

Definition 6. We say that a node $u \in V$ is a bridge if its betweenness centrality $Cb(u)$ has a value higher than the arithmetic mean of all nodes of the graph G.

$$u \text{ is bridge} \leftrightarrow Cb(u) > Cb(v)/|V|, \forall v \in V$$

Figure 1 exemplifies the properties described above.

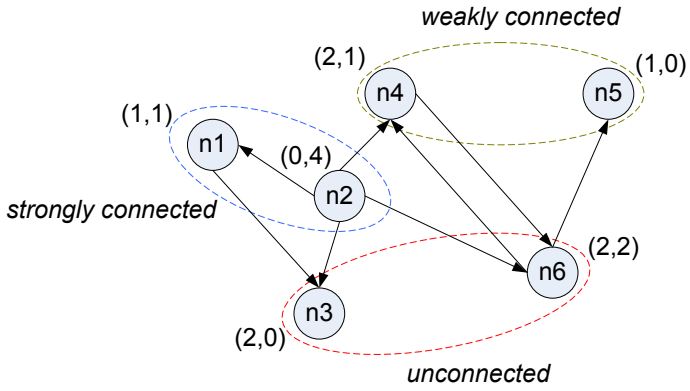

Figure 1. A graph example. The pair of numbers associated with each node indicate the indegree and outdegree values. Node n2 is a distributor, while nodes n3 and n5 are collectors. Node n6 is the only node that is part of a shortest path between two other nodes of the graph (i.e. path between n4 and n5), and thus, is the only one that complies with the bridge definition given above.

D. RISK ASSESSMENT FUNCTION

The edges of the graph have a weight that, in our case, is a richer concept than traditional edge weights. For cyber security information sharing communities, we add metadata to each edge that represents a function that calculates the risk level to which the originator of the information is exposed if certain piece of information is shared with certain strongly connected recipient.

In our scenario, we propose a simple formula for defining the risk to which a node is exposed when sharing information. It basically depends upon two well-differentiated factors that have played a key role in well-recognized risk assessment methodologies:

- First, the value of the information shared, and thus the impact caused on the entity (originator node) in the case that such information is accessed by unauthorised entities. We address this point later in Section 4, including those cases where the value of information varies over time [24].
- Second, the probability that such information is accessed by unauthorised entities (any node in the graph).

For the impact, its value strongly depends on the particularities of the originator, the information shared, and other contextual information.

It should be noted that the quantification of both the impact and the probability values is a difficult task whose precise estimation is generally impossible, as the knowledge required to do so is incomplete. For instance, the calculation of the probability of occurrence may be improved using intelligence obtained from the terrain (e.g. OSINT, HUMINT, trends, market analysis, etc.), but, unfortunately, we usually end up with a rough estimation that will be substantially different to the underlying reality. Notwithstanding, this problem is out of the scope of the present paper, and thus we do not question the trustworthiness of these values when used in our formulae.

Definition 7. Let $u \in V$, and δ be a piece of information originated by u. We define $I(u, \delta) \in [0, 1]$ as a measure of the impact on u caused by an unauthorised access to δ.

On the other hand, a node u may not be able to estimate the second factor above (the probability), especially when the recipient can further share this information with third nodes in the graph (and so forth). Instead, we use the trust that node u (originator) has in node v (recipient). Intuitively, a higher level of trust should pose a lower probably of unauthorised access if such trust has been adequately assessed based on empirical data or any other information derived from past experiences.

Specific mechanisms for trust computation fall out of the scope of the paper, though some proposals can be found in Section 2 C.

If *v* can share this information with a third-level authorised node *w*, then the risk level should consider the probability that *w* leaks the information to an unauthorised entity as well. For this case, the risk value also depends on the trust that *u* has on *w*. If it cannot be directly inferred by *u*, then it can be calculated using indirect means, such as by combining the trust that *u* has in *v* with the trust that *v* has in *w*.

Definition 8. Let $u, v \in V$. We define $T(u,v)$ as the function that calculates the trust that *u* has on *v*, denoted by $T(u,v) = \{t / t \in [0, 1]\}$.

In conclusion, we formalize the risk function β in a directed multigraph as follows.

$$\beta(u,v,\delta) = I(u, \delta) \bullet (1 - \Pi_{s \in S} T(u,s))$$

$$\text{where } S \subseteq V / s \in P(u_i, u_{i+k}), \text{ with } u_i = v, \forall s \in S.$$

In a nutshell, the risk value is computed multiplying the impact by the probability that any node weakly connected to the originator *u* through *v* discloses the information to an unauthorised entity. Please note that the resultant probability is expressed as 1 – the probability that no node discloses the information, and also considers the trust value of *u* on *v*.

Next, the risk function is generalized in order to calculate the risk to which a node is exposed in the case that it shares the information will all its strongly connected nodes:

$$\beta(u,\delta) = I(u, \delta) \bullet (1 - \Pi_{i = v \in V} A_{u,i} \bullet \Pi_{s \in S} T(u,s))$$

where $A_{u,i}$ is the adjacency matrix for *u*, and S the set of nodes reachable through a directed path starting in *i*, for all *i* strongly connected node with *u*.

$$A_{u,i} = \begin{cases} 1 & if\ nodes\ u\ and\ i\ are\ strongly\ connected \\ 0 & otherwise \end{cases}$$

The function above is applicable as long as $|A_{u,i}| > 0$.

For the risk value computation we assume that the nodes do not collude, and thus, the probability is calculated unconditionally.

From the formula $\beta(u,v,\delta)$ above, it can be easily inferred that the minimum and

maximum risk values for an originator *u* that shares a piece of information δ with a strongly connected recipient *v* correspond to 0 (i.e. both *v* and all weakly connected nodes are fully trusted, that is, $T(u,s) = 1 \ \forall s \in S$) and $I(u, \delta)$ (i.e. at least one of the nodes is fully distrusted, that is, $T(u,s_i) = 0$), respectively. Also, it can be observed that the risk value increases with the number of recipients, unless these are fully trusted.

It should be noted that the risk value is a dynamic value that has to be updated (re-calculated) when any of the forming factors changes, such as the impact value or the trust in any of the related nodes.

4. FORMAL MODELS OF CYBER SECURITY INFORMATION

We represent the knowledge of information security as a weighted graph $G = (V, E)$, where V is the set of nodes or vertices that represent pieces of well structured information from an *element of knowledge*, while E is a relation set given by $E \subseteq V \times V$, in which each of whose members is a pair (i.e., edge or link) representing a relationship between the different pieces of information.

Definition 9. Elements of knowledge κ define a set of ontologies $\kappa = \{\kappa_1, ..., \kappa_n\}$, where each κ_n encodes some knowledge about different domains of information security.

In our domain, nodes represent heterogeneous pieces of information from elements of the set κ. Thus, we represent this variety of information by using vertex-labelled graphs.

Definition 10. Let $\rho^{k_n} = \left\{\rho_1^{k_n}, ..., \rho_m^{k_n}\right\}$ be a set of properties of a specific κ_n. We say that two nodes *u* and *v* are related, denoted $\sigma(u, v)$, if there is a pair of properties (ρ_i, ρ_j) such that $\int(\rho_i^{k_u})$ is equal to $\int(\rho_j^{k_v})$. Formally:

$$\sigma(u,v) = true \leftrightarrow \exists(\rho_i, \rho_j) / \forall \rho_i \in \rho^{k_u}, \forall \rho_j \in \rho^{k_v} : \int(\rho_i^{k_u}) = \int(\rho_j^{k_v})$$

For simplicity, we denote the property throughout κ_u is associated with κ_v as $p^{k_{u,v}}$, and we denote the piece of information that satisfies $\sigma(u, v)$ as $\int(\rho^{k_{u,v}})$.

In our domain, an edge is represented as $e=(u, v, p)$, where *u* and *v* are two pieces of information that satisfy $\sigma(u, v)$, and *p* is a label representing the shared property $p^{k_{u,v}}$. We represent each shared property using edge-labelled graphs.

Henceforth, each node in this model corresponds to an element of the information security knowledge, and each edge corresponds to a relation between those elements.

Figure 2. shows an exemplification of an information knowledge graph structure, where assets are connected to each other, as well as to specific configurations. Additionally, exploits can target different vulnerable configurations held by the several assets. For instance, asset A_3 represents an Oracle Java runtime environment application installed on A_1 –a Red Hat Linux server. Here, certain configuration C_1 allows the asset to execute Java applet scripts (CCE-10083-4).

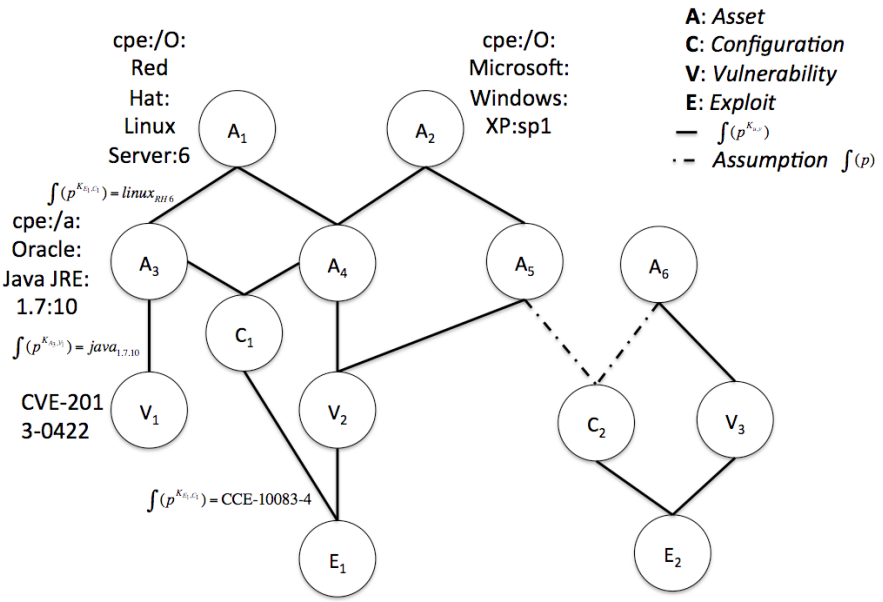

Figure 2. A graph example. Different elements of knowledge (CVE, CPE, CCE) are related by a number of pieces of information.

A. *INFORMATION VALUE AND REASONING OVER GRAPH STRUCTURES*

We present a quantification of information value based on the relation between different elements of knowledge. More precisely, we position that the value of the information directly depends on the information already owned by a community member, how this information is structured and to what extent is related to other pieces of information. In this regard, identifying missing information can also contribute to the quantification of information value. Furthermore, there are other key attributes for quantifying the information value, such as its relevance, timeliness, and accuracy, to name a few.

Definition 11. Let *c* be the *cost* of a piece of information $\int(\rho^{k_{u,v}})$, *t* the time window where such cost holds (assuming that cost decays over time), *a* the degree of reliability that the piece of information is *accurate*, and *r* how *relevant* the information is for an organization. We define the *value of information* of a node *V* as:

$$Vol_v(c,t,a,r) = \omega \cdot \psi(c_v,t_v,a_v,r_v) + (1-\omega) \cdot \sum_{u \in V : \sigma(v,u)=true} Vol(c_u,t_u,a_u,r_u)$$

where ψ determines the subjective value of a vertex *V* for a given community and the second term factors in the aggregated value of adjacent nodes. By *cost* we mean here the amount of resources (economic, computational, etc.) needed to acquire and process the information. In our model we assume that there are markets where such information can be acquired, and that costs can be known. On the contrary, the *value* is specific to each party and will very likely vary over time. As an example, we suggest the next measure for the subjective value of information:

$$\psi(c_v,t_v,a_v,r_v) = (c_v \cdot a_v) \cdot e^{-k \cdot r_v \cdot \left(\frac{t}{t_v}\right)}$$

where the first term represents the value of the information and the second an exponential decay function over time, *k* being a decay constant weighted by *r*. In other words, the value of relevant pieces of information will be exponentially bigger than non-relevant pieces and it will decay slower over *t*. Note, too, that the relevance may also serve to modulate the risk of disclosure of the information.

Here, the relationship among different nodes could be expressed in a more complex way. For instance, some relations are often due to **causality**, and some others are subject to a perception error, i.e. **uncertainty**. In this regard, graph described on Figure 2. shows how easy could be reasoning using a graph structure. For instance, if we knew that a given asset A_5 is from the same vendor as A_6, and we knew that the latter have an exploitable configuration, we could reason that the same exploit might be applied to A_5 with a certain probability.

Furthermore, graph structures also allow us to establish possible paths from a type of node, e.g. an asset, to all other nodes of the same or different type, e.g. exploits. In this regard, different conclusions might be extracted depending on the type of nodes through the path. On the one hand, if there were a direct connection between nodes of the same type such as A_1 and A_3 in Figure 2. , compromising A_3 would also compromise A_1 and A_4 –as asset A_1 is the operating system executing application A_3 and A_4. On the other hand, if there were a connection between nodes of different type such as A_3 and E_1 through a node of types vulnerability and configuration (V_2 and C_1 in the aforementioned example), we could conclude that A_3 could be compromised using E_1.

Thus, reasoning over heterogeneous graph structures in a complex task, which requires context-based reasoning, i.e., type of node, length of the walk, etc. Note that this section intends to introduce the concept of information value over graph structures, and we refer the reader to forthcoming publications for a deeper treatment of reasoning about cyber security information and its value.

5. EXAMPLE: AN INFORMATION SHARING ALGORITHM

In this section we present an algorithm that aims at achieving the need-to-share concept [29], so as to maximise the information sharing within a community while the risk value for the originator of information is kept below an established threshold (i.e. an acceptable risk level). In the next Subsection we first describe the general aspects behind the algorithm, and in the subsequent, a running example to illustrate its behaviour.

We do not claim that this algorithm is the only one applicable to the information sharing scenario. Actually, there are a number of approaches, where the most appropriate one should be selected depending on the particularities of the community, the policies applicable to the originator, the information to share at each moment, and other contextual information. For instance, the same node may decide to apply a different algorithm for different pieces of information depending on their level of classification. Or the same node may select a different algorithm for the same piece of information at different moments (e.g. a less conservative approach may be followed in a crisis situation).

A. *OVERVIEW*

The algorithm is a greedy algorithm in the sense that it follows the problem solving heuristic of making the locally optimal choice at each stage. The problem to solve at each stage corresponds to whether or not sharing a certain piece of information with an adjacent node depending on the accumulated risk value and the threshold established by the originator.

The algorithm consists of two well-differentiated phases. In the first one, that we call *Decision Phase*, the originator performs a simulation of how the information should be shared across the community in order to keep the accumulated risk value that results from the subsequent sharing actions below the desired threshold. At the end of this phase the originator is able to conclude what nodes of the graph are authorised to access the information.

In the second phase, named *Sharing Phase*, the sharing process itself is undertaken, started by the originator, and by which the pertinent information that allows each sharing node to know who are the authorised nodes amongst its adjacent ones is also transmitted.

During the *Decision Phase*, the simulation orders the adjacent nodes of a certain sharing node n_i by their trust value from higher to lower, discarding those in whom the originator fully distrusts ($T(u, x_i) = 0$) as well as those that have been already marked as authorised node. Then, it calculates the accumulated risk value $\beta_A(u,v,\delta)$, being u the originator and v the node adjacent to u through which n_i has been reached. β_A formula considers the trust values of every node that has already been marked as authorised nodes, plus the adjacent nodes of n_i. If the resultant value of β_A is greater than the established threshold, then the simulator discards the last adjacent node of the ordered list, and recalculates β_A. The analysis is iterated until the obtained β_A is below the threshold. The adjacent nodes that remain in the list when this condition is satisfied are marked as authorised nodes.

The simulation stops analysing a certain sharing node if any of the following conditions is met:

- The sharing node has an outdegree equals zero.
- The ordered list is empty or, after discarding the adjacent nodes during the β_A calculation, there is no one left. This means that the sharing node will not be authorised to share the information with any of its adjacent nodes.
- The sharing node already received that piece of information (the algorithm considers cyclic graphs).

At the end of the *Decision Phase*, a subset of nodes $V' \subseteq V$ and edges $E' \subseteq E$ will have been selected. The resultant subgraph $G' = (V', E')$ is an acyclic directed graph (i.e. tree) where the root node is the originator, the rest of the nodes are those authorised to access the information and the edges represent the sharing links between the nodes. In principle, the search strategy of the *Decision Phase* could be configured to follow either a Breadth- First -Search (BFS) or a Depth-First-Search (DFS) [30] as both approaches have the same time (O|E|) and space (O|V|) bounds. However, the vertex ordering produced in BFS (i.e. the order in which the vertices are explored) better reproduces the behaviour expected in an information sharing community. In these communities, each node is strongly connected to other nodes in which it explicitly trusts. Therefore, it is expected that any originator will preferably share the information with these nodes in the first instance, rather than leveraging on weakly connected nodes the increase of the accumulated risk value β_A.

The difference between the sharing process for the originator and any other node is that, for the latter, they can share the information (as long as the threshold condition

is satisfied) with adjacent nodes with which the originating node has no explicit trust calculated. For these nodes, the algorithm uses an indirect trust computation using the path of nodes from the originator to those nodes. For instance, the indirect trust computation for node *u* on a node *x* weakly connected through the walk *W: u* → *v* → *w* → *x* is as follows:

$$T(u,x) = T(u,v) \bullet T(v,w) \bullet T(w,x)$$

This approach helps maximizing the information sharing by permitting the sharing with unknown nodes as long as the threshold is not exceeded.

B. AN EXAMPLE

In this section we show the application of the *Decision Phase* to the graph example shown in Figure 3. following a BFS approach and considering the table of trust shown in Table II. In our example, the node n1, as the originator, wishes to share some piece of cyber security information with the information sharing community.

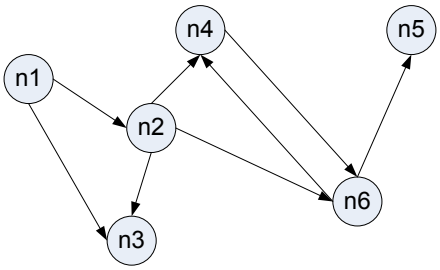

Figure 3. Graph G.

Table II. Table of trust for graph G. A value of 0 means no trust at all; a value within the range (0,1) means relative trust; and 1 means full trust. If a node has no explicit (dis)trust in some other node, then no value is indicated. No edge appears in the graph between those nodes if there is an explicit distrust or when no explicit trust exists.

$T_G(n_i)$	n_1	n_2	n_3	n_4	n_5	n_6
n_1	1.00	0.75	1.00	-	0.00	-
n_2	-	1.00	0.05	0.6	-	0.35
n_3	-	-	1.00	-	-	-
n_4	-	-	-	1.00	-	0.80
n_5	-	-	-	-	1.00	-
n_6	-	-	-	0.25	0.90	1.00

We define $\varphi_G(u,\delta)$ as the risk threshold, that is, the maximum risk value acceptable by u for the piece of information δ within the information sharing community G.

The initial values for our example are the following:

$\varphi_G(n_1,\delta)$: 0.7 $I(n_1, \delta)$: 0.3 $AuthNodes:=\{\}$

The initial values should be result of a risk analysis carried out by the originator, and by which the maximum tolerable risk $\varphi_G(u,\delta)$ and the impact $I(n_1, \delta)$ for the piece of information δ can be estimated. In this example both the initial values and the table of trust shown in Table II have been selected to serve for illustrative purposes only.

It is worth mentioning that in many cases the originator can exert some control over the impact –and, therefore, over the maximum tolerable risk– by selectively removing sensitive parts of the information to be shared. In scenarios other than cyber security information sharing this is commonly achieved by anonymising data, e.g. by removing or aggregating pieces of information.

We next proceed with the example:

1. Analysis of sharing node n_1 (originator)

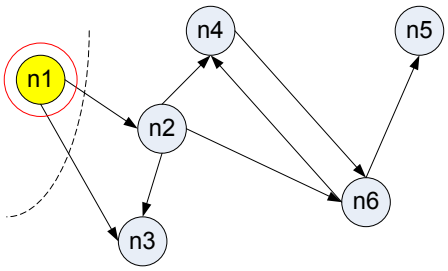

Figure 4. Sharing process for originator n_1

Ordered list of adjacent nodes $(n_1) := \{n_3, n_2\}$

Calculate accumulated risk level:

$$\beta_A(n_1,n_1,\delta) = I(n_1, \delta) \bullet (1 - T(n_1, n_1) \bullet T(n_1, n_3) \bullet T(n_1, n_2)) = 0.3 \bullet (1 - 1 \bullet 1 \bullet 0.75) = 0.075$$

If $0.075 < \varphi_G(n_1,\delta)$ then update *AuthNodes* and share with nodes remaining in the ordered list:

$$AuthNode\sigma := \{n_3,n_2\}$$

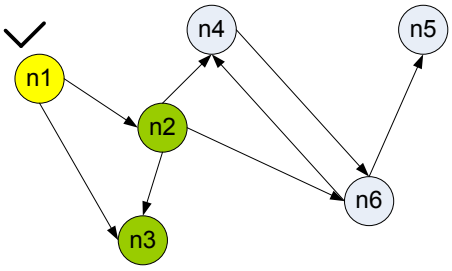

Figure 5. Result of sharing with n_2 and n_3[3]

[3] For clarity purposes, we mark the nodes that have already been analysed (n_2 in this case).

2. Analysis of sharing node n_3

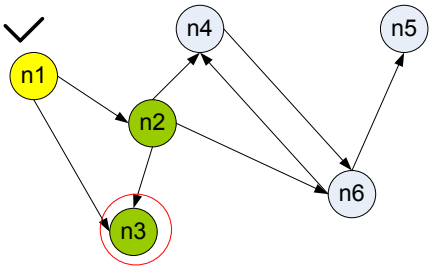

Figure 6. Sharing process for n_3

Stop condition applies: $deg^+(n_3) = 0$

3. Analysis of sharing node n_2

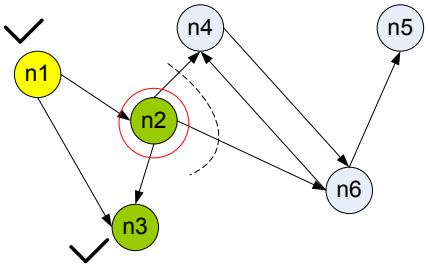

Figure 7. Sharing process for n2

Ordered list of adjacent nodes $(n_2) := \{n_4, n_6\}$[4]

Calculate accumulated risk level:

$$\beta_A(n_1,n_2,\delta) = I(n_1, \delta) \bullet [1 - T(n_1, n_1) \bullet T(n_1, n_3) \bullet T(n_1, n_2) \bullet T(n_1, n_4) \bullet T(n_1, n_6)]$$ [5] $$= I(n_1, \delta) \bullet [1 \quad T(n_1, n_1) \bullet T(n_1, n_3) \bullet T(n_1, n_2) \bullet (T(n_1, n_2) \bullet T(n_2, n_4)) \bullet (T(n_1, n_2) \bullet T(n_2, n_6))] = 0.3 \bullet [1 - 1 \bullet 1 \bullet 0.75 \bullet (0.75 \bullet 0.6) \bullet (0.75 \bullet 0.35)] = 0.273$$

If $0.273 < \varphi_G(n_1,\delta)$ then update *AuthNodes* and share with nodes remaining in the ordered list:

---

4 Please note that n_3 is not included as it has already been marked as authorised.

5 We underline the new factors that are incorporated to the B_A formula.

$AuthNode\sigma := \{n_3, n_2, n_4, n_6\}$

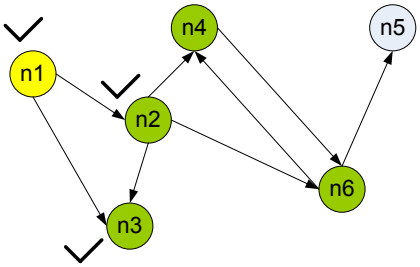

Figure 8. Result of sharing with n_4 and n_6

4. Analysis of sharing node n_4

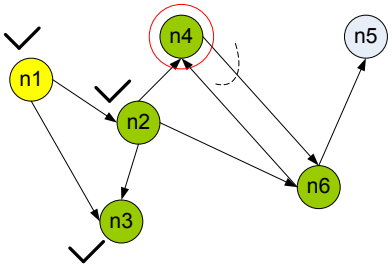

Figure 9. Sharing process for n_4

Ordered list of adjacent nodes $(n_4) := \{\}$[6]

Stop condition applies: list is empty.

[6] In this case, the ordered list is empty as n_6, the single node adjacent to n_4, has already been marked as authorised.

5. Analysis of sharing node n_6

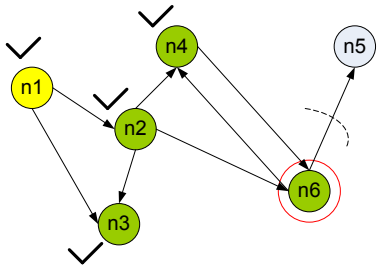

Figure 10. Sharing process for n_6

Stop condition applies: list is empty[7].

After the application of the *Decision Phase*, the list of authorised nodes to which the information can be shared is $\{n_3,n_2,n_4,n_6\}$.

[7] In this case, the ordered list is empty as n_5, the single node adjacent to n_6, is fully distrusted by the origin n_1.

6. CONCLUSIONS, CHALLENGES AND FUTURE WORK

Information sharing will be central to cooperation activities in cyber security operations. But the benefits derived from being a member of an information sharing community are not always perceived in the same way by different entities. Furthermore, organisations might well be reluctant to share sensitive information with partners whose trustworthiness is unclear and/or when the repercussions of sharing are not properly understood. These and other factors have been already identified as major inhibitors for the proliferation of information sharing communities and deterrents to members' active participation when being part of a community. In this paper, we shed some light on a few of these questions and point out the need to attack the problem from a formal perspective. In particular, we suggest analysing the *topology* of sharing by modelling as graphs both the community and the information network. In doing so, we can leverage a number of tools from a number of disciplines –notably graph theory, complex networks, and social network analysis– to study relevant aspects of the problem.

Due to space reasons, in this paper we have not given a deep account of any of these problems. Rather, our aim is to raise awareness about the benefits that such a perspective could bring to information sharing in cyber security. Our formal treatment of the information network and sharing communities, including the sharing algorithm discussed above, attempts to be merely illustrative of the potential that this approach could yield. In fact, this issue have received much attention in other contexts where information sharing is essential for agents that cooperate towards a common goal. For example, Zhu et al. present in [31] an algorithm to share information among a set of agents that operate in an ad hoc fashion. Each agent must decide whether to broadcast sensed and/or received information to neighbouring members. The approach is similar to ours in the sense that the problem is couched as one of optimal decision-making. However, the focus in [32] is on maximising sharing and minimising communication cost, whereas in cyber security risk factors are paramount.

We are currently exploring in greater depth several of the work areas discussed throughout this paper. Specifically:

- Some metrics and techniques well known in complex and social network analysis can easily be reinterpreted in this domain. For example, *information centrality* measures the efficiency of a network in delivering information. Similarly, the *betweenness centrality* of a node measures the importance of node in a network in terms of how many shortest paths between any other pair of nodes pass through it. Both measures, together with other centrality

quantities, can be valuable in establishing efficient sharing policies and assessing attributes of individual participants. Analogously, the centrality of a piece of information could be used as a proxy for, e.g., its relevance.

- Trust- and risk-based algorithms for the dissemination of information through the community. We are currently developing flexible but robust schemes that take as input a description of the sharing context (e.g., need-to-share this data, maximum risk allowed, etc.) and choose paths along the community graph so as to maximise dissemination while keeping risk of disclosure under control. We believe that this issue is particularly relevant when automating information exchange mechanisms, as the risks of sharing too much are apparent. However, unintended disclosures have very different consequences if the receiver is a highly trusted ally or an occasional collaborator, hence the need to explicitly consider trust in the decision making process. Similarly, privacy issues might be a major deterrent when parties face the problem of whether to share or not [31]. In this regard, both technical (e.g., trust-building mechanisms, data anonymisation) and non-technical (e.g., mutual agreements) measures should be further explored.
- Resilient but trusted communities. In many contexts, it is crucial to ensure that information reaches the intended recipients in time and with some minimum guarantees of risk containment. This requires building a community where paths with sufficient trust are always present, avoiding the presence of bridge nodes (i.e., nodes necessarily present in a subset of paths) and cut edges/nodes (i.e., those that make subset of nodes disconnected from each other if removed).

Acknowledgements

We thank the anonymous reviewers for their insightful comments and valuable suggestions, which have contributed to improve the quality of this work.

REFERENCES

[1] C. Fung, J. Zhang, I. Aib, and R. Boutaba, "Trust management and admission control for host-based collaborative intrusion detection," *Journal of Network and Systems Management*, vol. 19, no. 2, pp. 257-277, 2011.

[2] B. Schneier. (2012) The Vulnerabilities Market and the Future of Security. [Online]. http://www.schneier.com/blog/archives/2012/06/the_vulnerabili.html

[3] R. Langner, «Stuxnet: Dissecting a Cyberwarfare Weapon,» *IEEE Security & Privacy*, vol. 9, no. 3, pp. 49-51, 201.

[4] B. Bencsath, G. Pek, L. Buttyan, and M. Felegyhazim, «The Cousins of Stuxnet: Duqu, Flame, and Gauss,» *Future Internet*, vol. 4, no. 4, pp. 971-1003, 2012.

[5] D.E. Denning, «Stuxnet: What Has Changed?,» *Future Internet*, vol. 4, no. 3, pp. 672-687, 2012.

[6] R. A. Martin, «Making security measurable and manageable,» in *IEEE Military Communications Conference (MILCOM)*, 2008, pp. 1-9.

[7] L. Dandurand, «Cyber Defense Data Exchange and Collaboration Infrastructure (CDXI),» in *ITU-T Workshop*, 2010.

[8] S. Teng, W. Zhang, X. Fu, and W. Tan, «Cooperative intrusion detection model based on scenario,» in *Proc. 11th International Conference on Computer Supported Cooperative Work in Design*, 2007, pp. 876-881.

[9] C.V. Zhou, C. Leckie, and S. Karunasekera, «A survey of coordinated attacks and collaborative intrusion detection,» *Computers & Security*, vol. 29, pp. 124-140, 2012.

[10] S. Snapp et al., «DIDS (distributed intrusion detection system) – motivation, architecture, and an early prototype,» in *Proceedings of the 14th national computer security conference;*, 1991, pp. 167-176.

[11] Internet Storm Center. [Online]. http://www.dshield.org

[12] RA Kemmerer, «NSTAT: a model-based real-time network intrusion detection system.,» University of California at Santa Barbara, 1998.

[13] S. Staniford-Chen et al., «Grids-a graph based intrusion detection system for large networks,» in *Proceedings of the 19th national information systems security conference*, 1996, pp. 361-370.

[14] P. Porras and P. Neumann, «Emerald: event monitoring enabling responses to anomalous live disturbances,» in *Proceedings of the 20th national information systems security conference*, 1997, pp. 353-365.

[15] RB. Abdoul Karim Ganame, J. Bourgeoisa, and F. Spiesa, «A global security architecture for intrusion detection on computer networks,» *Computers & Security*, vol. 27, pp. 30-47, 2008.

[16] V. Yegneswaran, P. Barford, and S. Jha, «Global intrusion detection in the DOMINO overlay system,» in *Proceedings of network and distributed security symposium (NDSS)*, 2004.

[17] M. Locasto, J. Parekh, A. Keromytis, and S. Stolfo, «Towards collaborative security and P2P intrusion detection,» in *Proceedings of the 2005 IEEE workshop on information assurance and security*, 2005, pp. 333-339.

[18] H. Yu, «A Survey of Trust and Reputation Management Systems in Wireless Communications,» *Proceedings of the IEEE*, vol. 98, no. 10, pp. 1755-1772 , 2010.

[19] A. Josanga, R. Ismailb, and C. Boyd, «A survey of trust and reputation systems for online service provision,» *Decision Support Systems*, vol. 43, no. 2, pp. 618–644, 2007.

[20] J.H. Cho, A. Swarmi, and I.R. Chen, «A Survey on Trust Management for Mobile Ad Hoc Networks,» *IEEE Communications Surveys & Tutorials*, vol. 13, no. 4, pp. 562-583, 2011.

[21] D.E. Bell and L.J. La Padula, «Secure Computer Systems: Unified Exposition and Multics Interpretation,» The MITRE Corporation, ESD-TR-75-306 1976.

[22] MITRE, «Horizontal integration: Broader access models for realizing information dominance,» Availabe at http://www.fas.org/irp/agency/ dod/jason/classpol.pdf, 2004.

[23] P.-C. Chen et al., «Fuzzy multi–level security: An experiment on quantified risk adaptive access control,» in *IEEE Symposium on Security and Privacy*, 2007, pp. 222-230.

[24] J.A. Clark et al., «Risk based access control with uncertain and time-dependent sensitivity,» in *SECRYPT*, 2010, pp. 5-13.

[25] Q. Ni, E. Bertino, and J. Lobo, «Risk-based access control systems built on fuzzy inferences,» in *ASIACCS*, 2010.

[26] S. Kandala, R. Sandhu, and V. Bhamidipati, «An attribute-based framework for risk-adaptive access control models,» in *ARES*, 2011.

[27] J. Hu, R. Li, Z. Lu, J. Lu, and X. Ma, «RAR: A role-and-risk based flexible framework for secure collaboration,» *Future Generation Computer Systems*, vol. 27, pp. 574-586, 2011.

[28] L. Zhang, A. Brodsky, and S. Jajodia, «Toward Information Sharing: Benefit and Risk Access Control (BARAC),» in *POLICY*, 2006, pp. 45-53.

[29] R.A. Best Jr, «Need-to-Know vs. Need-to-Share,» Congressional Research Service, 7-5700, 2011.

[30] A. Gibbons, *Algorithmic Graph Theory*. Cambridge, UK: Cambridge University Press, 1985.

[31] ENISA, «Incentives and Challenges for Information Sharing in the Context of Network and Information Security,» 2010.

[32] L. Zhu, Y. Xu, P. Scerri, and H. Liang, «An Information Sharing Algorithm For Large Dynamic Mobile Multi-agent Teams,» in *11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2012.

APPENDIX A: MSM ACRONYMS

CAPEC – Common Attack Pattern Enumeration and Classification.

CCE – Common Configuration Enumeration.

CCSS – Configuration Scoring System.

CPE – Common Platform Enumeration.

CVE – Common Vulnerabilities and Exposures.

CVRF – Common Frameworks for Vulnerability Disclosure and Response.

CVSS – Common Vulnerability Scoring System.

CWE – Common Weakness Enumeration.

CWSS – Common Weakness Scoring System.

CybOX – Cyber Observable Expression.

CYBEX – The Cybersecurity Information Exchange Framework.

IODEF – Incident Object Description Exchange Format.

MAEC™ – Malware Attribute Enumeration and Characterization.

OVAL – Open Vulnerability and Assessment Language.

OCIL – Open Checklist Interactive Language.

RID – Real-time Inter-network Defense.

RID-T – Transport of Real-time Inter-network Defense.

SBVR – Semantics of Business Vocabulary and Business Rules.

STIX – Structured Threat Information Expression.

SWIDs – Software Identification Tags.

XCCDF – Extensible Configuration Checklist Description Format.